Signing with higher dimensional isogenies

Pierrick Dartois

Joint work with Antonin Leroux, Damien Robert and Benjamin Wesolowski
Acknowledgements to Luca De Feo

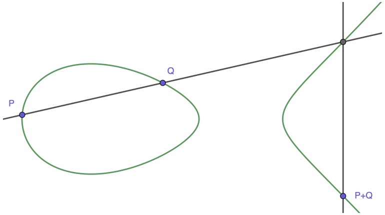
12 september 2023

Institut de Mathématiques de Bordeaux

CANARI

Quaternions and isogenies

Elliptic curves



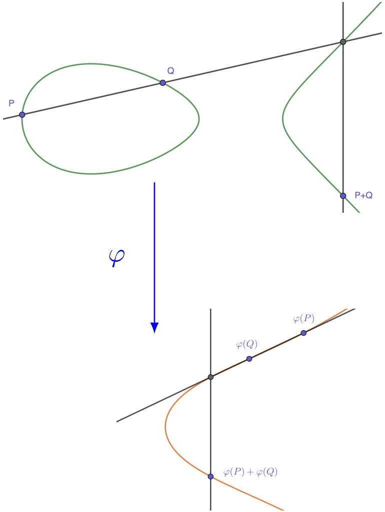
Elliptic curves:

- An elliptic curve $E/\mathbb{F}_q$ is defined by:

$$y^2 = x^3 + ax + b, \quad a, b \in \mathbb{F}_q$$

  with an infinite element 0_E.
- E is equipped with a commutative group law.

Isogenies - Definition



Isogenies:

Maps between elliptic curves $\varphi : E_1 \longrightarrow E_2$ such that:

- φ is a homomorphism of algebraic varieties.
- φ is a group homomorphism:

$$\varphi(P+Q) = \varphi(P) + \varphi(Q).$$

Isogenies - Examples

- The scalar multiplication $[n] : E \longrightarrow E$ is an isogeny.

Isogenies - Examples

- The scalar multiplication $[n] : E \longrightarrow E$ is an isogeny.
- The Frobenius:

$$\begin{array}{rcl} \pi_p : E & \longrightarrow & E^{(p)} \\ (x, y) & \longmapsto & (x^p, y^p) \end{array}$$

with $E/\mathbb{F}_{p^n}$ is an isogeny.

Isogenies - Examples

- The scalar multiplication $[n] : E \longrightarrow E$ is an isogeny.
- The Frobenius:

$$\begin{array}{rcl} \pi_p : E & \longrightarrow & E^{(p)} \\ (x, y) & \longmapsto & (x^p, y^p) \end{array}$$

  with $E/\mathbb{F}_{p^n}$ is an isogeny.
- An explicit example: Consider

$$E_1 : y^2 = x^3 + x + 4 \quad \text{and} \quad E_2 : y^2 = x^3 - x + 4$$

  over $\mathbb{F}_7$. Then

$$\begin{array}{rcl} \varphi : E_1 & \longrightarrow & E_2 \\ (x, y) & \longmapsto & \left(\dfrac{x^2 - 2x - 1}{x - 2}, y \dfrac{x^2 + 3x - 2}{(x - 2)^2} \right) \end{array}$$

  is an isogeny.

Isogenies - the degree

Definition (The degree)

- Measures the "size" of an isogeny. More precisely

$$\deg(\varphi) = [k(E_1) : \varphi^* k(E_2)] = \max(\deg(f), \deg(g))$$

  where $\varphi(x, y) = (f(x)/g(x), \cdots)$.
- Is multiplicative: $\deg(\varphi \circ \psi) = \deg(\varphi)\deg(\psi)$.

If $\deg(\varphi) = n$, we say that φ is an n-isogeny.

Isogenies - the degree

Definition (The degree)

- Measures the "size" of an isogeny. More precisely
$$\deg(\varphi) = [k(E_1) : \varphi^* k(E_2)] = \max(\deg(f), \deg(g))$$
where $\varphi(x, y) = (f(x)/g(x), \cdots)$.
- Is multiplicative: $\deg(\varphi \circ \psi) = \deg(\varphi) \deg(\psi)$.

If $\deg(\varphi) = n$, we say that φ is an n-isogeny.

Definition (Separability)

We say that φ is separable if $\# \ker(\varphi) = \deg(\varphi)$.

Isogenies - the degree

Definition (The degree)

- Measures the "size" of an isogeny. More precisely

$$\deg(\varphi) = [k(E_1) : \varphi^* k(E_2)] = \max(\deg(f), \deg(g))$$

where $\varphi(x, y) = (f(x)/g(x), \cdots)$.

- Is multiplicative: $\deg(\varphi \circ \psi) = \deg(\varphi)\deg(\psi)$.

If $\deg(\varphi) = n$, we say that φ is an n-isogeny.

Definition (Separability)

We say that φ is separable if $\# \ker(\varphi) = \deg(\varphi)$.

Definition (Dual isogeny)

If $\varphi : E_1 \longrightarrow E_2$ the dual $\widehat{\varphi} : E_2 \longrightarrow E_1$ satisfies $\widehat{\varphi} \circ \varphi = [\deg(\varphi)]_{E_1}$ and $\deg(\varphi) = \deg(\widehat{\varphi})$.

Isogenies - Examples

- The scalar multiplication $[n] : E \longrightarrow E$ is an isogeny of **degree** n^2.
- The Frobenius:

$$\begin{array}{rcl} \pi_p : E & \longrightarrow & E^{(p)} \\ (x, y) & \longmapsto & (x^p, y^p) \end{array}$$

  with $E/\mathbb{F}_{p^n}$ is an **inseparable** p**-isogeny**.
- An explicit example: Consider

$$E_1 : y^2 = x^3 + x + 4 \quad \text{and} \quad E_2 : y^2 = x^3 - x + 4$$

  over $\mathbb{F}_7$. Then

$$\begin{array}{rcl} \varphi : E_1 & \longrightarrow & E_2 \\ (x, y) & \longmapsto & \left(\dfrac{x^2 - 2x - 1}{x - 2}, y \dfrac{x^2 + 3x - 2}{(x-2)^2} \right) \end{array}$$

  is a **separable** 2-**isogeny**.

The Endomorphism ring

Definition (Endomorphism ring)

$$\mathrm{End}(E) = \{0\} \cup \{\text{Isogenies } \varphi : E \longrightarrow E\}$$

Defines a ring for the addition and composition of isogenies.

The Endomorphism ring

Definition (Endomorphism ring)

$$\mathrm{End}(E) = \{0\} \cup \{\text{Isogenies } \varphi : E \longrightarrow E\}$$

Defines a ring for the addition and composition of isogenies.

Theorem (Deuring)

Let $E/\mathbb{F}_q$ ($p = \mathrm{char}(\mathbb{F}_q)$). Then $\mathrm{End}(E)$ is either isomorphic to:

- *An order in a quadratic imaginary field. We say that E is ordinary.*
- *A maximal order in a quaternion algebra ramifying at p and ∞. We say that E is supersingular.*

If E is supersingular, we may assume $\mathbb{F}_q = \mathbb{F}_{p^2}$.

Quaternions - Definitions

- **Quaternion algebra ramifying at p and ∞:** A 4-dimensional non commutative division algebra over $\mathbb{Q}$:

$$\mathcal{B}_{p,\infty} = \mathbb{Q} \oplus \mathbb{Q}i \oplus \mathbb{Q}j \oplus \mathbb{Q}k,$$

with

$$i^2 = -1 \text{ (if } p \equiv 3 \mod 4), \quad j^2 = -p \quad \text{and} \quad k = ij = -ji.$$

Quaternions - Definitions

- **Quaternion algebra ramifying at p and ∞:** A 4-dimensional non commutative division algebra over $\mathbb{Q}$:

$$\mathcal{B}_{p,\infty} = \mathbb{Q} \oplus \mathbb{Q}i \oplus \mathbb{Q}j \oplus \mathbb{Q}k,$$

with

$$i^2 = -1 \text{ (if } p \equiv 3 \mod 4), \quad j^2 = -p \quad \text{and} \quad k = ij = -ji.$$

- **Order:** A full rank lattice $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ with a ring structure.
- **Maximal Order:** An order $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ such that for any other order $\mathcal{O}' \supseteq \mathcal{O}$, we have $\mathcal{O}' = \mathcal{O}$.

Quaternions - Definitions

- **Quaternion algebra ramifying at p and ∞:** A 4-dimensional non commutative division algebra over $\mathbb{Q}$:

$$\mathcal{B}_{p,\infty} = \mathbb{Q} \oplus \mathbb{Q}i \oplus \mathbb{Q}j \oplus \mathbb{Q}k,$$

with

$$i^2 = -1 \text{ (if } p \equiv 3 \mod 4), \quad j^2 = -p \quad \text{and} \quad k = ij = -ji.$$

- **Order:** A full rank lattice $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ with a ring structure.
- **Maximal Order:** An order $\mathcal{O} \subset \mathcal{B}_{p,\infty}$ such that for any other order $\mathcal{O}' \supseteq \mathcal{O}$, we have $\mathcal{O}' = \mathcal{O}$.
- **Left Ideal:** A left $\mathcal{O}$-ideal I is a full rank lattice $I \subset \mathcal{B}_{p,\infty}$ such that $\mathcal{O} \cdot I = I$.
- **Right Ideal:** A right $\mathcal{O}$-ideal I is a full rank lattice $I \subset \mathcal{B}_{p,\infty}$ such that $I \cdot \mathcal{O} = I$.

Quaternions - Definitions

- **Conjugation:**

$$\alpha = x + yi + zj + tk \longmapsto \overline{\alpha} = x - yi - zj - tk$$

Quaternions - Definitions

- **Conjugation:**

$$\alpha = x + yi + zj + tk \longmapsto \overline{\alpha} = x - yi - zj - tk$$

- **Norm:** $\mathrm{nrd}(\alpha) := \alpha\overline{\alpha} = x^2 + y^2 + p(z^2 + t^2)$.

Quaternions - Definitions

- **Conjugation:**

$$\alpha = x + yi + zj + tk \longmapsto \overline{\alpha} = x - yi - zj - tk$$

- **Norm:** $\mathrm{nrd}(\alpha) := \alpha\overline{\alpha} = x^2 + y^2 + p(z^2 + t^2)$.
- **Ideal norm:** $\mathrm{nrd}(I) := \gcd\{\mathrm{nrd}(\alpha) \mid \alpha \in I\}$.
- **Ideal conjugate:** $\overline{I} := \{\overline{\alpha} \mid \alpha \in I\}$.

Quaternions - Definitions

- **Conjugation:**

$$\alpha = x + yi + zj + tk \longmapsto \overline{\alpha} = x - yi - zj - tk$$

- **Norm:** $\mathrm{nrd}(\alpha) := \alpha\overline{\alpha} = x^2 + y^2 + p(z^2 + t^2)$.
- **Ideal norm:** $\mathrm{nrd}(I) := \gcd\{\mathrm{nrd}(\alpha) \mid \alpha \in I\}$.
- **Ideal conjugate:** $\overline{I} := \{\overline{\alpha} \mid \alpha \in I\}$.
- **Equivalent left $\mathcal{O}$-ideals:** $I \sim J \Longleftrightarrow \exists \alpha \in \mathcal{B}_{p,\infty}^*, \quad J = I\alpha$.

The Deuring correspondence

Supersingular elliptic curves	Quaternions
$j(E)$ or $j(E)^p$ supersingular	$\mathcal{O} \cong \mathrm{End}(E)$ maximal order in $\mathcal{B}_{p,\infty}$
$\varphi : E \longrightarrow E'$	left $\mathcal{O}$-ideal and right $\mathcal{O}'$-ideal I_φ
$\varphi, \psi : E \longrightarrow E'$	$I_\varphi \sim I_\psi$ $(I_\psi = I_\varphi \alpha)$
$\widehat{\varphi}$	$\overline{I_\varphi}$
$\varphi \circ \psi$	$I_\psi \cdot I_\varphi$
$\theta \in \mathrm{End}(E)$	Principal ideal $\mathcal{O}\theta$
$\deg(\varphi)$	$\mathrm{nrd}(I_\varphi)$

Computing isogenies (Easy)

Proposition

A separable isogeny is determined by its kernel. If $\varphi : E_1 \longrightarrow E_2$ *and* $\varphi' : E_1 \longrightarrow E_2'$ *are separable and* $\ker(\varphi) = \ker(\varphi')$, *then there exists an isomorphism* $\lambda : E_2 \xrightarrow{\sim} E_2'$ *such that* $\varphi' = \lambda \circ \varphi$.

Computing isogenies (Easy)

Proposition

A separable isogeny is determined by its kernel. If $\varphi : E_1 \longrightarrow E_2$ and $\varphi' : E_1 \longrightarrow E_2'$ are separable and $\ker(\varphi) = \ker(\varphi')$, then there exists an isomorphism $\lambda : E_2 \xrightarrow{\sim} E_2'$ such that $\varphi' = \lambda \circ \varphi$.

- Given $G = \ker(\varphi)$, we can compute φ in time $O(\sqrt{\#G})$ [BDLS20].

Computing isogenies (Easy)

Proposition

A separable isogeny is determined by its kernel. If $\varphi : E_1 \longrightarrow E_2$ *and* $\varphi' : E_1 \longrightarrow E_2'$ *are separable and* $\ker(\varphi) = \ker(\varphi')$*, then there exists an isomorphism* $\lambda : E_2 \xrightarrow{\sim} E_2'$ *such that* $\varphi' = \lambda \circ \varphi$*.*

- Given $G = \ker(\varphi)$, we can compute φ in time $O(\sqrt{\#G})$ [BDLS20].
- Given $P \in E$ such that $G = \langle P \rangle$ and if $\deg(\varphi) = \ell^n$ with ℓ small, we compute $\varphi : E \longrightarrow E'$ as a sequence of ℓ-isogenies

$$E = E_0 \longrightarrow E_1 \longrightarrow \cdots \longrightarrow E_{n-1} \longrightarrow E_n$$

in time $O(n \log(n))$ [JD11].

Computing isogenies (Easy)

Proposition

A separable isogeny is determined by its kernel. If $\varphi : E_1 \longrightarrow E_2$ and $\varphi' : E_1 \longrightarrow E_2'$ are separable and $\ker(\varphi) = \ker(\varphi')$, then there exists an isomorphism $\lambda : E_2 \xrightarrow{\sim} E_2'$ such that $\varphi' = \lambda \circ \varphi$.

- Given $G = \ker(\varphi)$, we can compute φ in time $O(\sqrt{\#G})$ [BDLS20].
- Given $P \in E$ such that $G = \langle P \rangle$ and if $\deg(\varphi) = \ell^n$ with ℓ small, we compute $\varphi : E \longrightarrow E'$ as a sequence of ℓ-isogenies

$$E = E_0 \longrightarrow E_1 \longrightarrow \cdots \longrightarrow E_{n-1} \longrightarrow E_n$$

  in time $O(n \log(n))$ [JD11].
- Only smooth degree isogenies can be computed efficiently.

The Supersingular Isogeny Problem (Hard)

Problem (Supersingular Isogeny Problem)

Given two supersingular elliptic curves $E_1, E_2/\mathbb{F}_{p^2}$, find an isogeny $\varphi : E_1 \longrightarrow E_2$.

When p has cryptographic size, this problem is hard for quantum computers.

The Supersingular Endomorphism Ring Problem (Hard)

Problem (Supersingular Endomorphism Ring Problem)

Given a supersingular elliptic curve $E/\mathbb{F}_{p^2}$, *compute* $\mathrm{End}(E)$.

The Supersingular Endomorphism Ring Problem (Hard)

Problem (Supersingular Endomorphism Ring Problem)

Given a supersingular elliptic curve $E/\mathbb{F}_{p^2}$, compute $\mathrm{End}(E)$.

An easy instance: Consider $E_0 : y^2 = x^3 + x$ over $\mathbb{F}_p$ ($p \equiv 3 \mod 4$) and

$$\pi_p : (x, y) \in E_0 \longmapsto (x^p, y^p) \in E_0$$

$$\iota : (x, y) \in E_0 \longmapsto (x, \sqrt{-1}y) \in E_0$$

Then $\pi_p^2 = [-p]$, $\iota^2 = [-1]$ and $\pi_p \circ \iota = -\iota \circ \pi_p$ and

$$\mathrm{End}(E_0) = \left\langle 1, \iota, \frac{\iota + \pi_p}{2}, \frac{1 + \iota\pi_p}{2} \right\rangle \cong \left\langle 1, i, \frac{i + j}{2}, \frac{1 + k}{2} \right\rangle$$

The Supersingular Endomorphism Ring Problem (Hard)

Problem (Supersingular Endomorphism Ring Problem)

Given a supersingular elliptic curve $E/\mathbb{F}_{p^2}$*, compute* $\mathrm{End}(E)$.

An easy instance: Consider $E_0 : y^2 = x^3 + x$ over $\mathbb{F}_p$ ($p \equiv 3 \mod 4$) and

$$\pi_p : (x, y) \in E_0 \longmapsto (x^p, y^p) \in E_0$$
$$\iota : (x, y) \in E_0 \longmapsto (x, \sqrt{-1}y) \in E_0$$

Then $\pi_p^2 = [-p]$, $\iota^2 = [-1]$ and $\pi_p \circ \iota = -\iota \circ \pi_p$ and

$$\mathrm{End}(E_0) = \left\langle 1, \iota, \frac{\iota + \pi_p}{2}, \frac{1 + \iota\pi_p}{2} \right\rangle \cong \left\langle 1, i, \frac{i + j}{2}, \frac{1 + k}{2} \right\rangle$$

Theorem (Wesolowski, 2022)

The Supersingular Isogeny Problem and the Supersingular Endomorphism Ring Problem are equivalent.

Quaternion path problem (Easy)

Problem (Connecting ideal)

Given two maximal orders $\mathcal{O}_1, \mathcal{O}_2 \subset \mathcal{B}_{p,\infty}$, find a left $\mathcal{O}_1$-ideal I that is also a right $\mathcal{O}_2$-ideal.

This is simple arithmetic $I \sim \mathcal{O}_1 \cdot \mathcal{O}_2$.

Problem (Quaternion path problem)

Given a left $\mathcal{O}$-ideal I, find $J \sim I$ of smooth norm.

Solved in polynomial time by the KLPT algorithm [KLPT14; DKLPW20].

Quaternion path problem (Easy)

Computing isogenies via the Deuring correspondence:

- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \mathrm{End}(E_1)$ and $\mathcal{O}_2 \cong \mathrm{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$.
- Compute $J \sim I$ of smooth norm via KLPT.
- Translate J into an isogeny $\varphi_J : E_1 \longrightarrow E_2$.

Quaternion path problem (Easy)

Computing isogenies via the Deuring correspondence:

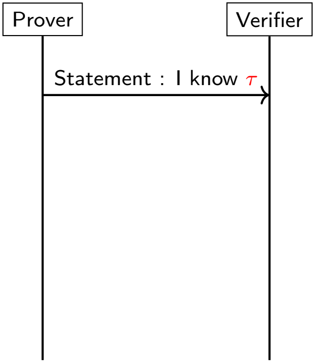
- Let E_1 and E_2 of known endomorphism rings $\mathcal{O}_1 \cong \text{End}(E_1)$ and $\mathcal{O}_2 \cong \text{End}(E_2)$.
- Compute a connecting ideal I between $\mathcal{O}_1$ and $\mathcal{O}_2$.
- Compute $J \sim I$ of smooth norm via KLPT.
- Translate J into an isogeny $\varphi_J : E_1 \longrightarrow E_2$.

Becomes hard when $\text{End}(E_1)$ or $\text{End}(E_2)$ is unknown.

SQISign and effective Deuring correspondence

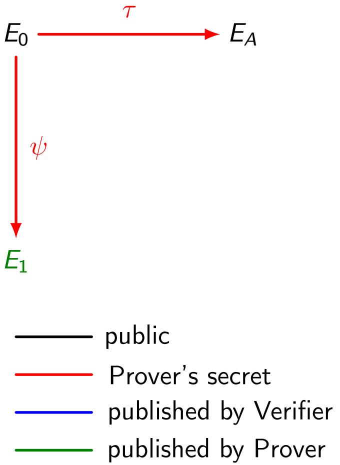
The SQISign identification scheme [DKLPW20; DLW22]

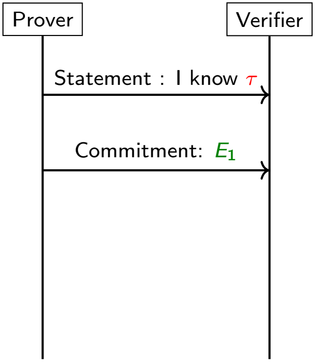
$$E_0 \xrightarrow{\tau} E_A$$



- public
- Prover's secret
- published by Verifier
- published by Prover

The SQISign identification scheme [DKLPW20; DLW22]

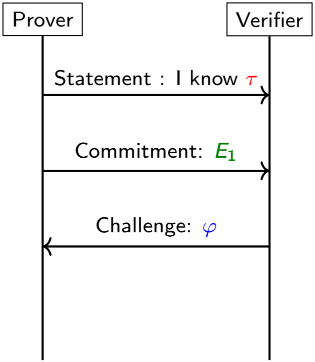




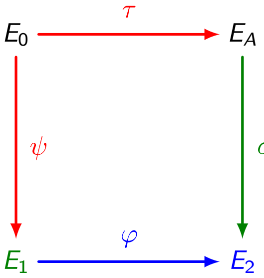
The SQISign identification scheme [DKLPW20; DLW22]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"] & E_A \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

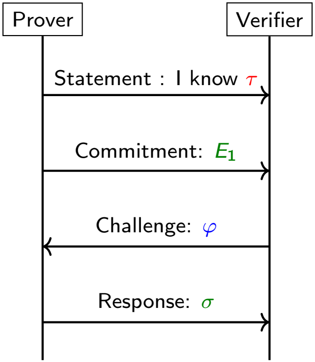
public
Prover's secret
published by Verifier
published by Prover



The SQISign identification scheme [DKLPW20; DLW22]



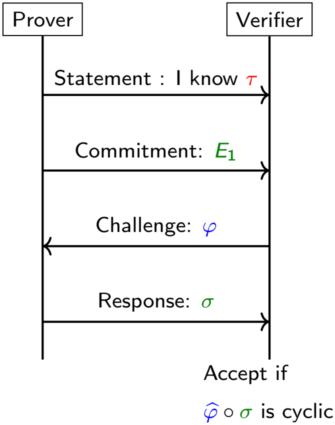
public
Prover's secret
published by Verifier
published by Prover



The SQISign identification scheme [DKLPW20; DLW22]

$$
\begin{tikzcd}
E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, "\sigma"] \\
E_1 \arrow[r, "\varphi"] & E_2
\end{tikzcd}
$$

- public
- Prover's secret
- published by Verifier
- published by Prover



Fiat-Shamir transform [FS87]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public
Signer's secret
published by Signer

Signature: message m, public key E_A, secret key τ.

- Commitment $\psi : E_0 \longrightarrow E_1$.
- Challenge $\varphi := H(E_1, m)$ (where H is a hash function).
- Compute and send signature (E_1, σ) to the verifier.

Fiat-Shamir transform [FS87]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public
Signer's secret
published by Signer

Signature: message m, public key E_A, secret key τ.

- Commitment $\psi : E_0 \longrightarrow E_1$.
- Challenge $\varphi := H(E_1, m)$ (where H is a hash function).
- Compute and send signature (E_1, σ) to the verifier.

Verification: $E_A, m, (E_1, \sigma)$.

- Recompute $\varphi := H(E_1, m)$.
- Verify that $\widehat{\varphi} \circ \sigma$ is cyclic.

How to compute the signature σ ?

$$
\begin{tikzcd}
E_0 \arrow[r, "\tau", "I_\tau"'] \arrow[d, "\psi", "I_\psi"'] & E_A \arrow[d, dashed, "\sigma", "I"'] \\
E_1 \arrow[r, "\varphi", "I_\varphi"'] & E_2
\end{tikzcd}
$$

- Compute $J := \overline{I_\tau} \cdot I_\psi \cdot I_\varphi$.
- Find $I \sim J$ random of norm $\mathrm{nrd}(I) = \ell^e$ (KLPT).
- Compute σ associated to I.

Ideal-to-isogeny I [GPS16]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and I a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_I : E \longrightarrow E_I$.

Ideal-to-isogeny I [GPS16]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and I a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_I : E \longrightarrow E_I$.

- Compute
$$E[I] := \{P \in E \mid \forall \alpha \in I, \quad \alpha(P) = 0\}.$$

Ideal-to-isogeny I [GPS16]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and I a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_I : E \longrightarrow E_I$.

- Compute
$$E[I] := \{P \in E \mid \forall \alpha \in I, \quad \alpha(P) = 0\}.$$
- Compute φ_I of kernel $E[I]$ in $O(\mathrm{poly}(\max_{\ell \mid \mathrm{nrd}(I)} \ell))$ operations over $\mathbb{F}_{p^k}$, where $E[I] \subseteq E(\mathbb{F}_{p^k})$.

Ideal-to-isogeny I [GPS16]

Input: $E/\mathbb{F}_{p^2}$ supersingular, $\mathcal{O} \cong \mathrm{End}(E)$ and I a left $\mathcal{O}$-ideal of smooth norm.

Output: $\varphi_I : E \longrightarrow E_I$.

- Compute
$$E[I] := \{P \in E \mid \forall \alpha \in I, \quad \alpha(P) = 0\}.$$
- Compute φ_I of kernel $E[I]$ in $O(\mathrm{poly}(\max_{\ell \mid \mathrm{nrd}(I)} \ell))$ operations over $\mathbb{F}_{p^k}$, where $E[I] \subseteq E(\mathbb{F}_{p^k})$.

Issue: If I is a KLPT output, then $\mathrm{nrd}(I) \simeq p^{15/4} \gg p$ so k is exponentially big. Not practical for SQISign !

Ideal-to-isogeny II [DLW22]

Main idea: Cut the computation into smaller pieces. Write

$$I = I_0 \cdot I_1 \cdots I_{n-1} \quad \text{and} \quad \varphi_I = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

with $\mathrm{nrd}(I_0) = \cdots = \mathrm{nrd}(I_{n-1}) = \ell^f$.

Ideal-to-isogeny II [DLW22]

Main idea: Cut the computation into smaller pieces. Write

$$I = I_0 \cdot I_1 \cdots I_{n-1} \quad \text{and} \quad \varphi_I = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

with $\mathrm{nrd}(I_0) = \cdots = \mathrm{nrd}(I_{n-1}) = \ell^f$.

$$\begin{tikzcd}
E = E_0 \arrow[r, "I_0", "\varphi_0"'] & E_1 \arrow[loop above, "\theta_1"] \arrow[r, "I_1", "\varphi_1"'] & E_2 \arrow[loop above, "\theta_2"] & \cdots & E_{n-1} \arrow[loop above, "\theta_{n-1}"] \arrow[r, "I_{n-1}", "\varphi_{n-1}"'] & E_n = E_I
\end{tikzcd}$$

Ideal-to-isogeny II [DLW22]

Main idea: Cut the computation into smaller pieces. Write

$$I = I_0 \cdot I_1 \cdots I_{n-1} \quad \text{and} \quad \varphi_I = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

with $\mathrm{nrd}(I_0) = \cdots = \mathrm{nrd}(I_{n-1}) = \ell^f$.

$$\begin{tikzcd}
E = E_0 \arrow[r, "I_0", "\varphi_0"'] & E_1 \arrow[loop above, "\theta_1"] \arrow[r, "I_1", "\varphi_1"'] & E_2 \arrow[loop above, "\theta_2"] & \cdots & E_{n-1} \arrow[loop above, "\theta_{n-1}"] \arrow[r, "I_{n-1}", "\varphi_{n-1}"'] & E_n = E_I
\end{tikzcd}$$

The endomorphisms θ_i are meant to refresh the ℓ^f-torsion.

Ideal-to-isogeny II [DLW22]

Main idea: Cut the computation into smaller pieces. Write

$$I = I_0 \cdot I_1 \cdots I_{n-1} \quad \text{and} \quad \varphi_I = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

with $\mathrm{nrd}(I_0) = \cdots = \mathrm{nrd}(I_{n-1}) = \ell^f$.

$$\begin{tikzcd}
E = E_0 \arrow[r, "I_0", "\varphi_0"'] & E_1 \arrow[loop above, "\theta_1"] \arrow[r, "I_1", "\varphi_1"'] & E_2 \arrow[loop above, "\theta_2"] & \cdots & E_{n-1} \arrow[loop above, "\theta_{n-1}"] \arrow[r, "I_{n-1}", "\varphi_{n-1}"'] & E_n = E_I
\end{tikzcd}$$

The endomorphisms θ_i are meant to refresh the ℓ^f-torsion.

Torsion requirements: $\ell^f T | p^2 - 1$ so that $E[\ell^f T] \subseteq E(\mathbb{F}_{p^4})$, where $\deg(\theta_i) = T^2$ and $T \simeq p^{5/4}$.

Ideal-to-isogeny II [DLW22]

Main idea: Cut the computation into smaller pieces. Write

$$I = I_0 \cdot I_1 \cdots I_{n-1} \quad \text{and} \quad \varphi_I = \varphi_{n-1} \circ \cdots \circ \varphi_1 \circ \varphi_0$$

with $\mathrm{nrd}(I_0) = \cdots = \mathrm{nrd}(I_{n-1}) = \ell^f$.

$$\begin{tikzcd}
E = E_0 \arrow[r, "I_0", "\varphi_0"'] & E_1 \arrow[loop above, "\theta_1"] \arrow[r, "I_1", "\varphi_1"'] & E_2 \arrow[loop above, "\theta_2"] & \cdots & E_{n-1} \arrow[loop above, "\theta_{n-1}"] \arrow[r, "I_{n-1}", "\varphi_{n-1}"'] & E_n = E_I
\end{tikzcd}$$

The endomorphisms θ_i are meant to refresh the ℓ^f-torsion.

Torsion requirements: $\ell^f T | p^2 - 1$ so that $E[\ell^f T] \subseteq E(\mathbb{F}_{p^4})$, where $\deg(\theta_i) = T^2$ and $T \simeq p^{5/4}$.

Issue: This is slow!

Higher dimensional isogenies

Another method to compute σ [DLRW23]

Issue in SQISign: $\deg(\sigma)$ has to be smooth $\deg(\sigma) = \ell^e \simeq p^{15/4}$.

Another method to compute σ [DLRW23]

Issue in SQISign: $\deg(\sigma)$ has to be smooth $\deg(\sigma) = \ell^e \simeq p^{15/4}$.

Our idea: Take $\deg(\sigma)$ non smooth. Then $\deg(\sigma) \simeq \sqrt{p}$.

Another method to compute σ [DLRW23]

Issue in SQISign: $\deg(\sigma)$ has to be smooth $\deg(\sigma) = \ell^e \simeq p^{15/4}$.

Our idea: Take $\deg(\sigma)$ non smooth. Then $\deg(\sigma) \simeq \sqrt{p}$.

- Evaluate σ on $E_A[\ell^e] \subseteq E_A(\mathbb{F}_{p^2})$.

Another method to compute σ [DLRW23]

Issue in SQISign: $\deg(\sigma)$ has to be smooth $\deg(\sigma) = \ell^e \simeq p^{15/4}$.

Our idea: Take $\deg(\sigma)$ non smooth. Then $\deg(\sigma) \simeq \sqrt{p}$.

- Evaluate σ on $E_A[\ell^e] \subseteq E_A(\mathbb{F}_{p^2})$.
- Use the following algorithm to evaluate σ everywhere.

Theorem (Robert, 2022)

Let $\sigma : E \longrightarrow E'$ of degree $q < \ell^e$. There exists a polynomial time algorithm with:

- ***Input:*** $(\sigma(P_1), \sigma(P_2))$, *where* (P_1, P_2) *is a basis of* $E[\ell^e]$ *and* $Q \in E(\mathbb{F}_{p^2})$.
- ***Output:*** $\sigma(Q)$.

Another method to compute σ [DLRW23]

Issue in SQISign: $\deg(\sigma)$ has to be smooth $\deg(\sigma) = \ell^e \simeq p^{15/4}$.

Our idea: Take $\deg(\sigma)$ non smooth. Then $\deg(\sigma) \simeq \sqrt{p}$.

- Evaluate σ on $E_A[\ell^e] \subseteq E_A(\mathbb{F}_{p^2})$.
- Use the following algorithm to evaluate σ everywhere.

Theorem (Robert, 2022)

Let $\sigma : E \longrightarrow E'$ of degree $q < \ell^e$. There exists a polynomial time algorithm with:

- ***Input:*** *$(\sigma(P_1), \sigma(P_2))$, where (P_1, P_2) is a basis of $E[\ell^e]$ and $Q \in E(\mathbb{F}_{p^2})$.*
- ***Output:*** *$\sigma(Q)$.*

Context: This idea comes from the attacks against SIDH [CD22; MM22; Rob22].

Evaluating σ

Main idea: Use the alternate path $\varphi \circ \psi \circ \widehat{\tau}$.

$$\begin{tikzcd}
E_0 \arrow[r, "\tau", "I_\tau"'] \arrow[d, "\psi"', "I_\psi"] & E_A \arrow[d, dashed, "\sigma", "I"'] \\
E_1 \arrow[r, "\varphi", "I_\varphi"'] & E_2
\end{tikzcd}$$

Evaluating σ

Main idea: Use the alternate path $\varphi \circ \psi \circ \widehat{\tau}$.

$$\begin{tikzcd} E_0 \arrow[r, "\tau", "I_\tau"'] \arrow[d, "\psi"', "I_\psi"] & E_A \arrow[d, dashed, "I"', "\sigma"] \\ E_1 \arrow[r, "\varphi", "I_\varphi"'] & E_2 \end{tikzcd}$$

- Let $\gamma := \widehat{\psi} \circ \widehat{\varphi} \circ \sigma \circ \tau \in \mathrm{End}(E_0)$.
- We have $\mathcal{O}_0\gamma = I_\tau \cdot I \cdot \overline{I_\varphi} \cdot \overline{I_\psi}$ so we can compute γ.

Evaluating σ

Main idea: Use the alternate path $\varphi \circ \psi \circ \widehat{\tau}$.

$$\begin{tikzcd} E_0 \arrow[r, "\tau", "I_\tau"'] \arrow[d, "\psi"', "I_\psi"] & E_A \arrow[d, dashed, "I"', "\sigma"] \\ E_1 \arrow[r, "\varphi", "I_\varphi"'] & E_2 \end{tikzcd}$$

- Let $\gamma := \widehat{\psi} \circ \widehat{\varphi} \circ \sigma \circ \tau \in \mathrm{End}(E_0)$.
- We have $\mathcal{O}_0\gamma = I_\tau \cdot I \cdot \overline{I_\varphi} \cdot \overline{I_\psi}$ so we can compute γ.
- Then:

$$[D_\psi D_\varphi D_\tau]\sigma = \varphi \circ \psi \circ \gamma \circ \widehat{\tau}$$

Evaluating σ

Main idea: Use the alternate path $\varphi \circ \psi \circ \widehat{\tau}$.

$$\begin{tikzcd}
E_0 \arrow[r, "\tau", "I_\tau"'] \arrow[d, "\psi"', "I_\psi"] & E_A \arrow[d, dashed, "I"', "\sigma"] \\
E_1 \arrow[r, "\varphi", "I_\varphi"'] & E_2
\end{tikzcd}$$

- Let $\gamma := \widehat{\psi} \circ \widehat{\varphi} \circ \sigma \circ \tau \in \mathrm{End}(E_0)$.
- We have $\mathcal{O}_0\gamma = I_\tau \cdot I \cdot \overline{I_\varphi} \cdot \overline{I_\psi}$ so we can compute γ.
- Then:

$$[D_\psi D_\varphi D_\tau]\sigma = \varphi \circ \psi \circ \gamma \circ \widehat{\tau}$$

- We can evaluate σ on $P \in E_A[\ell^e]$ provided $(D_\psi D_\varphi D_\tau) \wedge \ell = 1$:

$$\sigma(P) = [\lambda]\varphi \circ \psi \circ \gamma \circ \widehat{\tau}(P),$$

with $\lambda D_\psi D_\varphi D_\tau \equiv 1 \mod \ell^e$.

d-isogenies and the dual isogeny in higher dimension

Definition (d-isogeny)

Let $\varphi : (A, \lambda_A) \longrightarrow (B, \lambda_B)$ be an isogeny between two principally polarized abelian varieties (PPAV). We define:

- $\widetilde{\varphi} := \lambda_A^{-1} \circ \widehat{\varphi} \circ \lambda_B : B \longrightarrow A$.

$$B \xrightarrow{\lambda_B} \widehat{B} \xrightarrow{\widehat{\varphi}} \widehat{A} \xrightarrow{\lambda_A^{-1}} A$$

- We say that φ is a <u>d-isogeny</u> if $\widetilde{\varphi} \circ \varphi = [d]_A$.

Kani's embedding lemma [Kan97]

Definition (isogeny diamond)

An (a, b)-isogeny diamond is a commutative diagram s.t.:

$$\begin{tikzcd} A' \arrow[r, "\varphi'"] & B' \\ A \arrow[u, "\psi"] \arrow[r, "\varphi"] & B \arrow[u, "\psi'"'] \end{tikzcd}$$

where φ, φ' are a-isogenies and ψ, ψ' are b-isogenies.

Lemma (Kani)

Consider the (a, b)-isogeny diamond on the left. Then:

- $F : A \times B' \longrightarrow B \times A'$,

  $$F := \begin{pmatrix} \varphi & \widetilde{\psi}' \\ -\psi & \widetilde{\varphi}' \end{pmatrix}$$

  is a d-isogeny with $d = a + b$.
- *If $a \wedge b = 1$, then*

  $$\ker(F) = \{(\widetilde{\varphi}(x), \psi'(x)) \mid x \in B[d]\}.$$

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

- Let $q = \deg(\sigma)$.
- Let $a_1, a_2 \in \mathbb{Z}$ s.t. $a_1^2 + a_2^2 + q = \ell^e$.

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

- Let $q = \deg(\sigma)$.
- Let $a_1, a_2 \in \mathbb{Z}$ s.t. $a_1^2 + a_2^2 + q = \ell^e$.
- q should be good: $\ell^e - q$ prime $\equiv 1 \mod 4$.

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

- Let $q = \deg(\sigma)$.
- Let $a_1, a_2 \in \mathbb{Z}$ s.t. $a_1^2 + a_2^2 + q = \ell^e$.
- q should be good: $\ell^e - q$ prime $\equiv 1 \mod 4$.
- Consider the isogeny diamond:

$$\begin{tikzcd} E_2^2 \arrow[r, "\alpha_2"] & E_2^2 \\ E_A^2 \arrow[u, "\Sigma"] \arrow[r, "\alpha_A"] & E_A^2 \arrow[u, "\Sigma"'] \end{tikzcd}$$

where $\Sigma := \mathrm{Diag}(\sigma, \sigma)$ and for $i = A, 2$,

$$\alpha_i := \begin{pmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{pmatrix} \in \mathrm{End}(E_i^2).$$

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

- Then

$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Sigma} \\ -\Sigma & \widetilde{\alpha}_A \end{pmatrix} \in \mathrm{End}(E_A^2 \times E_2^2).$$

is an ℓ^e-isogeny.

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

- Then
$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Sigma} \\ -\Sigma & \widetilde{\alpha}_A \end{pmatrix} \in \mathrm{End}(E_A^2 \times E_2^2).$$
is an ℓ^e-isogeny.
- And
$$\ker(F) = \{([a_1]R - [a_2]S, [a_2]R + [a_1]S, \sigma(R), \sigma(S)) \mid R, S \in E_A[\ell^e]\}.$$

Application of Kani's lemma to SQISignHD

Embedding σ in higher dimension:

- Then
$$F := \begin{pmatrix} \alpha_1 & \widetilde{\Sigma} \\ -\Sigma & \widetilde{\alpha}_A \end{pmatrix} \in \mathrm{End}(E_A^2 \times E_2^2).$$
is an ℓ^e-isogeny.
- And
$$\ker(F) = \{([a_1]R - [a_2]S, [a_2]R + [a_1]S, \sigma(R), \sigma(S)) \mid R, S \in E_A[\ell^e]\}.$$
- F can be computed in polynomial time [LR12; LR15; LR23; DLRW23].

Algorithm for higher dimensional isogeny computations

- The ℓ^e-isogeny F can be computed as a chain of ℓ-isogenies:

$$\mathcal{A}_0 \xrightarrow{F_0} \mathcal{A}_1 \xrightarrow{F_2} \mathcal{A}_2 \quad \cdots \quad \mathcal{A}_{e-1} \xrightarrow{F_e} \mathcal{A}_e$$

- Each ℓ-isogeny can be computed in $O(\ell^g)$ efficiently in the Θ-model [LR12; LR15; LR23; DLRW23].
- The whole chain can be computed in time $O(\ell^g e \log(e))$ [JD11; DLRW23].
- This method is valid in any dimension g.

Concrete implementation in dimension 4

- We provide a `sagemath` implementation when $\ell = 2$ with Θ-coordinates of level 2 (16 coordinates).

Concrete implementation in dimension 4

- We provide a sagemath implementation when $\ell = 2$ with Θ-coordinates of level 2 (16 coordinates).
- Concrete test with $q = 3^{79}$ and $e = 128$.
- Runs in 600 ms on a 13th Gen Intel(R) Core(TM) i5-1335U (4600MHz) CPU.

```
Precomputation time: 0.049009084701538086 s
Computation time: 0.616366863250 7324 s
Did we recover the product theta-structure?
True
Does the isogeny chain represent phi well?
True
```

Concrete implementation in dimension 4

- We provide a `sagemath` implementation when $\ell = 2$ with Θ-coordinates of level 2 (16 coordinates).
- Concrete test with $q = 3^{79}$ and $e = 128$.
- Runs in 600 ms on a 13th Gen Intel(R) Core(TM) i5-1335U (4600MHz) CPU.

```
Precomputation time: 0.049009084701538086 s
Computation time: 0.6163668632507324 s
Did we recover the product theta-structure?
True
Does the isogeny chain represent phi well?
True
```

- Target after optimizations: gain a factor 10-50.

SQISignHD: signing with higher dimensional isogenies

SQISignHD identification scheme [DLRW23]

$$\begin{array}{ccc} E_0 & \xrightarrow{\tau} & E_A \\ \Big\downarrow \psi & & \Big\downarrow \sigma \\ E_1 & \xrightarrow{\varphi} & E_2 \end{array}$$

Secret key: τ

Commitment: E_1

Challenge: φ

public

Prover's secret

published by Verifier

published by Prover

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public
Prover's secret
published by Verifier
published by Prover

Secret key: τ

Commitment: E_1

Challenge: φ

Response: $(q, \sigma(P_1), \sigma(P_2))$

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public

Prover's secret

published by Verifier

published by Prover

Secret key: τ

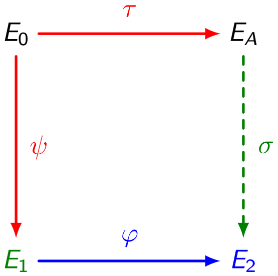
Commitment: E_1

Challenge: φ

Response: $(q, \sigma(P_1), \sigma(P_2))$

- Compute $I \sim \overline{I_\tau} \cdot I_\psi \cdot I_\varphi$ random of norm $q \simeq \sqrt{p}$.

SQISignHD identification scheme [DLRW23]



public
Prover's secret
published by Verifier
published by Prover

Secret key: τ

Commitment: E_1

Challenge: φ

Response: $(q, \sigma(P_1), \sigma(P_2))$

- Compute $I \sim \overline{I_\tau} \cdot I_\psi \cdot I_\varphi$ random of norm $q \simeq \sqrt{p}$.
- Compute a canonical basis (P_1, P_2) of $E_A[\ell^e]$.

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public
Prover's secret
published by Verifier
published by Prover

Secret key: τ

Commitment: E_1

Challenge: φ

Response: $(q, \sigma(P_1), \sigma(P_2))$

- Compute $I \sim \overline{I_\tau} \cdot I_\psi \cdot I_\varphi$ random of norm $q \simeq \sqrt{p}$.
- Compute a canonical basis (P_1, P_2) of $E_A[\ell^e]$.
- Evaluate $\sigma = \varphi_I$ on (P_1, P_2).

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd}
E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\
E_1 \arrow[r, "\varphi"] & E_2
\end{tikzcd}$$

public

Prover's secret

published by Verifier

published by Prover

Secret key: τ

Commitment: E_1

Challenge: φ

Response: $(q, \sigma(P_1), \sigma(P_2))$

- Compute $I \sim \overline{I_\tau} \cdot I_\psi \cdot I_\varphi$ random of norm $q \simeq \sqrt{p}$.
- Compute a canonical basis (P_1, P_2) of $E_A[\ell^e]$.
- Evaluate $\sigma = \varphi_I$ on (P_1, P_2).
- Send $(q, \sigma(P_1), \sigma(P_2))$.

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd}
E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\
E_1 \arrow[r, "\varphi"] & E_2
\end{tikzcd}$$

public

Prover's secret

published by Verifier

published by Prover

Secret key: τ

Commitment: E_1

Challenge: φ

Response: $(q, \sigma(P_1), \sigma(P_2))$

- Compute $I \sim \overline{I_\tau} \cdot I_\psi \cdot I_\varphi$ random of norm $q \simeq \sqrt{p}$.
- Compute a canonical basis (P_1, P_2) of $E_A[\ell^e]$.
- Evaluate $\sigma = \varphi_I$ on (P_1, P_2).
- Send $(q, \sigma(P_1), \sigma(P_2))$.

Very fast !

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

Response: $(q, \sigma(P_1), \sigma(P_2))$

public

Prover's secret

published by Verifier

published by Prover

SQISignHD identification scheme [DLRW23]

$$
\begin{tikzcd}
E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\
E_1 \arrow[r, "\varphi"] & E_2
\end{tikzcd}
$$

public

Prover's secret

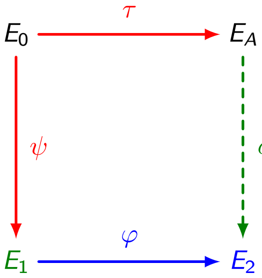
published by Verifier

published by Prover

Response: $(q, \sigma(P_1), \sigma(P_2))$

Verification: Compute the embedding $F \in \mathrm{End}(E_A^2 \times E_2^2)$ of σ.

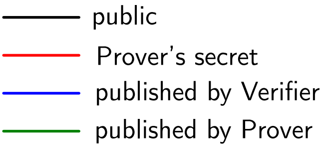
SQISignHD identification scheme [DLRW23]



Response: $(q, \sigma(P_1), \sigma(P_2))$

Verification: Compute the embedding $F \in \mathrm{End}(E_A^2 \times E_2^2)$ of σ.

- Find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 + q = \ell^e$ (Cornacchia).



SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public

Prover's secret

published by Verifier

published by Prover

Response: $(q, \sigma(P_1), \sigma(P_2))$

Verification: Compute the embedding $F \in \mathrm{End}(E_A^2 \times E_2^2)$ of σ.

- Find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 + q = \ell^e$ (Cornacchia).
- Compute the canonical basis (P_1, P_2) of $E_A[\ell^e]$.

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public

Prover's secret

published by Verifier

published by Prover

Response: $(q, \sigma(P_1), \sigma(P_2))$

Verification: Compute the embedding $F \in \mathrm{End}(E_A^2 \times E_2^2)$ of σ.

- Find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 + q = \ell^e$ (Cornacchia).
- Compute the canonical basis (P_1, P_2) of $E_A[\ell^e]$.
- Compute $\ker(F)$, knowing $a_1, a_2, P_1, P_2, \sigma(P_1), \sigma(P_2)$.

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public
Prover's secret
published by Verifier
published by Prover

Response: $(q, \sigma(P_1), \sigma(P_2))$

Verification: Compute the embedding $F \in \mathrm{End}(E_A^2 \times E_2^2)$ of σ.

- Find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 + q = \ell^e$ (Cornacchia).
- Compute the canonical basis (P_1, P_2) of $E_A[\ell^e]$.
- Compute $\ker(F)$, knowing $a_1, a_2, P_1, P_2, \sigma(P_1), \sigma(P_2)$.
- Compute F.

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public
Prover's secret
published by Verifier
published by Prover

Response: $(q, \sigma(P_1), \sigma(P_2))$

Verification: Compute the embedding $F \in \mathrm{End}(E_A^2 \times E_2^2)$ of σ.

- Find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 + q = \ell^e$ (Cornacchia).
- Compute the canonical basis (P_1, P_2) of $E_A[\ell^e]$.
- Compute $\ker(F)$, knowing $a_1, a_2, P_1, P_2, \sigma(P_1), \sigma(P_2)$.
- Compute F.
- Accept if $F \in \mathrm{End}(E_A^2 \times E_2^2)$ and $F(Q, 0, 0, 0) = ([a_1]Q, -[a_2]Q, *, 0)$.

SQISignHD identification scheme [DLRW23]

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public

Prover's secret

published by Verifier

published by Prover

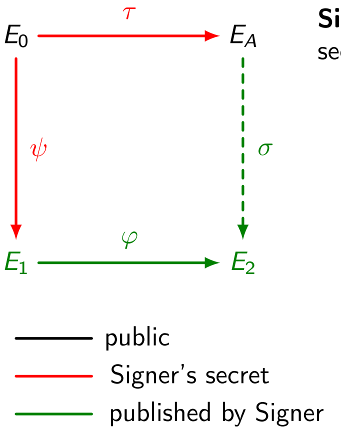
Response: $(q, \sigma(P_1), \sigma(P_2))$

Verification: Compute the embedding $F \in \mathrm{End}(E_A^2 \times E_2^2)$ of σ.

- Find $a_1, a_2 \in \mathbb{Z}$ such that $a_1^2 + a_2^2 + q = \ell^e$ (Cornacchia).
- Compute the canonical basis (P_1, P_2) of $E_A[\ell^e]$.
- Compute $\ker(F)$, knowing $a_1, a_2, P_1, P_2, \sigma(P_1), \sigma(P_2)$.
- Compute F.
- Accept if $F \in \mathrm{End}(E_A^2 \times E_2^2)$ and $F(Q, 0, 0, 0) = ([a_1]Q, -[a_2]Q, *, 0)$.

Implementation in progress.

Fiat-Shamir transform [FS87] of SQISignHD



Signature: message m, public key E_A, secret key τ.

- Commitment $\psi : E_0 \longrightarrow E_1$.
- Challenge $\varphi := H(E_1, m)$ (where H is a hash function).
- Compute and send signature $(E_1, q, \sigma(P_1), \sigma(P_2))$ to the verifier.

Fiat-Shamir transform [FS87] of SQISignHD

$$\begin{tikzcd} E_0 \arrow[r, "\tau"] \arrow[d, "\psi"'] & E_A \arrow[d, dashed, "\sigma"] \\ E_1 \arrow[r, "\varphi"] & E_2 \end{tikzcd}$$

public

Signer's secret

published by Signer

Signature: message m, public key E_A, secret key τ.

- Commitment $\psi : E_0 \longrightarrow E_1$.
- Challenge $\varphi := H(E_1, m)$ (where H is a hash function).
- Compute and send signature $(E_1, q, \sigma(P_1), \sigma(P_2))$ to the verifier.

Verification: $E_A, m, (E_1, q, \sigma(P_1), \sigma(P_2))$.

- Recompute $\varphi := H(E_1, m)$.
- Use $\varphi, q, \sigma(P_1), \sigma(P_2)$ to compute the embedding F of σ.
- Check that $F \in \mathrm{End}(E_A^2 \times E_2^2)$ and $F(Q, 0, 0, 0) = ([a_1]Q, -[a_2]Q, *, 0)$.

Outline of the security analysis

Theorem (Fiat-Shamir, 1986)

Let ID be an identification protocol that is:

- ***Complete:*** *a honest execution is always accepted by the verifier.*
- ***Sound:*** *an attacker cannot "guess" a response.*
- ***Zero-knowledge:*** *the response does not leak any information on the secret key.*

Then the Fiat-Shamir transform of ID is a universally unforgeable signature under chosen message attacks in the random oracle model.

Soundness

Similar to SQISign

Proposition (Special soundness)

Given two transcripts $(E_1, \varphi, q, \sigma(P_1), \sigma(P_2)), (E_1, \varphi', q', \sigma'(P_1'), \sigma'(P_2'))$ *with the same commitment* E_1 *and* $\varphi \neq \varphi'$, *we can extract* $\alpha \in \mathrm{End}(E_A)$ *non-scalar.*

Soundness

Similar to SQISign

Proposition (Special soundness)

Given two transcripts $(E_1, \varphi, q, \sigma(P_1), \sigma(P_2)), (E_1, \varphi', q', \sigma'(P_1'), \sigma'(P_2'))$ with the same commitment E_1 and $\varphi \neq \varphi'$, we can extract $\alpha \in \mathrm{End}(E_A)$ non-scalar.

Proof.

- Exctract σ from $(q, \sigma(P_1), \sigma(P_2))$ and σ' from $(q', \sigma'(P_1'), \sigma'(P_2'))$.
- Then $\alpha := \widehat{\varphi}' \circ \widehat{\sigma}' \circ \sigma \circ \varphi \in \mathrm{End}(E_A)$ is non-scalar.

$\square$

Zero-knowledge

Definition (Recall)

We say that an integer q is **good** if $\ell^e - q$ is a prime $\equiv 1 \mod 4$.

Zero-knowledge

Definition (Recall)

We say that an integer q is **good** if $\ell^e - q$ is a prime $\equiv 1 \mod 4$.

Definition (RUGDIO)

A random uniform good degree isogeny oracle (RUGDIO):
Input: A supersingular elliptic curve $E/\mathbb{F}_{p^2}$.
Output: An isogeny $\sigma : E \longrightarrow E'$ of good degree q s.t.

- E' is uniform among supersingular elliptic curves.
- Given E', σ is uniform among isogenies of good degree $E \longrightarrow E'$.

Zero-knowledge

Theorem

Assume that:

- *E_1 is computationally close to uniform.*
- *We have access to a RUGDIO.*

Then SQISignHD is computationally honest-verifier zero-knowledge.

Zero-knowledge

Theorem

Assume that:

- E_1 *is computationally close to uniform.*
- *We have access to a RUGDIO.*

Then SQISignHD is computationally honest-verifier zero-knowledge.

Proof.

We build a simulator $\mathcal{S}$ of protocol transcripts:

- $\mathcal{S}$ calls the RUGDIO to generate $(q, \sigma(P_1), \sigma(P_2))$.
- $\mathcal{S}$ generates a random challenge $\widehat{\varphi} : E_2 \longrightarrow E_1$.
- $\mathcal{S}$ outputs $(E_1, \varphi, q, \sigma(P_1), \sigma(P_2))$.

$\square$

Zero-knowledge: comparison with SQISign

Heuristic assumptions to prove the zero-knowledge property

In SQISign:

- $\sigma : E_A \longrightarrow E_2$ is computationally indistinguishable from a random isogeny of degree ℓ^e.

In SQISignHD:

- E_1 is computationally close to uniform.
- We have access to a RUGDIO.

Compact signatures

Signature size comparison

	In SQISign	In SQISignHD
Asymptotic (in bits)	$\sim 23/4 \log_2(p)$	$\sim 13/4 \log_2(p)$
NIST-1 security level (in bytes)	204	116

Conclusion

Comparison of SQISignHD with SQISign

	SQISign	SQISignHD
Security	✗ Ad-hoc heuristic: • Distribution of σ.	✓ Simpler heuristics: • RUGDIO; • Distribution of E_1.
Signing time	✗ 400 ms for NIST-1	✓ $<$ 60 ms for NIST-1
Signature size	✓ 204 bytes for NIST-1	✓ 116 bytes for NIST-1
Verification	✓ Fast (6 ms for NIST-1)	✗ 600 ms for NIST-1 in `sagemath`

Thank you for listening.

Find our pre-print here: https://eprint.iacr.org/2023/436