Torsion point images in SIDH: from savior to killer

Tako Boris Fouotsa, LASEC-EPFL

Isogeny Club, 8th November 2022

Introduction 1/3

Isogeny-Based cryptography: very compact keys and mathematically elegant.
But: young field, relatively slow.

First key exchange: CRS[1], uses ordinary isogenies.

Two main issues with using ordinary isogenies:

1. Small amount of smooth rational kernel
2. Arise from class group action : quantum sub-exponential time (CJS 2014)

[1] Couveignes-Rostotsev-Stulbunov 1996/2006

Introduction 1/3

Isogeny-Based cryptography: very compact keys and mathematically elegant.
But: young field, relatively slow.

First key exchange: CRS[1], uses ordinary isogenies.

Two main issues with using ordinary isogenies:

1. Small amount of smooth rational kernel
2. Arise from class group action : quantum sub-exponential time (CJS 2014)

[1] Couveignes-Rostotsev-Stulbunov 1996/2006

Introduction 1/3

Isogeny-Based cryptography: very compact keys and mathematically elegant.
But: young field, relatively slow.

First key exchange: CRS[1], uses ordinary isogenies.

Two main issues with using ordinary isogenies:

1. Small amount of smooth rational kernel
2. Arise from class group action : quantum sub-exponential time (CJS 2014)

[1] Couveignes-Rostotsev-Stulbunov 1996/2006

Jao-De Feo 2011: use supersingular isogenies !

Good news:

1. Large amount of rational torsion available (special primes)
2. No class group action $\Rightarrow$ No known sub-exponential quantum attacks

Bad news: supersingular isogenies do not commute !

Good news: revealing torsion point images solves the issue !

Introduction 2/3

Jao-De Feo 2011: use supersingular isogenies !

Good news:

1. Large amount of rational torsion available (special primes)
2. No class group action $\Rightarrow$ No known sub-exponential quantum attacks

Bad news: supersingular isogenies do not commute !

Good news: revealing torsion point images solves the issue !

Introduction 2/3

Jao-De Feo 2011: use supersingular isogenies !

Good news:

1. Large amount of rational torsion available (special primes)
2. No class group action $\Rightarrow$ No known sub-exponential quantum attacks

Bad news: supersingular isogenies do not commute !

Good news: revealing torsion point images solves the issue !

Introduction 2/3

Jao-De Feo 2011: use supersingular isogenies !

Good news:

1. Large amount of rational torsion available (special primes)
2. No class group action $\Rightarrow$ No known sub-exponential quantum attacks

Bad news: supersingular isogenies do not commute !

Good news: revealing torsion point images solves the issue !

Introduction 3/3

Life was nice till 2016: year where a demon possessed the TP!

GPST 2016: adaptive attack on SIDH, only countered by the FO transform

Petit 2017: torsion point attack on imbalanced SIDH, no impact on SIDH

dQKL+ 2021: improvement on Petit TPA, but SIDH still safe.

FP 2022: new adaptive attack on SIDH using TPA, no impact on SIDH

CD-MM-R 2022, final shot: SIDH/SIKE is broken in seconds...

All these attacks exploit torsion point information !!

Non exhaustive list: BdQL+ 2019, ...

Life was nice till 2016: year where a demon possessed the TP!

GPST 2016: adaptive attack on SIDH, only countered by the FO transform

Petit 2017: torsion point attack on imbalanced SIDH, no impact on SIDH

dQKL+ 2021: improvement on Petit TPA, but SIDH still safe.

FP 2022: new adaptive attack on SIDH using TPA, no impact on SIDH

CD-MM-R 2022, final shot: SIDH/SIKE is broken in seconds...

All these attacks exploit torsion point information !!

Non exhaustive list: BdQL+ 2019, ...

Life was nice till 2016: year where a demon possessed the TP!

GPST 2016: adaptive attack on SIDH,

Petit 2017: torsion point attack on imbalanced SIDH, no impact on SIDH

dQKL+ 2021: improvement on Petit TPA, but SIDH still safe.

FP 2022: new adaptive attack on SIDH using TPA, no impact on SIDH

CD-MM-R 2022, final shot: SIDH/SIKE is broken in seconds...

All these attacks exploit torsion point information !!

Non exhaustive list: BdQL+ 2019, ...

Life was nice till 2016: year where a demon possessed the TP!

GPST 2016: adaptive attack on SIDH,

Petit 2017: torsion point attack on imbalanced SIDH,

dQKL+ 2021: improvement on Petit TPA, but SIDH still safe.

FP 2022: new adaptive attack on SIDH using TPA, no impact on SIDH

CD-MM-R 2022, final shot: SIDH/SIKE is broken in seconds...

All these attacks exploit torsion point information !!

Non exhaustive list: BdQL+ 2019, ...

Introduction 3/3

Life was nice till 2016: year where a demon possessed the TP!

GPST 2016: adaptive attack on SIDH,

Petit 2017: torsion point attack on imbalanced SIDH,

dQKL+ 2021: improvement on Petit TPA, but SIDH still safe.

FP 2022: new adaptive attack on SIDH using TPA, no impact on SIDH

CD-MM-R 2022, final shot: SIDH/SIKE is broken in seconds...

All these attacks exploit torsion point information !!

Non exhaustive list: BdQL+ 2019, ...

Life was nice till 2016: year where a demon possessed the TP!

GPST 2016: adaptive attack on SIDH,

Petit 2017: torsion point attack on imbalanced SIDH,

dQKL+ 2021: improvement on Petit TPA, but SIDH still safe.

FP 2022: new adaptive attack on SIDH using TPA,

CD-MM-R 2022, final shot: SIDH/SIKE is broken in seconds...

All these attacks exploit torsion point information !!

Non exhaustive list: BdQL+ 2019, ...

Introduction 3/3

Life was nice till 2016: year where a demon possessed the TP!

GPST 2016: adaptive attack on SIDH,

Petit 2017: torsion point attack on imbalanced SIDH,

dQKL+ 2021: improvement on Petit TPA, but SIDH still safe.

FP 2022: new adaptive attack on SIDH using TPA,

CD-MM-R 2022, final shot: SIDH/SIKE is broken in seconds...

All these attacks exploit torsion point information !!

Non exhaustive list: BdQL+ 2019, ...

Outline

The role of torsion points in SIDH

GPST adaptive attack on SIDH

A framework for torsion point attacks

A new adaptive attack on SIDH

Summary

The role of torsion points in SIDH

Recall

Elliptic curve $E/\mathbb{F}_q$: abelian group structure, n-torsion group for n $(p \nmid n)$

$$E[n] = \langle P, Q \rangle \simeq \mathbb{Z}/n\mathbb{Z} \oplus \mathbb{Z}/n\mathbb{Z}$$

Supersingular curves:

- $\mathrm{End}(E) \simeq \mathcal{O}_{\max} \subset \mathcal{B}_{p,\infty}$
- defined over $\mathbb{F}_{p^2}$ and $E(\mathbb{F}_{p^2}) \simeq \mathbb{Z}/(p \pm 1)\mathbb{Z} \oplus \mathbb{Z}/(p \pm 1)\mathbb{Z}$
- Smooth order when $p \pm 1$ is smooth

Supersingular cyclic d-isogenies:

- do not commute
- can be defined by a scalar α where $\ker \phi = \langle P + [\alpha]Q \rangle$ and $E[d] = \langle P, Q \rangle$.

Recall

Elliptic curve $E/\mathbb{F}_q$: abelian group structure, n-torsion group for n $(p \nmid n)$

$$E[n] = \langle P, Q \rangle \simeq \mathbb{Z}/n\mathbb{Z} \oplus \mathbb{Z}/n\mathbb{Z}$$

Supersingular curves:

- $\mathrm{End}(E) \simeq \mathcal{O}_{\mathrm{max}} \subset \mathcal{B}_{p,\infty}$
- defined over $\mathbb{F}_{p^2}$ and $E(\mathbb{F}_{p^2}) \simeq \mathbb{Z}/(p \pm 1)\mathbb{Z} \oplus \mathbb{Z}/(p \pm 1)\mathbb{Z}$
- Smooth order when $p \pm 1$ is smooth

Supersingular cyclic d-isogenies:

- do not commute
- can be defined by a scalar α where $\ker \phi = \langle P + [\alpha]Q \rangle$ and $E[d] = \langle P, Q \rangle$.

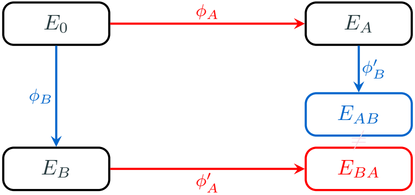
$$
\begin{tikzcd}
E_0 \arrow[r, "\phi_A"] \arrow[dd, "\phi_B"'] & E_A \arrow[d, "\phi'_B"] \\
 & E_{AB} \arrow[d, phantom, "\neq"] \\
E_B \arrow[r, "\phi'_A"'] & E_{BA}
\end{tikzcd}
$$

How would you define ϕ'_A and ϕ'_B? Will the resulting diagram commute?

$$
\begin{tikzcd}
E_0 \arrow[r, "\phi_A"] \arrow[dd, "\phi_B"'] & E_A \arrow[d, "\phi'_B"] \\
& E_{AB} \arrow[d, phantom, "\neq"] \\
E_B \arrow[r, "\phi'_A"'] & E_{BA}
\end{tikzcd}
$$

How would you define ϕ'_A and ϕ'_B? Will the resulting diagram commute?

SIDH



How would you define ϕ'_A and ϕ'_B? Will the resulting diagram commute?

$$\begin{tikzcd}
E_0 \arrow[r, "\phi_A"] \arrow[dd, "\phi_B"'] & E_A \arrow[d, "\phi'_B"] \\
 & E_{AB} \arrow[d, phantom, "\neq"] \\
E_B \arrow[r, "\phi'_A"'] & E_{BA}
\end{tikzcd}$$

How would you define ϕ'_A and ϕ'_B? Will the resulting diagram commute?

SIDH

$$p = N_A N_B - 1, \quad E_0[N_A] = \langle P_A, Q_A \rangle, \quad E_0[N_B] = \langle P_B, Q_B \rangle$$

$$\begin{tikzcd}
\boxed{E_0, P_A, Q_A, P_B, Q_B} \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & \boxed{E_A, \phi_A(P_B), \phi_A(Q_B)} \arrow[d, "\phi'_B"] \\
\boxed{E_B, \phi_B(P_A), \phi_B(Q_A)} \arrow[r, "\phi'_A"'] & \boxed{E_{AB} \;=\; E_{BA}}
\end{tikzcd}$$

$\ker \phi_A = \langle P_A + [\alpha] Q_A \rangle, \quad \ker \phi_B = \langle P_B + [\beta] Q_B \rangle$
$\ker \phi'_A = \langle \phi_B(P_A) + [\alpha] \phi_B(Q_A) \rangle,$
$\ker \phi'_B = \langle \phi_A(P_B) + [\beta] \phi_A(Q_B) \rangle$

SIDH

$$p = N_A N_B - 1, \quad E_0[N_A] = \langle P_A, Q_A \rangle, \quad E_0[N_B] = \langle P_B, Q_B \rangle$$

$$\begin{tikzcd}
E_0, P_A, Q_A, P_B, Q_B \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & E_A, \phi_A(P_B), \phi_A(Q_B) \arrow[d, "\phi'_B"] \\
E_B, \phi_B(P_A), \phi_B(Q_A) \arrow[r, "\phi'_A"'] & E_{AB} = E_{BA}
\end{tikzcd}$$

$$\ker \phi_A = \langle P_A + [\alpha] Q_A \rangle, \quad \ker \phi_B = \langle P_B + [\beta] Q_B \rangle$$

$$\ker \phi'_A = \langle \phi_B(P_A) + [\alpha]\phi_B(Q_A) \rangle,$$

$$\ker \phi'_B = \langle \phi_A(P_B) + [\beta]\phi_A(Q_B) \rangle$$

SIDH

$$p = N_A N_B - 1, \quad E_0[N_A] = \langle P_A, Q_A \rangle, \quad E_0[N_B] = \langle P_B, Q_B \rangle$$

$$\begin{tikzcd}
\boxed{E_0, P_A, Q_A, P_B, Q_B} \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & \boxed{E_A, \phi_A(P_B), \phi_A(Q_B)} \arrow[d, "\phi'_B"] \\
\boxed{E_B, \phi_B(P_A), \phi_B(Q_A)} \arrow[r, "\phi'_A"'] & \boxed{E_{AB} \;=\; E_{BA}}
\end{tikzcd}$$

$\ker \phi_A = \langle P_A + [\alpha] Q_A \rangle, \quad \ker \phi_B = \langle P_B + [\beta] Q_B \rangle$

$\ker \phi'_A = \langle \phi_B(P_A) + [\alpha] \phi_B(Q_A) \rangle,$

$\ker \phi'_B = \langle \phi_A(P_B) + [\beta] \phi_A(Q_B) \rangle$

SIDH

$$p = N_A N_B - 1, \quad E_0[N_A] = \langle P_A, Q_A \rangle, \quad E_0[N_B] = \langle P_B, Q_B \rangle$$

$$\begin{tikzcd}
\boxed{E_0, P_A, Q_A, P_B, Q_B} \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & \boxed{E_A, \phi_A(P_B), \phi_A(Q_B)} \arrow[d, "\phi_B'"] \\
\boxed{E_B, \phi_B(P_A), \phi_B(Q_A)} \arrow[r, "\phi_A'"'] & \boxed{E_{AB} \;=\; E_{BA}}
\end{tikzcd}$$

Validation method: $e_{N_A}(\phi_B(P_A), \phi_B(Q_A)) = e_{N_A}(P_A, Q_A)^{N_B}$

SSI-T Problem: Given E_0, P_B, Q_B, E_A, $\phi_A(P_B)$, $\phi_A(Q_B)$, compute ϕ_A.

SIDH

$$p = N_A N_B - 1, \quad E_0[N_A] = \langle P_A, Q_A \rangle, \quad E_0[N_B] = \langle P_B, Q_B \rangle$$

$$\begin{tikzcd}
\boxed{E_0, P_A, Q_A, P_B, Q_B} \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & \boxed{E_A, \phi_A(P_B), \phi_A(Q_B)} \arrow[d, "\phi'_B"] \\
\boxed{E_B, \phi_B(P_A), \phi_B(Q_A)} \arrow[r, "\phi'_A"'] & \boxed{E_{AB} = E_{BA}}
\end{tikzcd}$$

Validation method: $e_{N_A}(\phi_B(P_A), \phi_B(Q_A)) = e_{N_A}(P_A, Q_A)^{N_B}$

SSI-T Problem: Given E_0, P_B, Q_B, E_A, $\phi_A(P_B)$, $\phi_A(Q_B)$, compute ϕ_A.

GPST adaptive attack on SIDH

An overview

key exchange oracle:

$$O(E, R, S, E') = \begin{cases} 1 & \text{if } E/\langle R + [\alpha]S \rangle = E' \\ 0 & \text{if } E/\langle R + [\alpha]S \rangle \neq E' \end{cases}$$

Idea of the attack: recursively

- Add some well calibrated noise in the TP images
- Use the key exchange oracle determine if the noise was erased during the key exchange or not.
- Deduce a bit of α.

Adding noise := scaling the TP images by a 2×2 matrix M_i

key exchange oracle:

$$O(E, R, S, E') = \begin{cases} 1 & \text{if } E/\langle R + [\alpha]S\rangle = E' \\ 0 & \text{if } E/\langle R + [\alpha]S\rangle \neq E' \end{cases}$$

Idea of the attack: recursively

- Add some well calibrated noise in the TP images
- Use the key exchange oracle determine if the noise was erased during the key exchange or not.
- Deduce a bit of α.

Adding noise := scaling the TP images by a 2×2 matrix M_i

key exchange oracle:

$$O(E, R, S, E') = \begin{cases} 1 & \text{if } E/\langle R + [\alpha]S \rangle = E' \\ 0 & \text{if } E/\langle R + [\alpha]S \rangle \neq E' \end{cases}$$

Idea of the attack: recursively

- Add some well calibrated noise in the TP images
- Use the key exchange oracle determine if the noise was erased during the key exchange or not.
- Deduce a bit of α.

Adding noise := scaling the TP images by a 2×2 matrix M_i

The GPST: recovering α with $N_A = 2^a$

Parity of α : use $M_1 = \begin{bmatrix} 1 & 0 \\ 2^{a-1} & 1 \end{bmatrix}$,

$R_1 = \phi_B(P_A)$ and $S_1 = \phi_B(Q_A) + [2^{a-1}]\phi_B(P_A)]$.

$$\begin{aligned} O(E_B, R_1, S_1, E_{AB}) = 1 \quad &\Leftrightarrow \quad E/\langle R_1 + [\alpha]S_1 \rangle = E_{AB} \\ &\Leftrightarrow^* \quad \langle R_1 + [\alpha]S_1 \rangle = \langle \phi_B(P_A) + [\alpha]\phi_B(Q_A) \rangle \\ &\Leftrightarrow \quad \alpha \text{ is even} \end{aligned}$$

Continuing the attack : write $\alpha = K_i + 2^i\alpha_i + 2^{i+1}\alpha'$.

Use $M_i = \theta \begin{bmatrix} 1 & -2^{a-i-1}K_i \\ 0 & 1 + 2^{a-i-1} \end{bmatrix}$, where $\theta = \sqrt{(1 + 2^{a-i-1})^{-1}}$.

$$O(E_B, R_i, S_i, E_{AB}) = 1 \quad \Leftrightarrow \quad \alpha_i = 0$$

The GPST: recovering α with $N_A = 2^a$

Parity of α : use $M_1 = \begin{bmatrix} 1 & 0 \\ 2^{a-1} & 1 \end{bmatrix}$,
$R_1 = \phi_B(P_A)$ and $S_1 = \phi_B(Q_A) + [2^{a-1}]\phi_B(P_A)]$.

$$\begin{aligned} O(E_B, R_1, S_1, E_{AB}) = 1 \quad & \Leftrightarrow \quad E/\langle R_1 + [\alpha]S_1 \rangle = E_{AB} \\ & \Leftrightarrow^* \quad \langle R_1 + [\alpha]S_1 \rangle = \langle \phi_B(P_A) + [\alpha]\phi_B(Q_A) \rangle \\ & \Leftrightarrow \quad \alpha \text{ is even} \end{aligned}$$

Continuing the attack : write $\alpha = K_i + 2^i\alpha_i + 2^{i+1}\alpha'$.

Use $M_i = \theta \begin{bmatrix} 1 & -2^{a-i-1}K_i \\ 0 & 1 + 2^{a-i-1} \end{bmatrix}$, where $\theta = \sqrt{(1 + 2^{a-i-1})^{-1}}$.

$$O(E_B, R_i, S_i, E_{AB}) = 1 \quad \Leftrightarrow \quad \alpha_i = 0$$

The GPST: recovering α with $N_A = 2^a$

Parity of α : use $M_1 = \begin{bmatrix} 1 & 0 \\ 2^{a-1} & 1 \end{bmatrix}$,
$R_1 = \phi_B(P_A)$ and $S_1 = \phi_B(Q_A) + [2^{a-1}]\phi_B(P_A)]$.

$$\begin{aligned} O(E_B, R_1, S_1, E_{AB}) = 1 \quad & \Leftrightarrow \quad E/\langle R_1 + [\alpha]S_1\rangle = E_{AB} \\ & \Leftrightarrow^* \quad \langle R_1 + [\alpha]S_1\rangle = \langle \phi_B(P_A) + [\alpha]\phi_B(Q_A)\rangle \\ & \Leftrightarrow \quad \alpha \text{ is even} \end{aligned}$$

Continuing the attack : write $\alpha = K_i + 2^i\alpha_i + 2^{i+1}\alpha'$.

Use $M_i = \theta \begin{bmatrix} 1 & -2^{a-i-1}K_i \\ 0 & 1 + 2^{a-i-1} \end{bmatrix}$, where $\theta = \sqrt{(1+2^{a-i-1})^{-1}}$.

$O(E_B, R_i, S_i, E_{AB}) = 1 \quad \Leftrightarrow \quad \alpha_i = 0$

A framework for torsion point attacks

More facts about isogenies

- For any seperable d-isogeny $\varphi : E \to E'$, there exist a unique* d-isogeny $\hat{\varphi} : E' \to E$ called the dual the dual of φ such that $\hat{\varphi} \circ \varphi = [d]_E$ and $\varphi \circ \hat{\varphi} = [d]_{E'}$.
- We have

$$\ker \hat{\varphi} = \varphi(E[d]) \quad \text{and} \quad \ker \varphi = \hat{\varphi}(E'[d]).$$

Take away:

- The knowledge of φ is equivalent to the knowledge of $\hat{\varphi}$.
- You can recover the kernel of a d-isogeny φ by evaluating φ on the d-torsion group.

The framework

SSI-T Problem: Given E_0, $E[N_B] = \langle P, Q \rangle$, E, $\phi(P)$, $\phi(Q)$, compute ϕ.

Degree transformation: define a map Γ that can be used to transform ϕ to $\tau = \Gamma(\phi, input)$ such that:

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ
2. τ can be evaluated on the N_B-torsion
3. τ can be recovered from its action on the N_B-torsion

The attack: Given a suitable description of Γ,

- Use 2. and 3. to recover τ
- Use 1. to derive ϕ from τ

Petit2017 and dQKL+2021

Assumes that $\mathrm{End}(E_0)$ is known. $input = [\theta \in End(E_0),\, d \in \mathbb{Z}]$.

$$\tau = \Gamma(\phi, \theta, d) := [d] + \phi \circ \theta \circ \hat{\phi}$$

s.t. $\deg \tau = N_B^2 e$ with e small.

$$\begin{tikzcd} E_0 \arrow[loop left, "\theta"] \arrow[r, shift left, "\phi"] & E \arrow[l, shift left, "\hat{\phi}"] \arrow[loop right, "\tau"] \end{tikzcd}$$

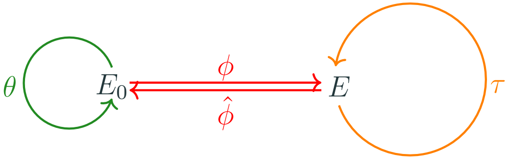
1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ
2. τ can be evaluated on the N_B-torsion
3. τ can be recovered from its action on the N_B-torsion

Petit2017 and dQKL+2021

Assumes that $\mathrm{End}(E_0)$ is known. $input = [\theta \in End(E_0),\, d \in \mathbb{Z}]$.

$$\tau = \Gamma(\phi, \theta, d) := [d] + \phi \circ \theta \circ \hat{\phi}$$

s.t. $\deg \tau = N_B^2 e$ with e small.



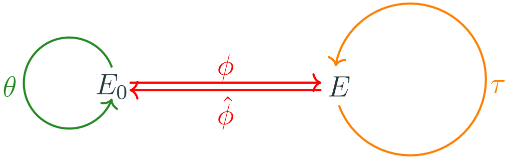
1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ
2. τ can be evaluated on the N_B-torsion
3. τ can be recovered from its action on the N_B-torsion

Petit2017 and dQKL+2021

Assumes that End(E_0) is known. $input = [\theta \in End(E_0),\ d \in \mathbb{Z}]$.

$$\tau = \Gamma(\phi, \theta, d) := [d] + \phi \circ \theta \circ \hat{\phi}$$

s.t. $\deg \tau = N_B^2 e$ with e small.



1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ
2. τ can be evaluated on the N_B-torsion
3. τ can be recovered from its action on the N_B-torsion

Petit2017 and dQKL+2021

Assumes that End(E_0) is known. $input = [\theta \in End(E_0),\, d \in \mathbb{Z}]$.

$$\tau = \Gamma(\phi, \theta, d) := [d] + \phi \circ \theta \circ \hat{\phi}$$

s.t. $\deg \tau = N_B^2 e$ with e small.

$$\begin{tikzcd} E_0 \arrow[loop left, "\theta"] \arrow[rr, shift left, "\phi"] & & E \arrow[ll, shift left, "\hat{\phi}"] \arrow[loop right, "\tau"] \end{tikzcd}$$

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ ✓
2. τ can be evaluated on the N_B-torsion
3. τ can be recovered from its action on the N_B-torsion

$\ker \hat{\phi} =^* \ker(\tau - [d]) \cap E[N_A]$

Petit2017 and dQKL+2021

Assumes that $\mathrm{End}(E_0)$ is known. $input = [\theta \in End(E_0),\, d \in \mathbb{Z}]$.

$$\tau = \Gamma(\phi, \theta, d) := [d] + \phi \circ \theta \circ \hat{\phi}$$

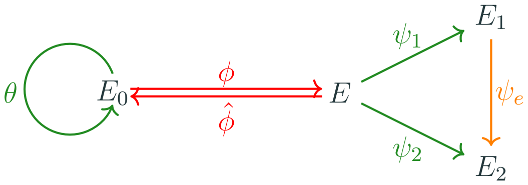
s.t. $\deg \tau = N_B^2 e$ with e small.

$$\begin{tikzcd} E_0 \arrow[loop left, "\theta"] \arrow[r, shift left, "\phi"] & E \arrow[l, shift left, "\hat{\phi}"] \arrow[loop right, "\tau"] \end{tikzcd}$$

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ
2. τ can be evaluated on the N_B-torsion
3. τ can be recovered from its action on the N_B-torsion

$\ker \hat{\phi} =^* \ker(\tau - [d]) \cap E[N_A]$

Since $\deg \tau = N_B^2 e$, then $\tau = \hat{\psi}_2 \circ \psi_e \circ \psi_1$.



- ψ_1 and ψ_2 can be computed from $\phi(P), \phi(Q)$.
- ψ_e is recovered by brute force.

Easy to find good $[\theta \in End(E_0),\ d \in \mathbb{Z}]$ when $N_B > pN_A$.

SIDH : $N_A \approx N_B \approx \sqrt{p}$. Still Secure !

dQKL+2021: τ from its action on the N_B-torsion

Since $\deg \tau = N_B^2 e$, then $\tau = \hat{\psi}_2 \circ \psi_e \circ \psi_1$.

$$
\begin{tikzcd}
 & & E_1 \arrow[dd, "\psi_e"] \\
E_0 \arrow[loop left, "\theta"] \arrow[r, shift left, "\phi"] & E \arrow[l, shift left, "\hat{\phi}"] \arrow[ur, "\psi_1"] \arrow[dr, "\psi_2"'] & \\
 & & E_2
\end{tikzcd}
$$

- ψ_1 and ψ_2 can be computed from $\phi(P), \phi(Q)$.
- ψ_e is recovered by brute force.

Easy to find good $[\theta \in End(E_0),\, d \in \mathbb{Z}]$ when $N_B > pN_A$.

SIDH : $N_A \approx N_B \approx \sqrt{p}$. Still Secure !

CD-MM-R 2022

Assume $\phi : E_0 \longrightarrow E_B$ has degree N_B and the TP have order N_A. Set $a = N_A - N_B = a_1^2 + a_2^2 + a_3^2 + a_4^2$.

$$\tau = \Gamma(\phi, a) := \begin{bmatrix} \alpha_0 & \hat{\phi} Id_4 \\ -\phi Id_4 & \hat{\alpha}_B \end{bmatrix} \in \mathrm{End}(E_0^4 \times E_B^4)$$

where

- $\phi Id_4 : E_0^4 \longrightarrow E_B^4$
- $\hat{\phi} Id_4 : E_B^4 \longrightarrow E_0^4$
- $\alpha_0 \in \mathrm{End}(E_0^4)$ and $\alpha_B \in \mathrm{End}(E_B^4)$ having the same matrix representation

$$M = \begin{bmatrix} a_1 & -a_2 & -a_3 & -a_4 \\ a_2 & a_1 & a_4 & -a_3 \\ a_3 & -a_4 & a_1 & a_2 \\ a_4 & a_3 & -a_2 & a_1 \end{bmatrix}$$

Assume $\phi : E_0 \longrightarrow E_B$ has degree N_B and the TP have order N_A. Set $a = N_A - N_B = a_1^2 + a_2^2 + a_3^2 + a_4^2$.

$$\tau = \Gamma(\phi, a) := \begin{bmatrix} \alpha_0 & \hat{\phi} Id_4 \\ -\phi Id_4 & \hat{\alpha}_B \end{bmatrix} \in \mathrm{End}(E_0^4 \times E_B^4)$$

where

- $\phi Id_4 : E_0^4 \longrightarrow E_B^4$
- $\hat{\phi} Id_4 : E_B^4 \longrightarrow E_0^4$
- $\alpha_0 \in \mathrm{End}(E_0^4)$ and $\alpha_B \in \mathrm{End}(E_B^4)$ having the same matrix representation

$$M = \begin{bmatrix} a_1 & -a_2 & -a_3 & -a_4 \\ a_2 & a_1 & a_4 & -a_3 \\ a_3 & -a_4 & a_1 & a_2 \\ a_4 & a_3 & -a_2 & a_1 \end{bmatrix}$$

Fact: τ has degree $N_B + a = N_A$

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ
2. τ can be evaluated on the N_A-torsion
3. τ can be recovered from its action on the N_A-torsion

Runs in polynomial time !! Breaks SIDH/SIKE/SETA/...

Countermeasures? ongoing...

More details: eprint 2022/1038 or Lorenz's blog post

CD-MM-R 2022

Fact: τ has degree $N_B + a = N_A$

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ
2. τ can be evaluated on the N_A-torsion
3. τ can be recovered from its action on the N_A-torsion

Runs in polynomial time !! Breaks SIDH/SIKE/SETA/...

Countermeasures? ongoing...

More details: eprint 2022/1038 or Lorenz's blog post

Fact: τ has degree $N_B + a = N_A$

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ ✓
2. τ can be evaluated on the N_A-torsion
3. τ can be recovered from its action on the N_A-torsion

Runs in polynomial time !! Breaks SIDH/SIKE/SETA/...

Countermeasures? ongoing...

More details: eprint 2022/1038 or Lorenz's blog post

Fact: τ has degree $N_B + a = N_A$

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ ✓
2. τ can be evaluated on the N_A-torsion ✓
3. τ can be recovered from its action on the N_A-torsion

Runs in polynomial time !! Breaks SIDH/SIKE/SETA/...

Countermeasures? ongoing...

More details: eprint 2022/1038 or Lorenz's blog post

Fact: τ has degree $N_B + a = N_A$

1. Knowing $\tau = \Gamma(\phi, input)$, one can recover ϕ ✓
2. τ can be evaluated on the N_A-torsion ✓
3. τ can be recovered from its action on the N_A-torsion ✓

Runs in polynomial time !! Breaks SIDH/SIKE/SETA/...

Countermeasures? ongoing...

More details: eprint 2022/1038 or Lorenz's blog post

A new adaptive attack on SIDH

key exchange oracle:

$$O(E,R,S,E') = \begin{cases} 1 & \text{if } E/\langle R+[\alpha]S\rangle = E' \\ 0 & \text{if } E/\langle R+[\alpha]S\rangle \neq E' \end{cases}$$

Idea of the attack

1 Actively (using the key exchange oracle) recover the action of ϕ_A on large pairwise disjoint cyclic groups $G_1, G_2, G_3 \subset E_0[NN_B]$ of order NN_B where $p < N$.

2 Use torsion point attacks to recover ϕ_A.

key exchange oracle:

$$O(E, R, S, E') = \begin{cases} 1 & \text{if } E/\langle R + [\alpha]S \rangle = E' \\ 0 & \text{if } E/\langle R + [\alpha]S \rangle \neq E' \end{cases}$$

Idea of the attack

1. Actively (using the key exchange oracle) recover the action of ϕ_A on large pairwise disjoint cyclic groups $G_1, G_2, G_3 \subset E_0[NN_B]$ of order NN_B where $p < N$.
2. Use torsion point attacks to recover ϕ_A.

key exchange oracle:

$$O(E,R,S,E') = \begin{cases} 1 & \text{if } E/\langle R+[\alpha]S\rangle = E' \\ 0 & \text{if } E/\langle R+[\alpha]S\rangle \neq E' \end{cases}$$

Idea of the attack

1. Actively (using the key exchange oracle) recover the action of ϕ_A on large pairwise disjoint cyclic groups $G_1, G_2, G_3 \subset E_0[NN_B]$ of order NN_B where $p < N$.
2. Use torsion point attacks to recover ϕ_A. ✓

Step 1: Recovering the action of ϕ_A on a larger group

Set $N = \prod_{i=1}^{e} \ell_i^2$, $\quad \ell_i$ coprime to $N_A N_B$.

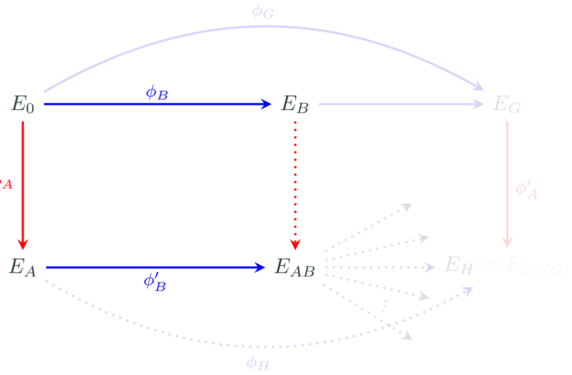
Let $\ell \mid N$ and let G be a cyclic group of order $\ell^2 N_B$.

Query: $O(E_G, R, S, E_H)$, $R = [\ell^{-1}]\phi_G(P_A)$, $S = [\ell^{-1}]\phi_G(Q_A)$

Step 1: Recovering the action of ϕ_A on a larger group

Set $N = \prod_{i=1}^{e} \ell_i^2$, $\quad \ell_i$ coprime to $N_A N_B$.

Let $\ell \mid N$ and let G be a cyclic group of order $\ell^2 N_B$.



Query: $O(E_G, R, S, E_H)$, $R = [\ell^{-1}]\phi_G(P_A)$, $S = [\ell^{-1}]\phi_G(Q_A)$

Step 1: Recovering the action of ϕ_A on a larger group

Set $N = \prod_{i=1}^{e} \ell_i^2$, $\quad \ell_i$ coprime to $N_A N_B$.

Let $\ell \mid N$ and let G be a cyclic group of order $\ell^2 N_B$.

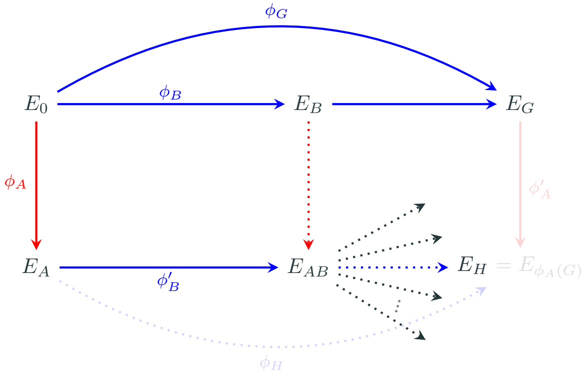
$$
\begin{tikzcd}
E_0 \arrow[r, "\phi_B"] \arrow[rr, bend left, "\phi_G"] \arrow[d, "\phi_A"'] & E_B \arrow[r] \arrow[d, dotted] & E_G \arrow[d, "\phi'_A"] \\
E_A \arrow[r, "\phi'_B"'] \arrow[rr, bend right, dotted, "\phi_H"'] & E_{AB} \arrow[r, dotted] & E_H = E_{\phi_A(G)}
\end{tikzcd}
$$

Query: $O(E_G, R, S, E_H)$, $R = [\ell^{-1}]\phi_G(P_A)$, $S = [\ell^{-1}]\phi_G(Q_A)$

Step 1: Recovering the action of ϕ_A on a larger group

Set $N = \prod_{i=1}^{e} \ell_i^2$, $\quad \ell_i$ coprime to $N_A N_B$.

Let $\ell \mid N$ and let G be a cyclic group of order $\ell^2 N_B$.

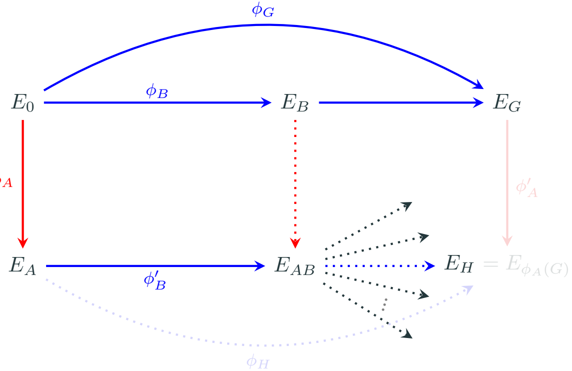


Query: $O(E_G, R, S, E_H)$, $R = [\ell^{-1}]\phi_G(P_A)$, $S = [\ell^{-1}]\phi_G(Q_A)$

Step 1: Recovering the action of ϕ_A on a larger group

Set $N = \prod_{i=1}^{e} \ell_i^2$, $\quad \ell_i$ coprime to $N_A N_B$.

Let $\ell \mid N$ and let G be a cyclic group of order $\ell^2 N_B$.

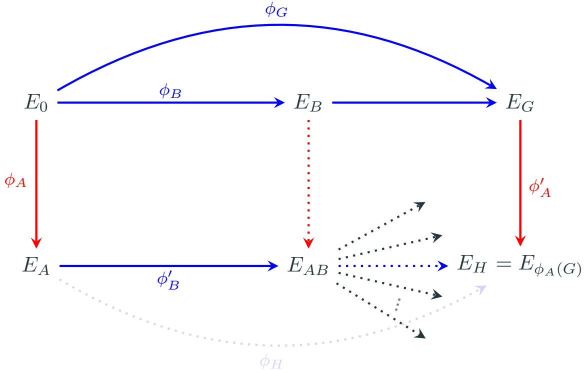


Query: $O(E_G, R, S, E_H)$, $R = [\ell^{-1}]\phi_G(P_A)$, $S = [\ell^{-1}]\phi_G(Q_A)$

Step 1: Recovering the action of ϕ_A on a larger group

Set $N = \prod_{i=1}^{e} \ell_i^2$, $\quad \ell_i$ coprime to $N_A N_B$.

Let $\ell \mid N$ and let G be a cyclic group of order $\ell^2 N_B$.



Query: $O(E_G, R, S, E_H)$, $R = [\ell^{-1}]\phi_G(P_A)$, $S = [\ell^{-1}]\phi_G(Q_A)$

Step 1: Recovering the action of ϕ_A on a larger group

Set $N = \prod_{i=1}^{e} \ell_i^2, \quad \ell_i$ coprime to $N_A N_B$.

Let $\ell \mid N$ and let G be a cyclic group of order $\ell^2 N_B$.

$$
\begin{tikzcd}
E_0 \arrow[r, "\phi_B"] \arrow[rr, bend left, "\phi_G"] \arrow[d, "\phi_A"'] & E_B \arrow[r] \arrow[d, dotted] & E_G \arrow[d, "\phi_A'"] \\
E_A \arrow[r, "\phi_B'"'] \arrow[rr, bend right, dotted, "\phi_H"'] & E_{AB} \arrow[r, dotted] & E_H = E_{\phi_A(G)}
\end{tikzcd}
$$

Query: $O(E_G, R, S, E_H)$, $R = [\ell^{-1}]\phi_G(P_A)$, $S = [\ell^{-1}]\phi_G(Q_A)$

Countermeasures

- Start from a supersingular curve E_0 with unknown endomorphism ring, this would counter the torsion point attacks that are used as subroutine in the attack.
- Use FO-transform as in SIKE: when running the re-encryption step in the FO, Alice will notice that the public key used was malicious.

Summary

Summary

Torsion points have caused the dead of SIDH/SIKE. Any hope for countermeasures? May be:

- Masked-degree SIDH? (Moriya 2022)
- Masked torsion points SIDH? (F. 2022)

Current analysis shows that the primes used should have at least ≈ 6000 bits !

Moreover, they are still vulnerable to adaptive attacks. So would still require FO to have IND-CCA security

More details here and there!

Summary

Torsion points have caused the dead of SIDH/SIKE. Any hope for countermeasures? May be:

- Masked-degree SIDH? (Moriya 2022)
- Masked torsion points SIDH? (F. 2022)

Current analysis shows that the primes used should have at least ≈ 6000 bits !

Moreover, they are still vulnerable to adaptive attacks. So would still require FO to have IND-CCA security

More details here and there!

Summary

Torsion points have caused the dead of SIDH/SIKE. Any hope for countermeasures? May be:

- Masked-degree SIDH? (Moriya 2022)
- Masked torsion points SIDH? (F. 2022)

Current analysis shows that the primes used should have at least ≈ 6000 bits !

Moreover, they are still vulnerable to adaptive attacks. So would still require FO to have IND-CCA security

More details here and there!

Summary

Torsion points have caused the dead of SIDH/SIKE. Any hope for countermeasures? May be:

- Masked-degree SIDH? (Moriya 2022)
- Masked torsion points SIDH? (F. 2022)

Current analysis shows that the primes used should have at least ≈ 6000 bits !

Moreover, they are still vulnerable to adaptive attacks. So would still require FO to have IND-CCA security

More details here and there!

Happy to discuss your comments and questions !!!