Faster $(2,2)$-isogenies for Faster Festive Encryption

Luciano Maino

University of Bristol
S3×E4

14^{th} November, 2023

References

- "FESTA: Fast Encryption from Supersingular Torsion Attacks",
  with Andrea Basso, and Giacomo Pope. (Asiacrypt 2023)
  https://eprint.iacr.org/2023/660
- "An Algorithmic Approach to $(2, 2)$-isogenies in the Theta Model and Applications to Isogeny-based Cryptography",
  with Pierrick Dartois, Giacomo Pope, and Damien Robert.
  https://eprint.iacr.org/2023/1747

Overview

Outline

Attack-based Encryption

- SIDH was introduced by De Feo, Jao, and Plut in 2011.

Attack-based Encryption

- SIDH was introduced by De Feo, Jao, and Plut in 2011.
- Petit (2017) describes an attack on some parameter sets

  $\hookrightarrow$ SÉTA: **S**upersingular **E**ncryption from **T**orsion **A**ttacks (2019)

  De Feo, de Saint Guilhem, Fouotsa, Kutas, Leroux, Petit, Silva, and Wesolowski

Attack-based Encryption

- SIDH was introduced by De Feo, Jao, and Plut in 2011.
- Petit (2017) describes an attack on some parameter sets
  - $\hookrightarrow$ SÉTA: **S**upersingular **E**ncryption from **T**orsion **A**ttacks (2019)
    De Feo, de Saint Guilhem, Fouotsa, Kutas, Leroux, Petit, Silva, and Wesolowski
- New attacks on SIDH (2022)
  - $\hookrightarrow$ FESTA: **F**ast **E**ncryption from **S**upersingular **T**orsion **A**ttacks

Attack-based Encryption

- SIDH was introduced by De Feo, Jao, and Plut in 2011.
- Petit (2017) describes an attack on some parameter sets

  $\hookrightarrow$ SÉTA: **S**upersingular **E**ncryption from **T**orsion **A**ttacks (2019)

  De Feo, de Saint Guilhem, Fouotsa, Kutas, Leroux, Petit, Silva, and Wesolowski
- New attacks on SIDH (2022)

  $\hookrightarrow$ FESTA: **F**ast **E**ncryption from **S**upersingular **T**orsion **A**ttacks

SIDH Attacks in a Nutshell

Given $(\varphi(P), \varphi(Q))$ under a secret isogeny $\varphi\colon E \to E'$ of degree d, where $\langle P, Q\rangle = E[n]$, and n is sufficiently large, it is possible to recover φ.

FESTA = Party in Italian

Trapdoor Function

Triple of algorithms $(\mathsf{KeyGen}, f, f^{-1})$

- $\mathsf{KeyGen}(\lambda) \xrightarrow{\$} (\mathsf{sk}, \mathsf{pk})$
- $f(\mathsf{pk}, x) \to y$
- $f^{-1}(\mathsf{sk}, y) \to x$

Trapdoor Function

Triple of algorithms $(\mathsf{KeyGen}, f, f^{-1})$

- $\mathsf{KeyGen}(\lambda) \xrightarrow{\$} (\mathsf{sk}, \mathsf{pk})$
- $f(\mathsf{pk}, x) \rightarrow y$
- $f^{-1}(\mathsf{sk}, y) \rightarrow x$

Correct
For all pk and x, $f^{-1}(\mathsf{sk}, f(\mathsf{pk}, x)) = x$.

Trapdoor Function

Triple of algorithms $(\mathsf{KeyGen}, f, f^{-1})$

- $\mathsf{KeyGen}(\lambda) \xrightarrow{\$} (\mathsf{sk}, \mathsf{pk})$
- $f(\mathsf{pk}, x) \to y$
- $f^{-1}(\mathsf{sk}, y) \to x$

Correct
For all pk and x, $f^{-1}(\mathsf{sk}, f(\mathsf{pk}, x)) = x$.

One-way
Given pk and y, finding x st $f(\mathsf{pk}, x) = y$ is hard.

FESTA Trapdoor

$$\begin{pmatrix} P_b \\ Q_b \end{pmatrix}$$

$$E_0$$

FESTA Trapdoor

$$\begin{pmatrix} P_b \\ Q_b \end{pmatrix}$$

$$E_0 \xrightarrow{\varphi_A} E_A$$

$\mathsf{sk} = (\alpha, \varphi_A)$

FESTA Trapdoor

$$\begin{tikzcd} E_0 \arrow[rr, "\varphi_A"] & & E_A \end{tikzcd}$$

$\begin{pmatrix} P_b \\ Q_b \end{pmatrix}$

$\begin{pmatrix} R_A \\ S_A \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi_A(P_b) \\ [\alpha^{-1}]\varphi_A(Q_b) \end{pmatrix}$

$\mathsf{sk} = (\alpha, \varphi_A)$

FESTA Trapdoor

$$\begin{tikzcd} \begin{pmatrix} P_b \\ Q_b \end{pmatrix} E_0 \arrow[r, "\varphi_A"] & E_A \begin{pmatrix} R_A \\ S_A \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi_A(P_b) \\ [\alpha^{-1}]\varphi_A(Q_b) \end{pmatrix} \end{tikzcd}$$

$\mathsf{sk} = (\alpha, \varphi_A)$

$\mathsf{pk} = (E_A, R_A, S_A)$

FESTA Trapdoor

$$
\begin{tikzcd}
\begin{pmatrix} P_b \\ Q_b \end{pmatrix} E_0 \arrow[rr, "\varphi_A"] \arrow[d, "\varphi_1"'] & & E_A \begin{pmatrix} R_A \\ S_A \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi_A(P_b) \\ [\alpha^{-1}]\varphi_A(Q_b) \end{pmatrix} \arrow[d, "\varphi_2"] \\
E_1 & & E_2
\end{tikzcd}
$$

$$
\begin{gathered}
\mathsf{sk} = (\alpha, \varphi_A) \\
\mathsf{pk} = (E_A, R_A, S_A) \\
x = (\varphi_1, \varphi_2, \beta)
\end{gathered}
$$

FESTA Trapdoor

$$\begin{pmatrix} P_b \\ Q_b \end{pmatrix} \qquad \begin{pmatrix} R_A \\ S_A \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi_A(P_b) \\ [\alpha^{-1}]\varphi_A(Q_b) \end{pmatrix}$$

$$\begin{tikzcd} E_0 \arrow[r, "\varphi_A"] \arrow[d, "\varphi_1"'] & E_A \arrow[d, "\varphi_2"] \\ E_1 & E_2 \end{tikzcd}$$

$$\begin{pmatrix} R_1 \\ S_1 \end{pmatrix} = \begin{pmatrix} [\beta]\varphi_1(P_b) \\ [\beta^{-1}]\varphi_1(Q_b) \end{pmatrix}$$

$$\mathsf{sk} = (\alpha, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \beta)$$

FESTA Trapdoor

$$\begin{tikzcd}
\begin{pmatrix} P_b \\ Q_b \end{pmatrix} E_0 \arrow[r, "\varphi_A"] \arrow[d, "\varphi_1"'] & E_A \begin{pmatrix} R_A \\ S_A \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi_A(P_b) \\ [\alpha^{-1}]\varphi_A(Q_b) \end{pmatrix} \arrow[d, "\varphi_2"] \\
\begin{pmatrix} R_1 \\ S_1 \end{pmatrix} = \begin{pmatrix} [\beta]\varphi_1(P_b) \\ [\beta^{-1}]\varphi_1(Q_b) \end{pmatrix} E_1 & E_2 \begin{pmatrix} R_2 \\ S_2 \end{pmatrix} = \begin{pmatrix} [\beta]\varphi_2(R_A) \\ [\beta^{-1}]\varphi_2(S_A) \end{pmatrix}
\end{tikzcd}$$

$$\mathsf{sk} = (\alpha, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \beta)$$

$$\begin{pmatrix}P_b\\ Q_b\end{pmatrix} \qquad \begin{pmatrix}R_A\\ S_A\end{pmatrix} = \begin{pmatrix}[\alpha]\varphi_A(P_b)\\ [\alpha^{-1}]\varphi_A(Q_b)\end{pmatrix}$$

$$\begin{tikzcd}
E_0 \arrow[r, "\varphi_A"] \arrow[d, "\varphi_1"'] & E_A \arrow[d, "\varphi_2"] \\
E_1 & E_2
\end{tikzcd}$$

$$\begin{pmatrix}R_1\\ S_1\end{pmatrix} = \begin{pmatrix}[\beta]\varphi_1(P_b)\\ [\beta^{-1}]\varphi_1(Q_b)\end{pmatrix} \qquad \begin{pmatrix}R_2\\ S_2\end{pmatrix} = \begin{pmatrix}[\beta]\varphi_2(R_A)\\ [\beta^{-1}]\varphi_2(S_A)\end{pmatrix}$$

$$\mathsf{sk} = (\alpha, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \beta)$$
$$y = (E_1, R_1, S_1, E_2, R_2, S_2)$$

FESTA Trapdoor

$$\begin{pmatrix} P_b \\ Q_b \end{pmatrix} \qquad \begin{pmatrix} R_A \\ S_A \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi_A(P_b) \\ [\alpha^{-1}]\varphi_A(Q_b) \end{pmatrix}$$

$$\begin{tikzcd}
E_0 \arrow[r, "\varphi_A"] \arrow[d, "\varphi_1"'] & E_A \arrow[d, "\varphi_2"] \\
E_1 \arrow[r, dashed, bend left, "\psi = \varphi_2 \circ \varphi_A \circ \hat{\varphi}_1"] & E_2
\end{tikzcd}$$

$$\begin{pmatrix} R_1 \\ S_1 \end{pmatrix} = \begin{pmatrix} [\beta]\varphi_1(P_b) \\ [\beta^{-1}]\varphi_1(Q_b) \end{pmatrix} \qquad \begin{pmatrix} R_2 \\ S_2 \end{pmatrix} = \begin{pmatrix} [\beta]\varphi_2(R_A) \\ [\beta^{-1}]\varphi_2(S_A) \end{pmatrix}$$

$\mathsf{sk} = (\alpha, \varphi_A)$

$\mathsf{pk} = (E_A, R_A, S_A)$

$x = (\varphi_1, \varphi_2, \beta)$

$y = (E_1, R_1, S_1, E_2, R_2, S_2)$

$$\begin{pmatrix} P_b \\ Q_b \end{pmatrix} \qquad \begin{pmatrix} R_A \\ S_A \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi_A(P_b) \\ [\alpha^{-1}]\varphi_A(Q_b) \end{pmatrix}$$

$$\begin{tikzcd}
E_0 \arrow[r, "\varphi_A"] \arrow[d, "\varphi_1"'] & E_A \arrow[d, "\varphi_2"] \\
E_1 \arrow[r, dashed, "\psi = \varphi_2 \circ \varphi_A \circ \hat{\varphi}_1"'] & E_2
\end{tikzcd}$$

$$\mathsf{sk} = (\alpha, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \beta)$$
$$y = (E_1, R_1, S_1, E_2, R_2, S_2)$$

$$\begin{pmatrix} R_1 \\ S_1 \end{pmatrix} = \begin{pmatrix} [\beta]\varphi_1(P_b) \\ [\beta^{-1}]\varphi_1(Q_b) \end{pmatrix} \qquad \begin{pmatrix} R_2 \\ S_2 \end{pmatrix} = \begin{pmatrix} [\beta]\varphi_2(R_A) \\ [\beta^{-1}]\varphi_2(S_A) \end{pmatrix}$$

$$\begin{pmatrix} \psi(R_1) \\ \psi(S_1) \end{pmatrix} = \deg(\varphi_1) \begin{pmatrix} [\alpha^{-1}]R_2 \\ [\alpha]S_2 \end{pmatrix}$$

- $\varphi_{N_1} : E_0 \to E_1$, $\varphi_{N_2} : E_0 \to E_2$ s.t. $\gcd(N_1, N_2) = 1$
- $K := \langle([N_2]\varphi_{N_1}(P), [N_1]\varphi_{N_2}(P)), ([N_2]\varphi_{N_1}(Q), [N_1]\varphi_{N_2}(Q))\rangle$, where $\langle P, Q\rangle = E_0[N_1 + N_2]$

Theorem

The $(N_1 + N_2, N_1 + N_2)$-polarised isogeny Φ with kernel K has matrix form

$$\begin{pmatrix} \widehat{\varphi}_{N_1} & -\widehat{\varphi}_{N_2} \\ g_{N_2} & \widehat{g}_{N_1} \end{pmatrix},$$

where g_{N_i} are N_i-isogenies such that $\varphi_{N_2} \circ \widehat{\varphi}_{N_1} = g_{N_1} \circ g_{N_2}$.

Concrete Inversion

$$
\begin{tikzcd}
 & E_0 \arrow[r, "\varphi_{A,1}"] \arrow[rr, bend left, "\varphi_A"] \arrow[d, "\varphi_1"'] & \tilde{E}_A \arrow[r, "\varphi_{A,2}"] & E_A \arrow[d, "\varphi_2"] & \\
\begin{pmatrix} R_1 \\ S_1 \end{pmatrix} & E_1 \arrow[rr, dashed, bend left, "\psi = \varphi_2 \circ \varphi_A \circ \hat{\varphi}_1"] & & E_2 & \begin{pmatrix} R_2 \\ S_2 \end{pmatrix}
\end{tikzcd}
$$

Concrete Inversion

$$\begin{tikzcd}
 & E_0 \arrow[r, "\varphi_{A,1}"] \arrow[rr, bend left, "\varphi_A"] \arrow[d, "\varphi_1"'] & \tilde{E}_A \arrow[r, "\varphi_{A,2}"] & E_A \arrow[d, "\varphi_2"] & \\
\binom{R_1}{S_1} & E_1 \arrow[rr, dashed, "\psi = \varphi_2 \circ \varphi_A \circ \hat{\varphi}_1"'] & & E_2 & \binom{R_2}{S_2}
\end{tikzcd}$$

- $d_{A,i} = \deg(\varphi_{A,i})$, $d_i = \deg(\varphi_i)$
- $m_1^2 d_{A,1} d_1 + m_2^2 d_{A,2} d_2 = 2^b$
- $\varphi_{N_1} = [m_1] \circ \varphi_1 \circ \widehat{\varphi}_{A,1}$,
  $\varphi_{N_2} = [m_2] \circ \varphi_2 \circ \varphi_{A,2}$
- $d_i > 2^{2\lambda}$, $d_{A,1} \cdot d_{A,2} > 2^{2\lambda}$

$$
\begin{tikzcd}
 & E_0 \arrow[r, "\varphi_{A,1}"] \arrow[rr, bend left, "\varphi_A"] \arrow[dl, "\varphi_1"'] & \tilde{E}_A \arrow[r, "\varphi_{A,2}"] & E_A \arrow[dr, "\varphi_2"] & \\
\left(\begin{smallmatrix} R_1 \\ S_1 \end{smallmatrix}\right) \; E_1 \arrow[rrrr, dashed, "\psi = \varphi_2 \circ \varphi_A \circ \hat{\varphi}_1"] & & & & E_2 \; \left(\begin{smallmatrix} R_2 \\ S_2 \end{smallmatrix}\right)
\end{tikzcd}
$$

- $d_{A,i} = \deg(\varphi_{A,i})$, $d_i = \deg(\varphi_i)$
- $m_1^2 d_{A,1} d_1 + m_2^2 d_{A,2} d_2 = 2^b$
- $\varphi_{N_1} = [m_1] \circ \varphi_1 \circ \widehat{\varphi}_{A,1}$,
  $\varphi_{N_2} = [m_2] \circ \varphi_2 \circ \varphi_{A,2}$
- $d_i > 2^{2\lambda}$, $d_{A,1} \cdot d_{A,2} > 2^{2\lambda}$

$$
K := \left\langle \begin{array}{r} ([m_2 d_{A,2} d_2]R_1, [d_1 m_1 \alpha^{-1}]R_2), \\ ([m_2 d_{A,2} d_2]S_1, [d_1 m_1 \alpha]S_2) \end{array} \right\rangle,
$$

Concrete Inversion

$$
\begin{tikzcd}
 & E_0 \arrow[r, "\varphi_{A,1}"] \arrow[d, "\varphi_1"'] \arrow[rr, bend left, "\varphi_A"] & \tilde{E}_A \arrow[r, "\varphi_{A,2}"] & E_A \arrow[d, "\varphi_2"] & \\
\begin{pmatrix} R_1 \\ S_1 \end{pmatrix} & E_1 \arrow[rr, dashed, bend left, "\psi = \varphi_2 \circ \varphi_A \circ \hat{\varphi}_1"] & & E_2 & \begin{pmatrix} R_2 \\ S_2 \end{pmatrix}
\end{tikzcd}
$$

- $d_{A,i} = \deg(\varphi_{A,i})$, $d_i = \deg(\varphi_i)$
- $m_1^2 d_{A,1} d_1 + m_2^2 d_{A,2} d_2 = 2^b$
- $\varphi_{N_1} = [m_1] \circ \varphi_1 \circ \widehat{\varphi}_{A,1}$,
  $\varphi_{N_2} = [m_2] \circ \varphi_2 \circ \varphi_{A,2}$
- $d_i > 2^{2\lambda}$, $d_{A,1} \cdot d_{A,2} > 2^{2\lambda}$

$$K := \left\langle \begin{array}{r} ([m_2 d_{A,2} d_2]R_1, [d_1 m_1 \alpha^{-1}]R_2), \\ ([m_2 d_{A,2} d_2]S_1, [d_1 m_1 \alpha]S_2) \end{array} \right\rangle,$$

$$\Phi = \begin{pmatrix} [m_1] \circ \varphi_{A,1} \circ \widehat{\varphi}_1 & -[m_2] \circ \widehat{\varphi}_{A,2} \circ \widehat{\varphi}_2 \\ [m_2] \circ g_{d_2 d_{A,2}} & [m_1] \circ \widehat{g}_{d_{A,1} d_1} \end{pmatrix}$$

Outline

Guessing some torsion

Let $\varphi : E_0 \to E$, and $\begin{pmatrix} R \\ S \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi(P) \\ [\alpha^{-1}]\varphi(Q) \end{pmatrix}$

Guessing some torsion

Let $\varphi : E_0 \to E$, and $\begin{pmatrix} R \\ S \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi(P) \\ [\alpha^{-1}]\varphi(Q) \end{pmatrix}$

Let n such that $2^n \geq 2\sqrt{\deg(\varphi)}$.

The idea is to guess $\alpha \pmod{2^n}$ and apply Robert's attack in dimension 4 (or 8).

Guessing some torsion

Let $\varphi : E_0 \to E$, and $\begin{pmatrix} R \\ S \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi(P) \\ [\alpha^{-1}]\varphi(Q) \end{pmatrix}$

Let n such that $2^n \geq 2\sqrt{\deg(\varphi)}$.

The idea is to guess $\alpha \pmod{2^n}$ and apply Robert's attack in dimension 4 (or 8).

$$\begin{pmatrix} R' \\ S' \end{pmatrix} = [2^{n-b}] \begin{pmatrix} R \\ S \end{pmatrix}$$

We need to find a $\beta \leq 2^n$ st

$$[2^{b-n}] \begin{pmatrix} \varphi(P) \\ \varphi(Q) \end{pmatrix} = \begin{pmatrix} [\beta]R' \\ [\beta^{-1}]S' \end{pmatrix}$$

Guessing some torsion

Let $\varphi : E_0 \to E$, and $\begin{pmatrix} R \\ S \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi(P) \\ [\alpha^{-1}]\varphi(Q) \end{pmatrix}$

Let n such that $2^n \geq 2\sqrt{\deg(\varphi)}$.

The idea is to guess $\alpha \pmod{2^n}$ and apply Robert's attack in dimension 4 (or 8).

$$\begin{pmatrix} R' \\ S' \end{pmatrix} = [2^{n-b}] \begin{pmatrix} R \\ S \end{pmatrix}$$

We need to find a $\beta \leq 2^n$ st

$$[2^{b-n}] \begin{pmatrix} \varphi(P) \\ \varphi(Q) \end{pmatrix} = \begin{pmatrix} [\beta]R' \\ [\beta^{-1}]S' \end{pmatrix}$$

Remark

Assuming the cost of running Robert's attack is negligible, $\deg(\varphi) > 2^{2\lambda}$.

FESTA-128

$$m_1^2 d_{A,1} d_1 + m_2^2 d_{A,2} d_2 = 2^b$$

$$\begin{aligned}
b &:= 632,\\
d_1 &:= (3^3 \cdot 19 \cdot 29 \cdot 37 \cdot 83 \cdot 139 \cdot 167 \cdot 251 \cdot 419 \cdot 421 \cdot 701 \cdot 839 \cdot 1009 \cdot\\
&\quad 1259 \cdot 3061 \cdot 3779)^2,\\
d_2 &:= 7 \cdot (5^2 \cdot 7 \cdot 11 \cdot 13 \cdot 17 \cdot 41 \cdot 43 \cdot 71 \cdot 89 \cdot 127 \cdot 211 \cdot 281 \cdot 503 \cdot 631 \cdot\\
&\quad 2309 \cdot 2521 \cdot 2647 \cdot 2729)^2,\\
d_{A,1} &:= (59 \cdot 6299 \cdot 6719 \cdot 9181)^2,\\
d_{A,2} &:= (3023 \cdot 3359 \cdot 4409 \cdot 5039 \cdot 19531 \cdot 22679 \cdot 41161)^2,\\
m_1 &:= 14921849450934765925202420839250441821039 21,\\
m_2 &:= 2561733133642993930016669306 9,\\
f &:= 107.
\end{aligned}$$

$p = 2^b d_1 (d_{A,1} d_{A,2})_{\mathsf{sf}} d_2 f - 1$ is 1292-bit long

$$\begin{tikzcd}
E_0 \arrow[r, "\varphi_{A,1}"] \arrow[rr, bend left, "\varphi_A"] \arrow[d, "\varphi_1"'] & \tilde{E}_A \arrow[r, "\varphi_{A,2}"] & E_A \arrow[d, "\varphi_2"] \\
E_1 & & E_2
\end{tikzcd}$$

Generalised Lollipop Attack (Castryck-Vercauteren 2023)

Let $\varphi : E_0 \to E$, and $\begin{pmatrix} R \\ S \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi(P) \\ [\alpha^{-1}]\varphi(Q) \end{pmatrix}$

$$\begin{tikzcd} E_0 \arrow[loop above, "\omega"] \arrow[d, shift left, "\varphi"] \\ E \arrow[u, shift left, "\widehat{\varphi}"] \end{tikzcd}$$

Generalised Lollipop Attack (Castryck-Vercauteren 2023)

Let $\varphi : E_0 \to E$, and $\begin{pmatrix} R \\ S \end{pmatrix} = \begin{pmatrix} [\alpha]\varphi(P) \\ [\alpha^{-1}]\varphi(Q) \end{pmatrix}$

$$\begin{tikzcd} & E_0 \arrow[loop above, "\omega"] \arrow[dl, "\varphi"'] \arrow[dr, "\varphi^{(p)}"] & \\ E & & E^{(p)} \end{tikzcd}$$

- $\psi := \varphi^{(p)} \circ \omega \circ \hat{\varphi}$
- **If** $\begin{pmatrix} \pi \circ \omega(P) \\ \pi \circ \omega(Q) \end{pmatrix} = \begin{pmatrix} [\beta_P]P \\ [\beta_Q]Q \end{pmatrix}$,

$$\begin{pmatrix} \psi(R) \\ \psi(S) \end{pmatrix} = [\deg \varphi] \begin{pmatrix} [\beta_P]\pi(R) \\ [\beta_Q]\pi(S) \end{pmatrix}$$

- Case of $E_0 : y^2 = x^3 + 6x^2 + x$ (no need for this)
- The 2^b-torsion basis is computed as in (Zanon, Simplicio Jr, Pereira, Doliskani, and Barreto, 2017)
  - Full basis scenario $\deg(\omega) < 2^{2b} / \deg(\varphi)^2$: $\mathcal{O}(\min\{2^{-4\lambda}, 2^{-b}\})$
  - Image of a single point $\deg(\omega) < 2^b / \deg(\varphi)$: $\mathcal{O}(2^{-4\lambda})$
- Polynomial-time attack if $2^b \gtrapprox pd^3$

Outline

Fundamental Facts

- On an elliptic curve E, we implicitly have some additional structure
- By Riemann–Roch, the following

$$\begin{aligned} E &\xrightarrow{\sim} \mathrm{Pic}^0(E) \\ P &\mapsto (P) - (0_E) \end{aligned}$$

  is an isomorphism between the elliptic curve and its dual.
- In practice, this isomorphism is described by a *line bundle* $\mathcal{L}((0_E))$.

Fundamental Facts

- On an elliptic curve E, we implicitly have some additional structure
- By Riemann–Roch, the following

$$\begin{aligned} E &\xrightarrow{\sim} \mathrm{Pic}^0(E) \\ P &\mapsto (P) - (0_E) \end{aligned}$$

  is an isomorphism between the elliptic curve and its dual.
- In practice, this isomorphism is described by a *line bundle* $\mathcal{L}((0_E))$.
- We have just described a (principal) *polarisation*!

Fundamental Facts

- On an elliptic curve E, we implicitly have some additional structure
- By Riemann–Roch, the following

$$\begin{aligned} E &\xrightarrow{\sim} \mathrm{Pic}^0(E) \\ P &\mapsto (P) - (0_E) \end{aligned}$$

  is an isomorphism between the elliptic curve and its dual.
- In practice, this isomorphism is described by a *line bundle* $\mathcal{L}((0_E))$.
- We have just described a (principal) *polarisation*!

Generalisation

The correct generalisation to higher dimension is principally polarised abelian varieties.

Fundamental Facts

- Let A be an abelian variety. A *polarisation* is an isogeny $\xi : A \to \hat{A}$ coming from a line bundle $\mathcal{L}_A$. A *pricipal polarisation* is a polarisation that is also an isomorphism.
- Let (A, ξ_A) and (B, ξ_B) be two principally polarised abelian varieties. The isogeny $f : A \to B$ is a N-isogeny if $[N] \circ \xi_A = \hat{f} \circ \xi_B \circ f$.

Fundamental Facts

- Let A be an abelian variety. A *polarisation* is an isogeny $\xi : A \to \hat{A}$ coming from a line bundle $\mathcal{L}_A$. A *pricipal polarisation* is a polarisation that is also an isomorphism.
- Let (A, ξ_A) and (B, ξ_B) be two principally polarised abelian varieties. The isogeny $f : A \to B$ is a N-isogeny if $[N] \circ \xi_A = \hat{f} \circ \xi_B \circ f$.
- In dimension 2, we have only two types of principally polarised abelian surfaces:
  - products of elliptic curves
  - Jacobians of hyperelliptic genus-2 curves
- We note the N-isogenies in dimension 2 by (N, N).

Chains of $(2,2)$-isogenies between elliptic products

Goal: Compute the $(2^b, 2^b)$-isogeny $f : E_1 \times E_2 \to E_1' \times E_2'$

Chains of $(2, 2)$-isogenies between elliptic products

Goal: Compute the $(2^b, 2^b)$-isogeny $f : E_1 \times E_2 \to E_1' \times E_2'$
We compute f as a chain of $(2, 2)$-isogenies:

$$f = f_b \circ \ldots \circ f_1$$

Goal: Compute the $(2^b, 2^b)$-isogeny $f : E_1 \times E_2 \to E_1' \times E_2'$
We compute f as a chain of $(2,2)$-isogenies:

$$f = f_b \circ \ldots \circ f_1$$

The state of the art:

- Gluing isogeny $f_1 : E_1 \times E_2 \to \mathsf{Jac}(\mathcal{C})$ (Howe, Leprévost, and Poonen, 2000)
- Splitting Isogeny $f_b : \mathsf{Jac}(\mathcal{C}) \to E_1' \times E_2'$ (Smith, 2005)
- Richelot Isogenies $f_i : \mathsf{Jac}(\mathcal{C}_i) \to \mathsf{Jac}(\mathcal{C}_{i+1})$, for $i = 2, \ldots, b-1$ (Smith, 2005)

Theta Structures

Let A be a principally polarised abelian surface, e.g. a product of two elliptic curves.
Let $A[4] = \langle S_1', S_2', T_1', T_2' \rangle$ be a symplectic 4-torsion basis

- $e(S_1', T_1') = e(S_2', T_2') = \mu$
- $e(S_1', S_2') = e(T_1', T_2') = e(S_1', T_2') = e(S_2', T_1') = 1$

Theta Structures

Let A be a principally polarised abelian surface, e.g. a product of two elliptic curves.
Let $A[4] = \langle S_1', S_2', T_1', T_2' \rangle$ be a symplectic 4-torsion basis

- $e(S_1', T_1') = e(S_2', T_2') = \mu$
- $e(S_1', S_2') = e(T_1', T_2') = e(S_1', T_2') = e(S_2', T_1') = 1$

$\langle S_1', S_2', T_1', T_2' \rangle \rightsquigarrow \theta_{00}, \theta_{10}, \theta_{01}, \theta_{11}$

$$P \in A \to (\theta_{00}(P) : \theta_{10}(P) : \theta_{01}(P) : \theta_{11}(P)) \in \mathbb{P}^3$$

The projective point $(\theta_{00}(0) : \theta_{10}(0) : \theta_{01}(0) : \theta_{11}(0))$ is enough to describe A.

An example – Elliptic Products

In S3×E3, Sarkis gave conversion formulae between Montgomery curves and its theta squared model.
Picking any square root of the squared theta-null coordinates, we obtain a theta structure on the elliptic curve.

$$E \rightsquigarrow (a : b) \in \mathbb{P}^2$$

An example – Elliptic Products

In S3×E3, Sarkis gave conversion formulae between Montgomery curves and its theta squared model.
Picking any square root of the squared theta-null coordinates, we obtain a theta structure on the elliptic curve.

$$E \rightsquigarrow (a : b) \in \mathbb{P}^2$$

$$P = (X : Z) \mapsto (\theta_0(P) : \theta_1(P)) = (a(X - Z) : b(X + Z))$$

An example – Elliptic Products

In S3×E3, Sarkis gave conversion formulae between Montgomery curves and its theta squared model.
Picking any square root of the squared theta-null coordinates, we obtain a theta structure on the elliptic curve.

$$E \rightsquigarrow (a : b) \in \mathbb{P}^2$$

$$P = (X : Z) \mapsto (\theta_0(P) : \theta_1(P)) = (a(X - Z) : b(X + Z))$$

Product theta structure on $E_1 \times E_2$

$$(P_1, P_2) \in E_1 \times E_2 \mapsto \\ (\theta_0^{E_1}(P_1)\theta_0^{E_2}(P_2) : \theta_1^{E_1}(P_1)\theta_0^{E_2}(P_2) : \theta_0^{E_1}(P_1)\theta_1^{E_2}(P_2) : \theta_1^{E_1}(P_1)\theta_1^{E_2}(P_2))$$

Some Operators

The Hadamard transform

$$\mathcal{H}(x,y,z,w) := \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix}$$

We define $(\tilde{\theta}_{00}(P) : \tilde{\theta}_{10}(P) : \tilde{\theta}_{01}(P) : \tilde{\theta}_{11}(P)) = \mathcal{H}\left(\theta_{00}(P), \theta_{10}(P), \theta_{01}(P), \theta_{11}(P)\right)$ to be the *dual coordinates* of P.
Also $\mathcal{H} \circ \mathcal{H}(x,y,z,w) = (x,y,z,w)$.

Some Operators

The Hadamard transform

$$\mathcal{H}(x,y,z,w) := \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix}$$

We define $(\tilde{\theta}_{00}(P) : \tilde{\theta}_{10}(P) : \tilde{\theta}_{01}(P) : \tilde{\theta}_{11}(P)) = \mathcal{H}\left(\theta_{00}(P), \theta_{10}(P), \theta_{01}(P), \theta_{11}(P)\right)$ to be the *dual coordinates* of P.
Also $\mathcal{H} \circ \mathcal{H}(x,y,z,w) = (x,y,z,w)$.
The squaring operator

$$\mathcal{S}(x,y,z,w) := (x^2, y^2, z^2, w^2)$$

Some Operators

The Hadamard transform

$$\mathcal{H}(x,y,z,w) := \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix}$$

We define $(\tilde{\theta}_{00}(P) : \tilde{\theta}_{10}(P) : \tilde{\theta}_{01}(P) : \tilde{\theta}_{11}(P)) = \mathcal{H}\left(\theta_{00}(P), \theta_{10}(P), \theta_{01}(P), \theta_{11}(P)\right)$ to be the *dual coordinates* of P.

Also $\mathcal{H} \circ \mathcal{H}(x,y,z,w) = (x,y,z,w)$.

The squaring operator

$$\mathcal{S}(x,y,z,w) := (x^2, y^2, z^2, w^2)$$

The $\star$ operator

$$(x,y,z,w) \star (x',y',z',w') = (xx', yy', zz', ww')$$

Duplication Formula

Let $f : A \to B$, $\ker f = \langle T_1, T_2 \rangle$, where $T_i = [2]T_i'$. (Remember the decomposition $\langle S_1', S_2', T_1', T_2' \rangle$)

$$\left(\theta_i^A(P+Q)\right)_i \star \left(\theta_i^A(P-Q)\right)_i = \mathcal{H}\left(\left(\tilde{\theta}_i^B(f(P))\right)_i \star \left(\tilde{\theta}_i^B(f(Q))\right)_i\right).$$

We can obtain addition formulae

- Differential addition: $8\mathbf{S} + 17\mathbf{M}$
- Doubling: $8\mathbf{S} + 6\mathbf{M}$

The same formulae as in (Gaudry, 2005)

The isogeny formula

Goal: To compute the isogeny $f : A \to B$ with $\ker f = \langle T_1, T_2 \rangle$, where $T_i = [2]T_i'$.

The isogeny formula

Goal: To compute the isogeny $f : A \to B$ with $\ker f = \langle T_1, T_2 \rangle$, where $T_i = [2]T_i'$.
Assume that we have an isotropic group $\langle T_1'', T_2'' \rangle$ such that $T_i' = [2]T_i''$.

The isogeny formula

Goal: To compute the isogeny $f : A \to B$ with $\ker f = \langle T_1, T_2 \rangle$, where $T_i = [2]T_i'$.
Assume that we have an isotropic group $\langle T_1'', T_2'' \rangle$ such that $T_i' = [2]T_i''$.
Define $(\alpha : \beta : \gamma : \delta) = (\tilde{\theta}_{00}^B(0) : \tilde{\theta}_{10}^B(0) : \tilde{\theta}_{01}^B(0) : \tilde{\theta}_{11}^B(0))$

The isogeny formula

Goal: To compute the isogeny $f : A \to B$ with $\ker f = \langle T_1, T_2 \rangle$, where $T_i = [2]T_i'$.
Assume that we have an isotropic group $\langle T_1'', T_2'' \rangle$ such that $T_i' = [2]T_i''$.
Define $(\alpha : \beta : \gamma : \delta) = (\tilde{\theta}^B_{00}(0) : \tilde{\theta}^B_{10}(0) : \tilde{\theta}^B_{01}(0) : \tilde{\theta}^B_{11}(0))$
Trust me, we did the maths:

$$\mathcal{H} \circ \mathcal{S}(\theta^A_{00}(T_1''), \theta^A_{10}(T_1''), \theta^A_{01}(T_1''), \theta^A_{11}(T_1'')) = (x\alpha, x\beta, y\gamma, y\delta),$$
$$\mathcal{H} \circ \mathcal{S}(\theta^A_{00}(T_2''), \theta^A_{10}(T_2''), \theta^A_{01}(T_2''), \theta^A_{11}(T_2'')) = (z\alpha, w\beta, z\gamma, w\delta),$$

for some unknown x, y, z, w.

Goal: To compute the isogeny $f : A \to B$ with $\ker f = \langle T_1, T_2 \rangle$, where $T_i = [2]T_i'$.
Assume that we have an isotropic group $\langle T_1'', T_2'' \rangle$ such that $T_i' = [2]T_i''$.
Define $(\alpha : \beta : \gamma : \delta) = (\tilde{\theta}_{00}^B(0) : \tilde{\theta}_{10}^B(0) : \tilde{\theta}_{01}^B(0) : \tilde{\theta}_{11}^B(0))$
Trust me, we did the maths:

$$\mathcal{H} \circ \mathcal{S}(\theta_{00}^A(T_1''), \theta_{10}^A(T_1''), \theta_{01}^A(T_1''), \theta_{11}^A(T_1'')) = (x\alpha, x\beta, y\gamma, y\delta),$$
$$\mathcal{H} \circ \mathcal{S}(\theta_{00}^A(T_2''), \theta_{10}^A(T_2''), \theta_{01}^A(T_2''), \theta_{11}^A(T_2'')) = (z\alpha, w\beta, z\gamma, w\delta),$$

for some unknown x, y, z, w.
Hence, we can recover the dual theta-null point $(\alpha : \beta : \gamma : \delta)$ for B, and in turn the theta-null point $\mathcal{H}(\alpha : \beta : \gamma : \delta)$ on B.

We can also evaluate the isogeny f at any point P:

$$\begin{aligned}(\tilde{\theta}^B_{00}(f(P)), \tilde{\theta}^B_{10}(f(P)), \tilde{\theta}^B_{01}(f(P)), \tilde{\theta}^B_{11}(f(P))) = \\ (\alpha^{-1}, \beta^{-1}, \gamma^{-1}, \delta^{-1}) \star \mathcal{H} \circ \mathcal{S}\left((\theta^A_i(P))_i\right),\end{aligned}$$

from which we can compute

$$\begin{aligned}(\theta^B_{00}(f(P)), \theta^B_{10}(f(P)), \theta^B_{01}(f(P)), \theta^B_{11}(f(P))) = \\ \mathcal{H}(\tilde{\theta}^B_{00}(f(P)), \tilde{\theta}^B_{10}(f(P)), \tilde{\theta}^B_{01}(f(P)), \tilde{\theta}^B_{11}(f(P)))\end{aligned}$$

Operation Counting

Isogeny Type	Doubling	Codomain		Evaluation
		Generic	Optimised	
Generic	$8\mathbf{S} + 6\mathbf{M}$	$8\mathbf{S} + 29\mathbf{M} + 1\mathbf{I}$	$8\mathbf{S} + 9\mathbf{M} + 1\mathbf{I}$	$4\mathbf{S} + 3\mathbf{M}$
Gluing	$12\mathbf{S} + 12\mathbf{M}$	$8\mathbf{S} + 13\mathbf{M} + 1\mathbf{I}$		$8\mathbf{S} + 5\mathbf{M} + 1\mathbf{I}$

Details I skated over

- The formulae I showed you assume that $\ker(f)[4] = \langle T_1', T_2' \rangle$.
- The correction formula requires $100\mathbf{M} + 8\mathbf{S} + 4\mathbf{I}$
- At the end of the chain, we are left with an elliptic product in theta coordinates.
- Switching to the Montgomery model for the two curves is not expensive.

Table 1: Running times of computing the codomain and evaluating a $(2^n, 2^n)$-isogeny between elliptic products over the base field $\mathbb{F}_{p^2}$. Times were recorded on a Intel Core i7-9750H CPU with a clock-speed of 2.6 GHz with turbo-boost disabled.

		Codomain			Evaluation		
$\log p$	n	Theta Rust	Theta SageMath	Richelot SageMath	Theta Rust	Theta SageMath	Richelot SageMath
254	126	**2.85 ms**	108 ms	1028 ms	**161 μs**	5.43 ms	114 ms
381	208	**11.2 ms**	201 ms	1998 ms	**411 μs**	8.68 ms	208 ms
1293	632	**495 ms**	1225 ms	12840 ms	**17.8 ms**	40.8 ms	1203 ms

FESTA-128

Apple M1 PRO CPU clocked at 3.2 GHz using a single performance core

```
================================================================================
                              Running FESTA_128
================================================================================
================================================================================
                         Keygen took: 4.703 seconds
================================================================================
================================================================================
                        Encrypt took: 3.067 seconds
================================================================================
================================================================================
                        Decrypt took: 2.823 seconds
================================================================================
```

Outline

Conlusions

- These $(2,2)$-isogeny formulae can speed-up all the protocols currently relying on chains of $(2,2)$-isogenies, e.g. IS-CUBE (Moriya 2023, S3×E2) and QFESTA (Onuki-Nakagawa, 2023)
- What about (ℓ,ℓ)-isogenies and $(2,2,2,2)$-isogenies?

Thanks for your attention!

Questions?