Practical Encryption from Isogenies between Elliptic Products

A. Basso, L. Maino, and G. Pope

University of Bristol

8th June, 2023

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
  - ↪ 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
  - ↪ 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)
  - ↪ 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
  - ↪ 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)
  - ↪ 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 $\langle P, Q\rangle = E[n]$, it is possible to recover φ.

FESTA = Party in Italian and Basque (and many other languages)

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

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

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

$$\binom{P_b}{Q_b}$$

$$E_0$$

FESTA Trapdoor

$\mathsf{sk} = (\mathbf{A}, \varphi_A)$

FESTA Trapdoor

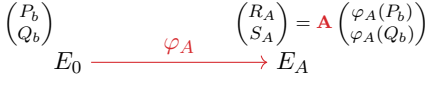
$$\begin{pmatrix} P_b \\ Q_b \end{pmatrix} \qquad \begin{pmatrix} R_A \\ S_A \end{pmatrix} = \mathbf{A} \begin{pmatrix} \varphi_A(P_b) \\ \varphi_A(Q_b) \end{pmatrix}$$

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

$$\boxed{\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}}$$

$\mathsf{sk} = (\mathbf{A}, \varphi_A)$

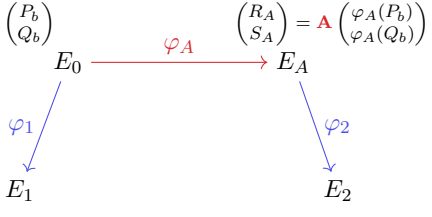
FESTA Trapdoor



$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

$\mathsf{sk} = (\mathbf{A}, \varphi_A)$
$\mathsf{pk} = (E_A, R_A, S_A)$

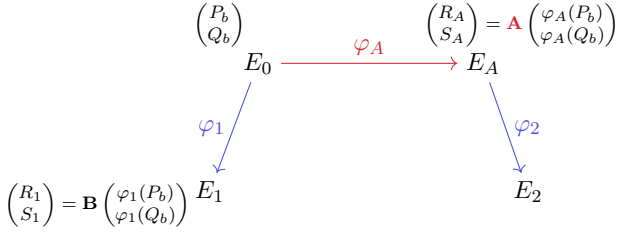
FESTA Trapdoor



$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

$$\mathsf{sk} = (\mathbf{A}, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \mathbf{B})$$

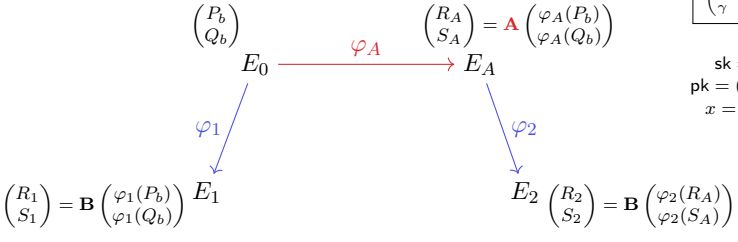
FESTA Trapdoor



$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

$$\mathsf{sk} = (\mathbf{A}, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \mathbf{B})$$

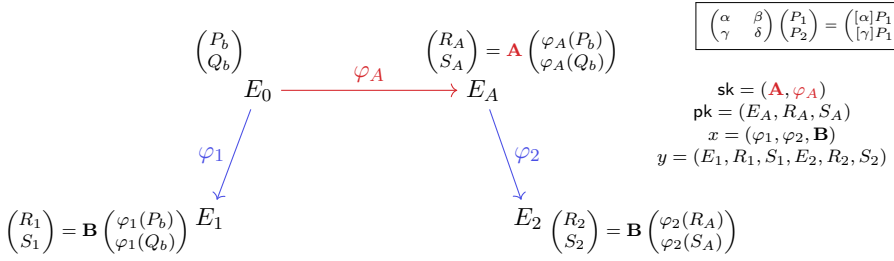
FESTA Trapdoor



$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

$\mathsf{sk} = (\mathbf{A}, \varphi_A)$
$\mathsf{pk} = (E_A, R_A, S_A)$
$x = (\varphi_1, \varphi_2, \mathbf{B})$

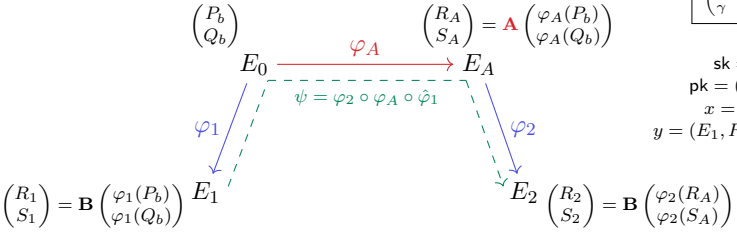
FESTA Trapdoor



$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

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

FESTA Trapdoor

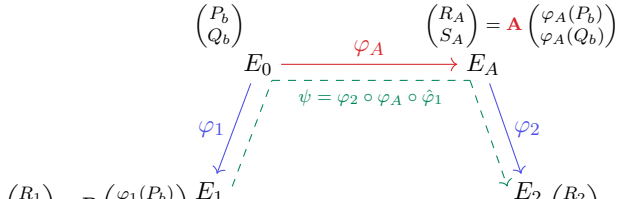


$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

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

FESTA Trapdoor

$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

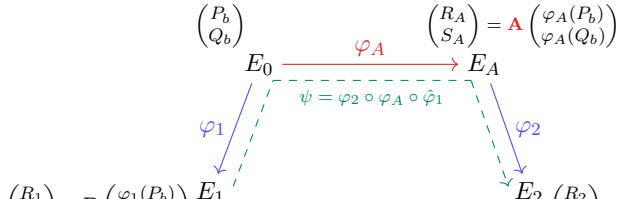


$$\mathsf{sk} = (\mathbf{A}, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \mathbf{B})$$
$$y = (E_1, R_1, S_1, E_2, R_2, S_2)$$

$$\mathbf{B} \cdot \mathbf{A} \cdot \mathbf{B}^{-1} \begin{pmatrix} \psi(R_1) \\ \psi(S_1) \end{pmatrix} = \deg(\varphi_1) \begin{pmatrix} R_2 \\ S_2 \end{pmatrix}$$

FESTA Trapdoor

$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$

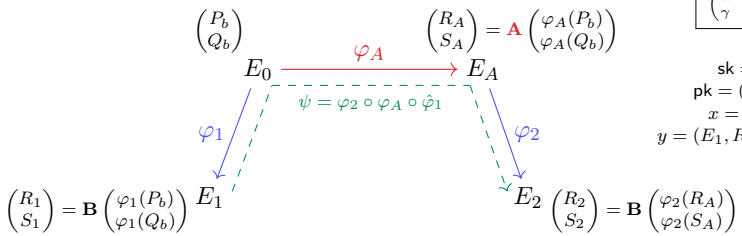


$$\mathsf{sk} = (\mathbf{A}, \varphi_A)$$
$$\mathsf{pk} = (E_A, R_A, S_A)$$
$$x = (\varphi_1, \varphi_2, \mathbf{B})$$
$$y = (E_1, R_1, S_1, E_2, R_2, S_2)$$

$$\mathbf{A} \begin{pmatrix} \psi(R_1) \\ \psi(S_1) \end{pmatrix} = \mathbf{B} \cdot \mathbf{A} \cdot \mathbf{B}^{-1} \begin{pmatrix} \psi(R_1) \\ \psi(S_1) \end{pmatrix} = \deg(\varphi_1) \begin{pmatrix} R_2 \\ S_2 \end{pmatrix}$$

FESTA Trapdoor

$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} [\alpha]P_1 + [\beta]P_2 \\ [\gamma]P_1 + [\delta]P_2 \end{pmatrix}$$



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

$$\mathbf{A} \begin{pmatrix} \psi(R_1) \\ \psi(S_1) \end{pmatrix} = \mathbf{B} \cdot \mathbf{A} \cdot \mathbf{B}^{-1} \begin{pmatrix} \psi(R_1) \\ \psi(S_1) \end{pmatrix} = \deg(\varphi_1) \begin{pmatrix} R_2 \\ S_2 \end{pmatrix}$$

$$\mathbf{A} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \quad \text{or} \quad \mathbf{A} = \begin{pmatrix} \alpha & \beta \\ \beta & \alpha \end{pmatrix}$$

Kani's Criterion

- $\varphi_{N_1} \colon E_0 \to E_1$, $\varphi_{N_2} \colon E_0 \to E_2$ 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 $\langle P, Q\rangle = E_0[N_1 + N_2]$

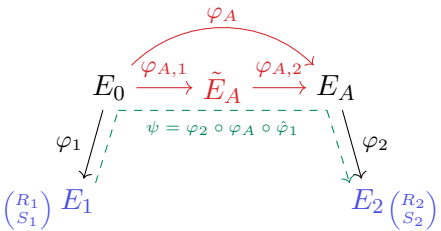
Theorem

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

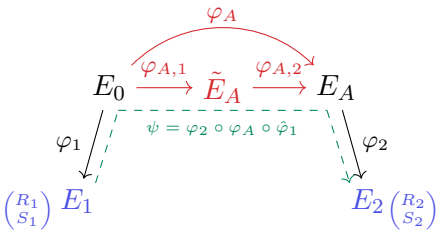
$$\begin{pmatrix} \widehat{\varphi}_{N_1} & -\widehat{\varphi}_{N_2} \\ g_{N_2} & \widehat{g}_{N_1} \end{pmatrix},$$

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

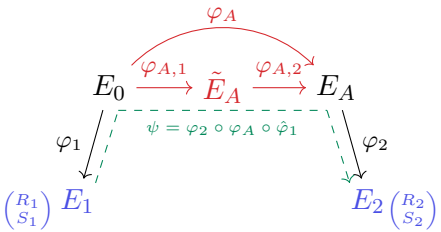


Concrete Inversion



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

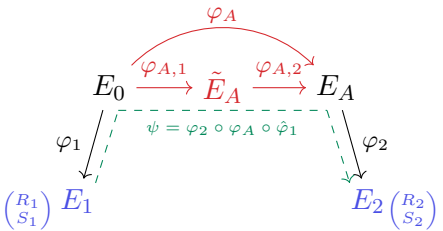
Concrete Inversion



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

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

Concrete Inversion



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

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

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

Proof of Concept

https://github.com/FESTA-PKE/FESTA-SageMath

```
================================================================================
                              Running FESTA_128
================================================================================
================================================================================
                            Keygen took: 4.853 seconds
================================================================================
--------------------------------------------------------------------------------
                         Compressed public key: 561 bytes
--------------------------------------------------------------------------------
================================================================================
                            Encrypt took: 3.513 seconds
================================================================================
--------------------------------------------------------------------------------
                        Compressed ciphertext: 1122 bytes
--------------------------------------------------------------------------------
================================================================================
                            Decrypt took: 10.102 seconds
================================================================================
```

Thanks for your attention!

Questions?