SCALLOP-HD: group action from 2-dimensional isogenies

Mingjie Chen, Antonin Leroux, Lorenz Panny

Université Libre de Bruxelles

April 16, 2024

PKC 2024
Sydney

Group Action

Group Action

Let G be a group and S be a set. A map $\star : G \times S \to S$ is a group action if:

- $e \star s = s$ where $e \in G$ is the identity element and $s \in S$,
- $(gh) \star s = g \star (h \star s)$ where $g, h \in G$ and $s \in S$.

Group Action

Let G be a group and S be a set. A map $\star : G \times S \to S$ is a group action if:

- $e \star s = s$ where $e \in G$ is the identity element and $s \in S$,
- $(gh) \star s = g \star (h \star s)$ where $g, h \in G$ and $s \in S$.

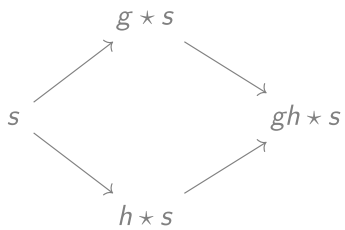
A simple key exchange when G is abelian:

Group Action

Let G be a group and S be a set. A map $\star : G \times S \to S$ is a group action if:

- $e \star s = s$ where $e \in G$ is the identity element and $s \in S$,
- $(gh) \star s = g \star (h \star s)$ where $g, h \in G$ and $s \in S$.

A simple key exchange when G is abelian:



CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

$--$ the first post-quantum NIKE; it has small key size and competitive speed among post-quantum candidates.

CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

—— the first post-quantum NIKE; it has small key size and competitive speed among post-quantum candidates.

Set S

$\{$supersingular elliptic curves $E/\mathbb{F}_p\}$

CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

—— the first post-quantum NIKE; it has small key size and competitive speed among post-quantum candidates.

Set S

{supersingular elliptic curves $E/\mathbb{F}_p$}

Group G

$G = \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ is the ideal class group of the imaginary quadratic order $\mathbb{Z}[\sqrt{-p}]$.

CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

−− the first post-quantum NIKE; it has small key size and competitive speed among post-quantum candidates.

Set S

$\{$supersingular elliptic curves $E/\mathbb{F}_p\}$

Group G

$G = \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ is the ideal class group of the imaginary quadratic order $\mathbb{Z}[\sqrt{-p}]$.

Let $(e_i)_{i=1,\ldots,n} \in [-m, \ldots, m]^n$ (e.g., $m = 5, n = 74$), by the design of CSIDH, one can compute efficiently the action of

CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

−− the first post-quantum NIKE; it has small key size and competitive speed among post-quantum candidates.

Set S

{supersingular elliptic curves $E/\mathbb{F}_p$}

Group G

$G = \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ is the ideal class group of the imaginary quadratic order $\mathbb{Z}[\sqrt{-p}]$.

Let $(e_i)_{i=1,\dots,n} \in [-m, \dots, m]^n$ (e.g., $m = 5, n = 74$), by the design of CSIDH, one can compute efficiently the action of

$$\mathfrak{l}_1^{e_1} \cdot \cdots \cdot \mathfrak{l}_n^{e_n} \text{ on } E.$$

CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

−− the first post-quantum NIKE; it has small key size and competitive speed among post-quantum candidates.

Set S

$\{$supersingular elliptic curves $E/\mathbb{F}_p\}$

Group G

$G = \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ is the ideal class group of the imaginary quadratic order $\mathbb{Z}[\sqrt{-p}]$.

Let $(e_i)_{i=1,\dots,n} \in [-m,\dots,m]^n$ (e.g., $m = 5, n = 74$), by the design of CSIDH, one can compute efficiently the action of

$$\mathfrak{l}_1^{e_1} \cdot \dots \cdot \mathfrak{l}_n^{e_n} \text{ on } E.$$

Here $\{\mathfrak{l}_1,\dots,\mathfrak{l}_n\}$ is the set of prime ideals of small prime norm ℓ_i in $\mathbb{Z}[\sqrt{-p}]$.

CSIDH [Castryck-Lange-Martindale-Panny-Renes 2018]

—— the first post-quantum NIKE; it has small key size and competitive speed among post-quantum candidates.

Set S

$\{$supersingular elliptic curves $E/\mathbb{F}_p\}$

Group G

$G = \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ is the ideal class group of the imaginary quadratic order $\mathbb{Z}[\sqrt{-p}]$.

Let $(e_i)_{i=1,\dots,n} \in [-m,\dots,m]^n$ (e.g., $m=5, n=74$), by the design of CSIDH, one can compute efficiently the action of

$$\mathfrak{l}_1^{e_1} \cdot \dots \cdot \mathfrak{l}_n^{e_n} \text{ on } E.$$

Here $\{\mathfrak{l}_1,\dots,\mathfrak{l}_n\}$ is the set of prime ideals of small prime norm ℓ_i in $\mathbb{Z}[\sqrt{-p}]$.

$\Longrightarrow$ $(e_i)_i$ is the secret key.

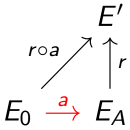
Signatures? I

Signatures? I

Proof of knowledge:

Signatures? I

Proof of knowledge:



Signatures? I

In the context of **CSIDH** group action:

Proof of knowledge:

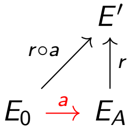
$$\begin{array}{ccc} & & E' \\ & \overset{r \circ a}{\nearrow} & \uparrow r \\ E_0 & \xrightarrow{a} & E_A \end{array}$$

Signatures? I

In the context of **CSIDH** group action:

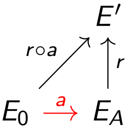
$$a, r \in [-m, m]^n$$

Proof of knowledge:



Signatures? I

Proof of knowledge:



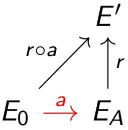
In the context of **CSIDH** group action:

$$a, r \in [-m, m]^n$$

$$r \circ a \in [-2m, 2m]^n$$

Signatures? I

Proof of knowledge:



In the context of CSIDH group action:

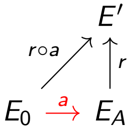
$$a, r \in [-m, m]^n$$

$$r \circ a \in [-2m, 2m]^n$$

! This leaks the secret information !

Signatures? I

Proof of knowledge:



In the context of CSIDH group action:

$$a, r \in [-m, m]^n$$

$$r \circ a \in [-2m, 2m]^n$$

! This leaks the secret information !

— eg: when $r \circ a = [2m, \ldots]$, the adversary knows that $a = [m, \ldots]$.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$. $\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$. $\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 $\longleftarrow$ record breaking.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$. $\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 $\longleftarrow$ record breaking.
- Compute $\mathcal{L}$, a lattice of relations.
  This involves computing $r_i's$ such that $[\mathfrak{l}_i] = [\mathfrak{g}^{r_i}]$ in $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$. ⟵ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 ⟵ record breaking.
- Compute $\mathcal{L}$, a lattice of relations.
  This involves computing r_i's such that $[\mathfrak{l}_i] = [\mathfrak{g}^{r_i}]$ in $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.
- Lattice reduction.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 $\longleftarrow$ record breaking.
- Compute $\mathcal{L}$, a lattice of relations.
  This involves computing $r_i's$ such that $[\mathfrak{l}_i] = [\mathfrak{g}^{r_i}]$ in $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.
- Lattice reduction.

Online:

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 $\longleftarrow$ record breaking.
- Compute $\mathcal{L}$, a lattice of relations.
  This involves computing $r_i's$ such that $[\mathfrak{l}_i] = [\mathfrak{g}^{r_i}]$ in $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.
- Lattice reduction.

Online:

- Approximate-CVP $\Rightarrow \mathfrak{g}^e = \prod_{i=1}^{i=n} \mathfrak{l}_i^{e_i}$.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 $\longleftarrow$ record breaking.
- Compute $\mathcal{L}$, a lattice of relations.
  This involves computing $r_i' s$ such that $[\mathfrak{l}_i] = [\mathfrak{g}^{r_i}]$ in $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.
- Lattice reduction.

Online:

- Approximate-CVP $\Rightarrow \mathfrak{g}^e = \prod_{i=1}^{i=n} \mathfrak{l}_i^{e_i}$.
- Do the action! $\mathfrak{g}^e \star E = \prod_{i=1}^{i=n} \mathfrak{l}_i^{e_i} \star E$.

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 $\longleftarrow$ record breaking.
- Compute $\mathcal{L}$, a lattice of relations.
  This involves computing $r_i's$ such that $[\mathfrak{l}_i] = [\mathfrak{g}^{r_i}]$ in $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.
- Lattice reduction.

Online:

- Approximate-CVP $\Rightarrow \mathfrak{g}^e = \prod_{i=1}^{i=n} \mathfrak{l}_i^{e_i}$.
- Do the action! $\mathfrak{g}^e \star E = \prod_{i=1}^{i=n} \mathfrak{l}_i^{e_i} \star E$.

This strategy turns CSIDH group action into an effective group action (**EGA**)!

Signatures? II — CSI-FiSh [Beullens-Kleinjung-Vercauteren 2019]

Intuition: The Σ-protocol is secure if one can compute directly the action of $\mathfrak{g}^e$ on E. **But:**

- It is hard to find a generator of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.$\longleftarrow$ takes subexponential time on a classical computer
- Direct computation of $\mathfrak{g}^e \star E$ is not efficient.

Strategy of CSI-FiSh:

Offline:

- Find a generator $\mathfrak{g}$ of $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$ for p as in CSIDH-512 $\longleftarrow$ record breaking.
- Compute $\mathcal{L}$, a lattice of relations.
  This involves computing $r_i's$ such that $[\mathfrak{l}_i] = [\mathfrak{g}^{r_i}]$ in $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.
- Lattice reduction.

Online:

- Approximate-CVP $\Rightarrow \mathfrak{g}^e = \prod_{i=1}^{i=n} \mathfrak{l}_i^{e_i}$.
- Do the action! $\mathfrak{g}^e \star E = \prod_{i=1}^{i=n} \mathfrak{l}_i^{e_i} \star E$.

This strategy turns CSIDH group action into an effective group action (**EGA**)! But it does **NOT scale**.

Benefits of an EGA (compared with R(estricted)EGA)

Benefits of an EGA (compared with R(estricted)EGA)

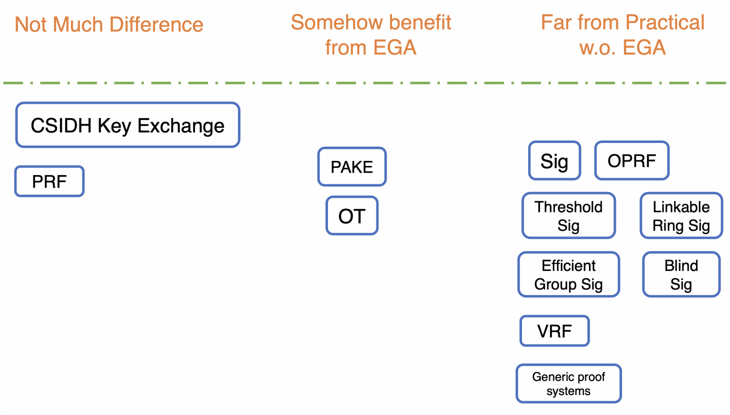


Figure: Table credit to Yi-Fu Lai.

Security of CSIDH

Security of CSIDH

Group Action Computational Diffie-Hellman

Given $s \in S$, $g \star s$ and $h \star s$ for $g, h \in G$, compute $(gh) \star s$.

Vectorization

Given $s, t \in S$, find $g \in G$ such that $t = g \star s$.

Security of CSIDH

Group Action Computational Diffie-Hellman

Given $s \in S$, $g \star s$ and $h \star s$ for $g, h \in G$, compute $(gh) \star s$.

Vectorization

Given $s, t \in S$, find $g \in G$ such that $t = g \star s$.

- There is a subexponential-time quantum algorithm to solve the vectorization problem for abelian groups – this is an abelian hidden shift problem and one can use Kuperberg's algorithm.

Security of CSIDH

Group Action Computational Diffie-Hellman

Given $s \in S$, $g \star s$ and $h \star s$ for $g, h \in G$, compute $(gh) \star s$.

Vectorization

Given $s, t \in S$, find $g \in G$ such that $t = g \star s$.

- There is a subexponential-time quantum algorithm to solve the vectorization problem for abelian groups – this is an abelian hidden shift problem and one can use Kuperberg's algorithm.
- Since 2019, a series of papers studied the quantum security of CSIDH, leaving whether CSIDH-512 and CSIDH-1024 achieve **NIST level 1 security** under doubt.

Generalizing the CSIDH group action

Generalizing the CSIDH group action

CSIDH group action

Generalizing the CSIDH group action

CSIDH group action

$$\mathsf{Cl}(\mathbb{Z}[\sqrt{-p}]) \curvearrowright \{\text{supersingular elliptic curves } E/\mathbb{F}_p\}$$

Generalizing the CSIDH group action

CSIDH group action

$$\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}]) \curvearrowright \{\text{supersingular elliptic curves } E/\mathbb{F}_p\}$$

General Oriention induced group action [Colò-Kohel 2020]

Generalizing the CSIDH group action

CSIDH group action

$$\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}]) \curvearrowright \{\text{supersingular elliptic curves } E/\mathbb{F}_p\}$$

General Oriention induced group action [Colò-Kohel 2020]

$$\mathrm{Cl}(\mathfrak{O}) \curvearrowright \mathcal{S}_{\mathfrak{O}}(p) = \{(E, \theta) \mid \theta \text{ defines an } \mathfrak{O}\text{-orientation on } E\}$$

Generalizing the CSIDH group action

CSIDH group action

$$\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}]) \curvearrowright \{\text{supersingular elliptic curves } E/\mathbb{F}_p\}$$

General Oriention induced group action [Colò-Kohel 2020]

$$\mathrm{Cl}(\mathfrak{O}) \curvearrowright \mathcal{S}_{\mathfrak{O}}(p) = \{(E, \theta) \mid \theta \text{ defines an } \mathfrak{O}\text{-orientation on } E\}$$

- $\mathfrak{O}$ is taken to be $\mathbb{Z}[\sqrt{-p}]$ in CSIDH;

Generalizing the CSIDH group action

CSIDH group action

$$\mathsf{Cl}(\mathbb{Z}[\sqrt{-p}]) \curvearrowright \{\text{supersingular elliptic curves } E/\mathbb{F}_p\}$$

General Oriention induced group action [Colò-Kohel 2020]

$$\mathsf{Cl}(\mathfrak{O}) \curvearrowright \mathcal{S}_{\mathfrak{O}}(p) = \{(E, \theta) \mid \theta \text{ defines an } \mathfrak{O}\text{-orientation on } E\}$$

- $\mathfrak{O}$ is taken to be $\mathbb{Z}[\sqrt{-p}]$ in CSIDH;
- θ in CSIDH is the natural Frobenius map on curves over $\mathbb{F}_p$.

SCALLOP [De Feo-Fouotsa-Kutas-Leroux-Merz-Panny-Wesolowski 2023]

SCALLOP [De Feo-Fouotsa-Kutas-Leroux-Merz-Panny-Wesolowski 2023]

Main idea: $\mathrm{Cl}(\mathfrak{O})$ is easy to compute for orders of the form $\mathbb{Z}[f\sqrt{-d}]$ with d being a small positive integer.

SCALLOP [De Feo-Fouotsa-Kutas-Leroux-Merz-Panny-Wesolowski 2023]

Main idea: $\mathrm{Cl}(\mathfrak{O})$ is easy to compute for orders of the form $\mathbb{Z}[f\sqrt{-d}]$ with d being a small positive integer.

Summary:

- SCALLOP follows the overall strategy proposed by CSI-FiSh.

SCALLOP [De Feo-Fouotsa-Kutas-Leroux-Merz-Panny-Wesolowski 2023]

Main idea: $\mathrm{Cl}(\mathfrak{O})$ is easy to compute for orders of the form $\mathbb{Z}[f\sqrt{-d}]$ with d being a small positive integer.

Summary:

- SCALLOP follows the overall strategy proposed by CSI-FiSh.
- SCALLOP resolves the scaling issue faced by CSI-FiSh.

SCALLOP [De Feo-Fouotsa-Kutas-Leroux-Merz-Panny-Wesolowski 2023]

Main idea: $\mathrm{Cl}(\mathfrak{O})$ is easy to compute for orders of the form $\mathbb{Z}[f\sqrt{-d}]$ with d being a small positive integer.

Summary:

- SCALLOP follows the overall strategy proposed by CSI-FiSh.
- SCALLOP resolves the scaling issue faced by CSI-FiSh.
- For security reasons, f is chosen to be a large prime.

SCALLOP [De Feo-Fouotsa-Kutas-Leroux-Merz-Panny-Wesolowski 2023]

Main idea: $\mathrm{Cl}(\mathfrak{O})$ is easy to compute for orders of the form $\mathbb{Z}[f\sqrt{-d}]$ with d being a small positive integer.

Summary:

- SCALLOP follows the overall strategy proposed by CSI-FiSh.
- SCALLOP resolves the scaling issue faced by CSI-FiSh.
- For security reasons, f is chosen to be a large prime.
- There is a tradeoff between choosing f so that there is an efficient representation of θ or having a smoother $\#\mathrm{Cl}(\mathfrak{O})$ which is helpful for solving the discrete log.

SCALLOP [De Feo-Fouotsa-Kutas-Leroux-Merz-Panny-Wesolowski 2023]

Main idea: $\mathrm{Cl}(\mathfrak{O})$ is easy to compute for orders of the form $\mathbb{Z}[f\sqrt{-d}]$ with d being a small positive integer.

Summary:

- SCALLOP follows the overall strategy proposed by CSI-FiSh.
- SCALLOP resolves the scaling issue faced by CSI-FiSh.
- For security reasons, f is chosen to be a large prime.
- There is a tradeoff between choosing f so that there is an efficient representation of θ or having a smoother $\#\mathrm{Cl}(\mathfrak{O})$ which is helpful for solving the discrete log.
  $\longleftarrow$ SCALLOP still has its scaling bottleneck

A quick recap of what we have achieved so far

A quick recap of what we have achieved so far

	year	$\mathrm{Cl}(\mathfrak{O})$	$\mathcal{S}_{\mathfrak{O}}(p)$	type	scalability
CSIDH	2018	$\mathfrak{O} = \mathbb{Z}[\sqrt{-p}]$	E	REGA	freely
CSI-FiSh	2019	$\mathfrak{O} = \mathbb{Z}[\sqrt{-p}]$	E	EGA	CSIDH-512
SCALLOP	2023	$\mathfrak{O} = \mathbb{Z}[f\sqrt{-d}]$	(E, ι)	EGA	CSIDH-1024
SCALLOP-HD	2024	$\mathfrak{O} = \mathbb{Z}[f\sqrt{-d}]$	(E, ι)	EGA	??

New isogeny representation after SIDH attacks

New isogeny representation after SIDH attacks

Let φ, φ' be *a*-isogenies and ψ, ψ' be *b*-isogenies for integers *a*, *b* that satisfy the commutative diagram:

New isogeny representation after SIDH attacks

Let φ, φ' be a-isogenies and ψ, ψ' be b-isogenies for integers a, b that satisfy the commutative diagram:

$$\begin{array}{ccc} E_1' & \xrightarrow{\varphi'} & E_2' \\ \uparrow{\scriptstyle\psi} & & \uparrow{\scriptstyle\psi'} \\ E_1 & \xrightarrow{\varphi} & E_2. \end{array}$$

New isogeny representation after SIDH attacks

Let φ, φ' be a-isogenies and ψ, ψ' be b-isogenies for integers a, b that satisfy the commutative diagram:

$$\begin{array}{ccc} E_1' & \xrightarrow{\varphi'} & E_2' \\ \uparrow{\scriptstyle \psi} & & \uparrow{\scriptstyle \psi'} \\ E_1 & \xrightarrow{\varphi} & E_2. \end{array}$$

Define $F : E_2 \times E_1' \longrightarrow E_1 \times E_2'$ by the matrix form $\begin{pmatrix} \hat{\varphi} & -\hat{\psi} \\ \psi' & \varphi' \end{pmatrix}$.

New isogeny representation after SIDH attacks

Let φ, φ' be a-isogenies and ψ, ψ' be b-isogenies for integers a, b that satisfy the commutative diagram:

$$\begin{array}{ccc} E_1' & \xrightarrow{\varphi'} & E_2' \\ \uparrow{\scriptstyle \psi} & & \uparrow{\scriptstyle \psi'} \\ E_1 & \xrightarrow{\varphi} & E_2. \end{array}$$

Define $F : E_2 \times E_1' \longrightarrow E_1 \times E_2'$ by the matrix form $\begin{pmatrix} \hat{\varphi} & -\hat{\psi} \\ \psi' & \varphi' \end{pmatrix}$.

F is a d-isogeny between abelian surfaces with $d = a + b$.

New isogeny representation after SIDH attacks

Let φ, φ' be a-isogenies and ψ, ψ' be b-isogenies for integers a, b that satisfy the commutative diagram:

$$\begin{array}{ccc} E_1' & \xrightarrow{\varphi'} & E_2' \\ \uparrow{\scriptstyle \psi} & & \uparrow{\scriptstyle \psi'} \\ E_1 & \xrightarrow{\varphi} & E_2. \end{array}$$

Define $F : E_2 \times E_1' \longrightarrow E_1 \times E_2'$ by the matrix form $\begin{pmatrix} \hat{\varphi} & -\hat{\psi} \\ \psi' & \varphi' \end{pmatrix}$.

F is a d-isogeny between abelian surfaces with $d = a + b$.

If $\ker \varphi \cap \ker \psi = \{0\}$,

$$\ker(F) = \{(\varphi(x), \psi(x)) \mid x \in E_1[d]\}. \text{ [Kani97']}$$

New isogeny representation after SIDH attacks

Let φ, φ' be a-isogenies and ψ, ψ' be b-isogenies for integers a, b that satisfy the commutative diagram:

$$\begin{array}{ccc} E_1' & \xrightarrow{\varphi'} & E_2' \\ \uparrow{\scriptstyle\psi} & & \uparrow{\scriptstyle\psi'} \\ E_1 & \xrightarrow{\varphi} & E_2. \end{array}$$

Define $F : E_2 \times E_1' \longrightarrow E_1 \times E_2'$ by the matrix form $\begin{pmatrix} \hat{\varphi} & -\hat{\psi} \\ \psi' & \varphi' \end{pmatrix}$.

F is a d-isogeny between abelian surfaces with $d = a + b$.
If $\ker \varphi \cap \ker \psi = \{0\}$,

$$\ker(F) = \{(\varphi(x), \psi(x)) \mid x \in E_1[d]\}. \text{ [Kani97']}$$

$\longrightarrow$ Upshot: An isogeny can be represented by its evaluation on torsion points! (a priori only kernel representation)

2dim-representation of orientations and endomorphisms

2dim-representation of orientations and endomorphisms

Definition

Let $\mathfrak{O}$ be an imaginary quadratic order with discriminant $D_{\mathfrak{O}}$. Given an $\mathfrak{O}$-oriented supersingular elliptic curve (E, ι), take any $\omega \in \mathfrak{O}$ such that $\mathfrak{O} = \mathbb{Z}[\omega]$ and define $\omega_E := \iota(\omega)$. Let $\beta \in \mathfrak{O}$ such that $n(\omega) + n(\beta) = 2^e$ and $\gcd(n(\beta), n(\omega)) = 1$. Let P, Q be a basis of $E[2^e]$. Then the tuple $(E, \omega, \beta, P, Q, \omega_E(P), \omega_E(Q))$ is called a 2dim-**representation of** (E, ι).

2dim-representation of orientations and endomorphisms

Definition

Let $\mathfrak{O}$ be an imaginary quadratic order with discriminant $D_{\mathfrak{O}}$. Given an $\mathfrak{O}$-oriented supersingular elliptic curve (E, ι), take any $\omega \in \mathfrak{O}$ such that $\mathfrak{O} = \mathbb{Z}[\omega]$ and define $\omega_E := \iota(\omega)$. Let $\beta \in \mathfrak{O}$ such that $n(\omega) + n(\beta) = 2^e$ and $\gcd(n(\beta), n(\omega)) = 1$. Let P, Q be a basis of $E[2^e]$. Then the tuple $(E, \omega, \beta, P, Q, \omega_E(P), \omega_E(Q))$ is called a 2dim-**representation of** (E, ι).

Proposition

Let $\mathfrak{O}$ be an imaginary quadratic order of discriminant $D_{\mathfrak{O}} \equiv 5 \bmod 8$, then any $(E, \iota) \in \mathcal{S}_{\mathfrak{O}}(p)$ admits a 2dim-representation.

SCALLOP-HD

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

Benefits:

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

Benefits:

- Much simpler group action formula.

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

Benefits:

- Much simpler group action formula. $\longleftarrow$ more details in the paper

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

Benefits:

- Much simpler group action formula. $\longleftarrow$ more details in the paper
- No restriction on f coming from obtaining an efficient representation for the orientation.

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

Benefits:

- Much simpler group action formula. $\longleftarrow$ more details in the paper
- No restriction on f coming from obtaining an efficient representation for the orientation.
- Therefore we can choose f so that $\#\mathrm{Cl}(\mathfrak{O})$ is smooth, and use Pohlig-Hellman algorithm to solve the discrete log problems efficiently.

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

Benefits:

- Much simpler group action formula. $\longleftarrow$ more details in the paper
- No restriction on f coming from obtaining an efficient representation for the orientation.
- Therefore we can choose f so that $\#\mathrm{Cl}(\mathfrak{O})$ is smooth, and use Pohlig-Hellman algorithm to solve the discrete log problems efficiently. $\longleftarrow$ overcomes the scaling bottleneck of SCALLOP

SCALLOP-HD

Main idea: use 2dim-representation to represent θ in (E, θ).

Benefits:

- Much simpler group action formula. $\longleftarrow$ more details in the paper
- No restriction on f coming from obtaining an efficient representation for the orientation.
- Therefore we can choose f so that $\#\mathrm{Cl}(\mathfrak{O})$ is smooth, and use Pohlig-Hellman algorithm to solve the discrete log problems efficiently. $\longleftarrow$ overcomes the scaling bottleneck of SCALLOP
    - The scalability of SCALLOP-HD now depends only on lattice algorithms.

Implementation and performance

Implementation and performance

Scalability

Implementation and performance

Scalability We managed to scale to CSIDH-4096.

Implementation and performance

Scalability We managed to scale to CSIDH-4096. **An issue** We haven't finished generating a starting curve for 2048 and 4096, due to the lack of sufficiently general genus-2 isogeny libraries.

Implementation and performance

Scalability We managed to scale to CSIDH-4096. **An issue** We haven't finished generating a starting curve for 2048 and 4096, due to the lack of sufficiently general genus-2 isogeny libraries.

Performance

Implementation and performance

Scalability We managed to scale to CSIDH-4096. **An issue** We haven't finished generating a starting curve for 2048 and 4096, due to the lack of sufficiently general genus-2 isogeny libraries.

Performance

CSIDH-n	512	1024	2048	4096
f	254	508	1021	2043
n	74	100	200	300
p	1137	1909	tbf	tbf

Table: Bit-size for f, n and p.

Implementation and performance

Scalability We managed to scale to CSIDH-4096. **An issue** We haven't finished generating a starting curve for 2048 and 4096, due to the lack of sufficiently general genus-2 isogeny libraries.

Performance

CSIDH-n	512	1024	2048	4096
f	254	508	1021	2043
n	74	100	200	300
p	1137	1909	tbf	tbf

Table: Bit-size for f, n and p.

	512	1024	2048 & 4096
SCALLOP	42 sec	15 min	—
SCALLOP-HD	88 sec	19 min	tbf

Table: Runtime for a single group action evaluation. Experiments run on an Intel Alder Lake CPU core clocked at 2.1 GHz. C++ implementation of SCALLOP compared with SageMath implementation of SCALLOP-HD.

Thank you!

ePrint:2023/1488