Post-Quantum Adaptor Signature for Privacy-Preserving Off-Chain Payments

Erkan Tairi [1]   Pedro Moreno-Sanchez [2]   Matteo Maffei [1]

Financial Cryptography and Data Privacy 2021

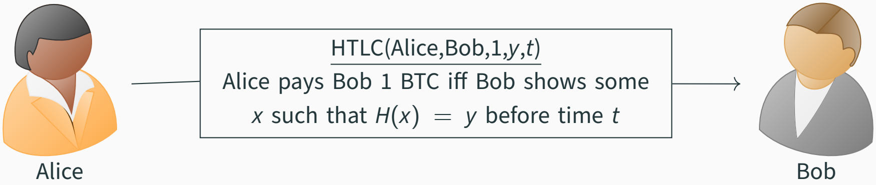
[1] TU Wien

[2] IMDEA Software Institute

Der Wissenschaftsfonds.

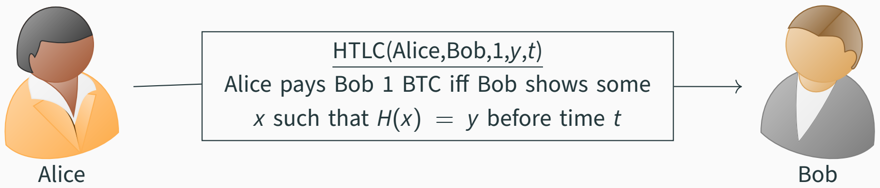
Conditional Payments

Conditional payments allow for rich set of off-chain functionalities such as payment channel networks, payment channel hubs, atomic swaps, etc. We can use a hash-time lock contract (HTLC) for conditional payment.



Conditional Payments

Conditional payments allow for rich set of off-chain functionalities such as payment channel networks, payment channel hubs, atomic swaps, etc. We can use a hash-time lock contract (HTLC) for conditional payment.

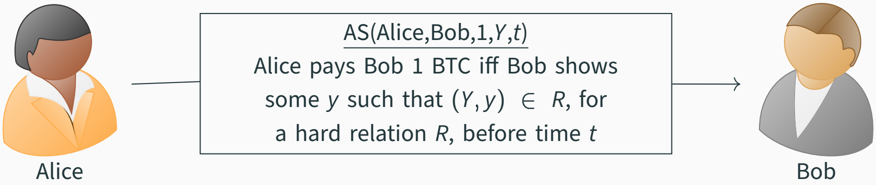


HTLC disadvantages: requires all cryptocurrencies to support the same hash function, using the same hash value causes privacy issues, undesirable on-chain footprint, lack of fungibility, etc.

Adaptor signature (AS) extends ordinary signature with a compatible hard relation. It was first introduced by Poelstra, and recently formalized by Aumayr et al. [AEE+20]. We can perform conditional payment using adaptor signature.

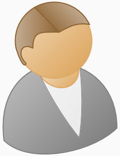
Adaptor Signatures

Adaptor signature (AS) extends ordinary signature with a compatible hard relation. It was first introduced by Poelstra, and recently formalized by Aumayr et al. [AEE+20]. We can perform conditional payment using adaptor signature.



AS advantages: can leverage the existing signature of the cryptocurrency, low on-chain cost, improved fungibility of transactions, etc.

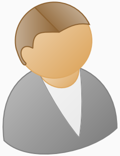
Let $\Sigma = (\mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ be a signature scheme and $(Y, y) \in R$ be a hard relation (y witness, Y statement). An adaptor signature $\Xi_{\Sigma,R} = (\mathsf{PreSig}, \mathsf{PreVer}, \mathsf{Adapt}, \mathsf{Ext})$ works as follows:



$(Y, y) \in R$
pk, m

$(\mathsf{sk}, \mathsf{pk})$
Y, m

Let $\Sigma = (\mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ be a signature scheme and $(Y, y) \in R$ be a hard relation (y witness, Y statement). An adaptor signature $\Xi_{\Sigma,R} = (\mathsf{PreSig}, \mathsf{PreVer}, \mathsf{Adapt}, \mathsf{Ext})$ works as follows:



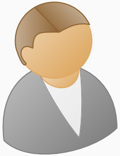
$(Y, y) \in R$
pk, m

$(\mathsf{sk}, \mathsf{pk})$
Y, m

$\hat{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, Y)$

Adaptor Signatures

Let $\Sigma = (\mathrm{KGen}, \mathrm{Sig}, \mathrm{Ver})$ be a signature scheme and $(Y, y) \in R$ be a hard relation (y witness, Y statement). An adaptor signature $\Xi_{\Sigma,R} = (\mathrm{PreSig}, \mathrm{PreVer}, \mathrm{Adapt}, \mathrm{Ext})$ works as follows:



$(Y, y) \in R$
pk, m

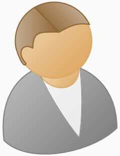
$(\mathrm{sk}, \mathrm{pk})$
Y, m

$\hat{\sigma} \leftarrow \mathrm{PreSig}(\mathrm{sk}, m, Y)$

$\xleftarrow{\quad\hat{\sigma}\quad}$

Adaptor Signatures

Let $\Sigma = (\mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ be a signature scheme and $(Y, y) \in R$ be a hard relation (y witness, Y statement). An adaptor signature $\Xi_{\Sigma,R} = (\mathsf{PreSig}, \mathsf{PreVer}, \mathsf{Adapt}, \mathsf{Ext})$ works as follows:



$(Y, y) \in R$
pk, m

$(\mathsf{sk}, \mathsf{pk})$
Y, m

$\hat{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, Y)$

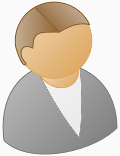
$\xleftarrow{\quad\hat{\sigma}\quad}$

$0/1 \leftarrow \mathsf{PreVer}(\mathsf{pk}, m, Y, \hat{\sigma})$

$\sigma \leftarrow \mathsf{Adapt}(\hat{\sigma}, y)$

Adaptor Signatures

Let $\Sigma = (\mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ be a signature scheme and $(Y, y) \in R$ be a hard relation (y witness, Y statement). An adaptor signature $\Xi_{\Sigma,R} = (\mathsf{PreSig}, \mathsf{PreVer}, \mathsf{Adapt}, \mathsf{Ext})$ works as follows:



$(Y, y) \in R$
pk, m

$(\mathsf{sk}, \mathsf{pk})$
Y, m

$\hat{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, Y)$

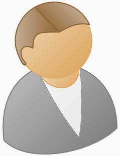
$\xleftarrow{\quad\hat{\sigma}\quad}$

$0/1 \leftarrow \mathsf{PreVer}(\mathsf{pk}, m, Y, \hat{\sigma})$

$\sigma \leftarrow \mathsf{Adapt}(\hat{\sigma}, y)$

$\xrightarrow{\quad\sigma\quad}$

Let $\Sigma = (\mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ be a signature scheme and $(Y, y) \in R$ be a hard relation (y witness, Y statement). An adaptor signature $\Xi_{\Sigma,R} = (\mathsf{PreSig}, \mathsf{PreVer}, \mathsf{Adapt}, \mathsf{Ext})$ works as follows:



$(Y, y) \in R$
pk, m

$(\mathsf{sk}, \mathsf{pk})$
Y, m

$\hat{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, Y)$

$\xleftarrow{\quad\hat{\sigma}\quad}$

$0/1 \leftarrow \mathsf{PreVer}(\mathsf{pk}, m, Y, \hat{\sigma})$

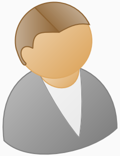
$\sigma \leftarrow \mathsf{Adapt}(\hat{\sigma}, y)$

$\xrightarrow{\quad\sigma\quad}$

$y \leftarrow \mathsf{Ext}(\sigma, \hat{\sigma}, Y)$

Adaptor Signatures

Let $\Sigma = (\mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ be a signature scheme and $(Y, y) \in R$ be a hard relation (y witness, Y statement). An adaptor signature $\Xi_{\Sigma,R} = (\mathsf{PreSig}, \mathsf{PreVer}, \mathsf{Adapt}, \mathsf{Ext})$ works as follows:



$(Y, y) \in R$
pk, m

$(\mathsf{sk}, \mathsf{pk})$
Y, m

$\hat{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, Y)$

$\xleftarrow{\hat{\sigma}}$

$0/1 \leftarrow \mathsf{PreVer}(\mathsf{pk}, m, Y, \hat{\sigma})$

$\sigma \leftarrow \mathsf{Adapt}(\hat{\sigma}, y)$

$\xrightarrow{\sigma}$

$y \leftarrow \mathsf{Ext}(\sigma, \hat{\sigma}, Y)$

PreSig is like a **commitment**, such that Alice with a valid witness can **complete** the signature. Moreover, any valid $(\sigma, \hat{\sigma})$ pair **reveals** the witness.

Adaptor Signature Properties

- **Unforgeability:** infeasible to forge a signature even when pre-signature is given without knowing a witness to R
- **Pre-signature Adaptability:** anyone that knows a witness to Y can complete a pre-signature computed with Y
- **Witness Extractability:** any valid (pre-signature, signature) pair computed with the statement Y reveals a witness to Y

Why Post-quantum Adaptor Signature?

- Existing adaptor signatures (i.e., Schnorr and ECDSA) from [AEE+20] are broken with a quantum computer due to Shor's algorithm.
- Ongoing standardization process by NIST (only limited set of candidate post-quantum assumptions).

Why Post-quantum Adaptor Signature?

- Existing adaptor signatures (i.e., Schnorr and ECDSA) from [AEE$^+$20] are broken with a quantum computer due to Shor's algorithm.
- Ongoing standardization process by NIST (only limited set of candidate post-quantum assumptions).

Esgin et al. [EEE20] introduced lattice-based adaptor signature (LAS), which is based on Module-SIS and Module-LWE problems.

Drawbacks of LAS

- Due to inherent **knowledge gap** in lattice-based ZK proofs, it requires an **extended** relation R' such that $R \subseteq R'$ (i.e., witnesses can have bigger norm in R')
- **Weak Pre-signature Adaptability:** anyone that knows a y with $(Y, y) \in R$ can complete a pre-signature conditioned on Y
  - $\sigma \leftarrow \mathsf{Adapt}(\hat{\sigma}, y)$ where $(Y, y) \in R$
- **Witness Extractability:** any given (pre-signature, signature) pair on the same statement Y reveals a witness y' such that $(Y, y') \in R'$
  - $y'/\perp \leftarrow \mathsf{Ext}(\sigma, \hat{\sigma}, Y)$ such that $(Y, y') \in R'$

Drawbacks of LAS

- Due to inherent **knowledge gap** in lattice-based ZK proofs, it requires an **extended** relation R' such that $R \subseteq R'$ (i.e., witnesses can have bigger norm in R')
- **Weak Pre-signature Adaptability:** anyone that knows a y with $(Y, y) \in R$ can complete a pre-signature conditioned on Y
    - $\sigma \leftarrow \mathrm{Adapt}(\hat{\sigma}, y)$ where $(Y, y) \in R$
- **Witness Extractability:** any given (pre-signature, signature) pair on the same statement Y reveals a witness y' such that $(Y, y') \in R'$
    - $y'/\perp \leftarrow \mathrm{Ext}(\sigma, \hat{\sigma}, Y)$ such that $(Y, y') \in R'$

Drawback

Extracted witnesses do **NOT** guarantee adaptability (i.e., imperfect correctness). We can guarantee correctness by using an expensive ZK proof that the witness has small norm (e.g., the proof from [ENS20] is 47KB).

Some off-chain applications (e.g., payment channel network of Malavolta et al. [MMS+19]) require several concatenated instances of pre-signatures (i.e., interleaved conditions).

$$\mathsf{PreSig}(\mathsf{sk}_1, m_1, Y_1) \to \cdots \to \mathsf{PreSig}(\mathsf{sk}_n, m_n, Y_n),$$

for a hard relation R and statement/witness pairs $(Y_i, y_i) \in R$, such that $Y_{i+1} = f(Y_i, z_{i+1})$ for a function f and a random value z_{i+1}. The privacy of these constructions require that each pair of (Y_i, y_i) is indistinguishable from others.

Some off-chain applications (e.g., payment channel network of Malavolta et al. [MMS+19]) require several concatenated instances of pre-signatures (i.e., interleaved conditions).

$$\mathsf{PreSig}(\mathsf{sk}_1, m_1, Y_1) \to \cdots \to \mathsf{PreSig}(\mathsf{sk}_n, m_n, Y_n),$$

for a hard relation R and statement/witness pairs $(Y_i, y_i) \in R$, such that $Y_{i+1} = f(Y_i, z_{i+1})$ for a function f and a random value z_{i+1}. The privacy of these constructions require that each pair of (Y_i, y_i) is indistinguishable from others.

- In group-based setting (e.g., Schnorr/ECDSA), we have that $Y_1 = g^{z_1}$ and $Y_{i+1} = Y_i \cdot g^{z_{i+1}}$, for random scalars $z_i \leftarrow_{\$} \mathbb{Z}_q$.

Drawbacks of LAS

Some off-chain applications (e.g., payment channel network of Malavolta et al. [MMS+19]) require several concatenated instances of pre-signatures (i.e., interleaved conditions).

$$\mathsf{PreSig}(\mathsf{sk}_1, m_1, Y_1) \rightarrow \cdots \rightarrow \mathsf{PreSig}(\mathsf{sk}_n, m_n, Y_n),$$

for a hard relation R and statement/witness pairs $(Y_i, y_i) \in R$, such that $Y_{i+1} = f(Y_i, z_{i+1})$ for a function f and a random value z_{i+1}. The privacy of these constructions require that each pair of (Y_i, y_i) is indistinguishable from others.

- In group-based setting (e.g., Schnorr/ECDSA), we have that $Y_1 = g^{z_1}$ and $Y_{i+1} = Y_i \cdot g^{z_{i+1}}$, for random scalars $z_i \leftarrow_{\$} \mathbb{Z}_q$.
- In lattice-based setting (e.g., LAS), we have that $Y_1 = Az_1$ and $Y_{i+1} = Y_i + Az_{i+1}$, for random vectors $z_i \leftarrow_{\$} \mathbb{S}_1^{n+\ell}$ (i.e., vectors of norm 1).

Drawbacks of LAS

Some off-chain applications (e.g., payment channel network of Malavolta et al. [MMS$^+$19]) require several concatenated instances of pre-signatures (i.e., interleaved conditions).

$$\mathsf{PreSig}(\mathsf{sk}_1, m_1, Y_1) \to \cdots \to \mathsf{PreSig}(\mathsf{sk}_n, m_n, Y_n),$$

for a hard relation R and statement/witness pairs $(Y_i, y_i) \in R$, such that $Y_{i+1} = f(Y_i, z_{i+1})$ for a function f and a random value z_{i+1}. The privacy of these constructions require that each pair of (Y_i, y_i) is indistinguishable from others.

- In group-based setting (e.g., Schnorr/ECDSA), we have that $Y_1 = g^{z_1}$ and $Y_{i+1} = Y_i \cdot g^{z_{i+1}}$, for random scalars $z_i \leftarrow_{\$} \mathbb{Z}_q$.
- In lattice-based setting (e.g., LAS), we have that $Y_1 = Az_1$ and $Y_{i+1} = Y_i + Az_{i+1}$, for random vectors $z_i \leftarrow_{\$} \mathbb{S}_1^{n+\ell}$ (i.e., vectors of norm 1).

Drawback

In lattice-based setting the norm of the witness vectors is increasing along the path (i.e., $\|z_1 + \cdots + z_i\| \leq \|z_1\| + \cdots + \|z_i\|$), which in turn hinders the privacy of applications.

Research Question

The only existing post-quantum adaptor signature LAS [EEE20] has an imperfect correctness and hinders the privacy of off-chain applications that use it. This naturally leads us to the following question:

> *Can we construct an adaptor signature scheme that is correct and secure against quantum adversaries, but preserves the privacy guarantees of the off-chain applications built on top of it?*

The only existing post-quantum adaptor signature LAS [EEE20] has an imperfect correctness and hinders the privacy of off-chain applications that use it. This naturally leads us to the following question:

> *Can we construct an adaptor signature scheme that is correct and secure against quantum adversaries, but preserves the privacy guarantees of the off-chain applications built on top of it?*

Yes!

Group Action

Definition

A **group action** on G of X is a function $\star\colon G \times X \to X$, such that

- $e \star x = x$,
- $(gh) \star x = g \star (h \star x)$,

for an identity element e of G, and $g, h \in G$ and $x \in X$. Furthermore, we say that the group action is **one-way** if given $(x, y = g \star x)$, where $g \leftarrow_{\$} G$, no efficient attacker can find g.

Group Action

Definition

A **group action** on G of X is a function $\star\colon G \times X \to X$, such that

- $e \star x = x$,
- $(gh) \star x = g \star (h \star x)$,

for an identity element e of G, and $g, h \in G$ and $x \in X$. Furthermore, we say that the group action is **one-way** if given $(x, y = g \star x)$, where $g \leftarrow_{\$} G$, no efficient attacker can find g.

Example

Let $\mathbb{H}$ be a group of prime order q with generator h. Consider $\star\colon \mathbb{Z}_q^* \times \mathbb{H} \to \mathbb{H}$ where

$$z \star h := h^z.$$

- $\mathbb{Z}_q^*$ is the "group" of action, and $\mathbb{H}$ is the "set" of action (although $\mathbb{H}$ is a group here).
- If DLog is hard over $\mathbb{H}$, then $\star\colon \mathbb{Z}_q^* \times \mathbb{H} \to \mathbb{H}$ is one-way.
- The "set" $\mathbb{H}$ is a group, hence, one-wayness does **NOT** hold against quantum attackers.

We can construct an isogeny-based group action by letting G be the class group $\mathrm{Cl}(\mathcal{O})$ of an order $\mathcal{O} \subset \mathbb{Q}(\sqrt{-D})$, and X be the set of elliptic curves with complex multiplication by $\mathcal{O}$ (as in [CLM$^+$18, BKV19]).

Isogeny-based Group Action

We can construct an isogeny-based group action by letting G be the class group $\mathrm{Cl}(\mathcal{O})$ of an order $\mathcal{O} \subset \mathbb{Q}(\sqrt{-D})$, and X be the set of elliptic curves with complex multiplication by $\mathcal{O}$ (as in [CLM+18, BKV19]).

- For isogeny-based $\star\colon G \times X \to X$, there is no meaningful multiplication $x \cdot x'$.
- For DDH-based $\star\colon \mathbb{Z}_q^* \times \mathbb{H} \to \mathbb{H}$, we can compute $h \cdot h'$.

For a detailed exposition of cryptographic group actions refer to the work of Alamati et al. [ADFMP20].

We can construct an isogeny-based group action by letting G be the class group $\mathrm{Cl}(\mathcal{O})$ of an order $\mathcal{O} \subset \mathbb{Q}(\sqrt{-D})$, and X be the set of elliptic curves with complex multiplication by $\mathcal{O}$ (as in [CLM$^+$18, BKV19]).

- For isogeny-based $\star\colon G \times X \to X$, there is no meaningful multiplication $x \cdot x'$.
- For DDH-based $\star\colon \mathbb{Z}_q^* \times \mathbb{H} \to \mathbb{H}$, we can compute $h \cdot h'$.

For a detailed exposition of cryptographic group actions refer to the work of Alamati et al. [ADFMP20].

Notation

We uniquely represent elements of $\mathrm{Cl}(\mathcal{O})$ as $[a] = \mathfrak{g}^a$ for $a \in \mathbb{Z}_N$, and $N = \#\mathrm{Cl}(\mathcal{O})$, and generator $\mathfrak{g}$. Thus, we can write $[a]E$ for $\mathfrak{g}^a \star E$, and have $[a][b]E = [a+b]E$.

Definition (Group Action Inversion Problem (GAIP) [DFG19])

Given two elliptic curves E and E' over the same finite field and with $\mathrm{End}(E) = \mathrm{End}(E') = \mathcal{O}$, find an ideal $\mathfrak{a} \subset \mathcal{O}$ such that $E' = \mathfrak{a} \star E$.

Definition (Group Action Inversion Problem (GAIP) [DFG19])

Given two elliptic curves E and E' over the same finite field and with $\mathrm{End}(E) = \mathrm{End}(E') = \mathcal{O}$, find an ideal $\mathfrak{a} \subset \mathcal{O}$ such that $E' = \mathfrak{a} \star E$.

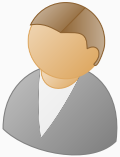
The best known quantum algorithm to solve GAIP is Kuperberg's algorithm for the hidden shift problem with subexponential complexity [Kup05].

Identification scheme from isogenies

E_0 is a designated base (starting) curve that is part of public parameters.

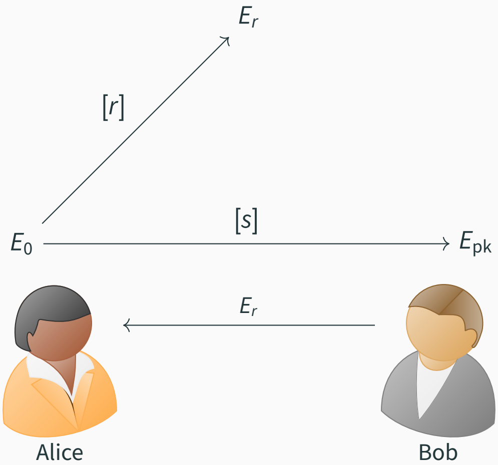
$$E_0 \xrightarrow{\qquad\qquad [s] \qquad\qquad} E_{\mathrm{pk}}$$

Alice



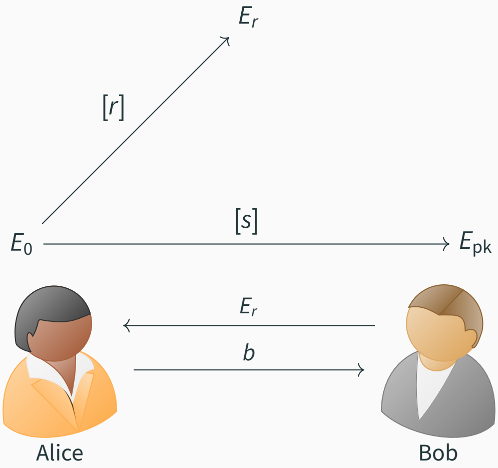
Bob

E_0 is a designated base (starting) curve that is part of public parameters.

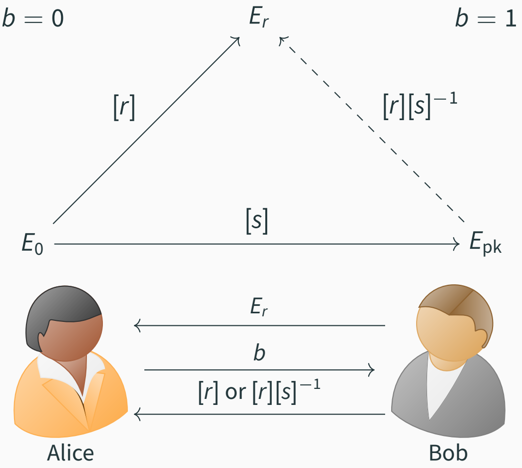


Identification scheme from isogenies

E_0 is a designated base (starting) curve that is part of public parameters.

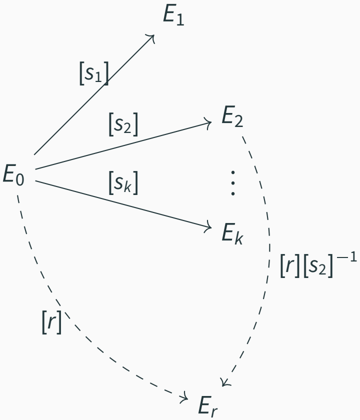


E_0 is a designated base (starting) curve that is part of public parameters.



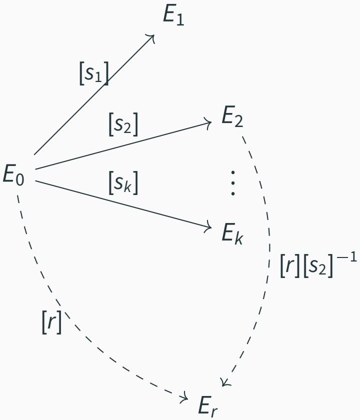
Increasing soundness

- The previous method only has soundness of $\frac{1}{2}$.
- We can increase soundness to $\frac{1}{2S-1}$ by using S public keys (elliptic curves) along with their quadratic twists.



Increasing soundness

- The previous method only has soundness of $\frac{1}{2}$.
- We can increase soundness to $\frac{1}{2S-1}$ by using S public keys (elliptic curves) along with their quadratic twists.
- Applying Fiat-Shamir transform and doing $t = \frac{\lambda}{\log_2 S}$ iterations to achieve security level λ, we obtain the signature scheme CSI-FiSh [BKV19].



Isogeny Adaptor Signature (IAS)

- We can construct an adaptor signature from CSI-FiSh [BKV19] using GAIP as the hard relation (for simplicity we consider the base scheme with challenge space $\{0, 1\}$).
- The main technical challenge appears in PreSig algorithm.

- We can construct an adaptor signature from CSI-FiSh [BKV19] using GAIP as the hard relation (for simplicity we consider the base scheme with challenge space $\{0, 1\}$).
- The main technical challenge appears in PreSig algorithm.

Schnorr

procedure $\mathrm{PreSig}(\mathrm{sk}, m, Y)$
$\quad r \leftarrow_{\$} \mathbb{Z}_q, R := g^r$
$\quad e := H(\mathrm{pk} \| R \cdot Y \| m)$
$\quad \hat{s} := r - e \cdot \mathrm{sk} \bmod q$
$\quad$ **return** $\hat{\sigma} := (e, \hat{s})$

IAS

procedure $\mathrm{PreSig}(\mathrm{sk}, m, E_Y)$
$\quad r \leftarrow_{\$} \mathrm{Cl}(\mathcal{O}), E_R := [r]E_0$
$\quad e := H(\mathrm{pk} \| E_R \cdot E_Y \| m)$
$\quad \hat{s} := r - e \cdot \mathrm{sk} \bmod N$
$\quad$ **return** $\hat{\sigma} := (e, \hat{s})$

- We can construct an adaptor signature from CSI-FiSh [BKV19] using GAIP as the hard relation (for simplicity we consider the base scheme with challenge space $\{0, 1\}$).
- The main technical challenge appears in PreSig algorithm.

Schnorr

procedure $\mathrm{PreSig}(\mathrm{sk}, m, Y)$

$\boxed{r \leftarrow_{\$} \mathbb{Z}_q, R := g^r}$

$e := H(\mathrm{pk} \| R \cdot Y \| m)$

$\hat{s} := r - e \cdot \mathrm{sk} \bmod q$

return $\hat{\sigma} := (e, \hat{s})$

IAS

procedure $\mathrm{PreSig}(\mathrm{sk}, m, E_Y)$

$\boxed{r \leftarrow_{\$} \mathrm{Cl}(\mathcal{O}), E_R := [r]E_0}$

$e := H(\mathrm{pk} \| E_R \cdot E_Y \| m)$

$\hat{s} := r - e \cdot \mathrm{sk} \bmod N$

return $\hat{\sigma} := (e, \hat{s})$

Isogeny Adaptor Signature (IAS)

- We can construct an adaptor signature from CSI-FiSh [BKV19] using GAIP as the hard relation (for simplicity we consider the base scheme with challenge space $\{0, 1\}$).
- The main technical challenge appears in PreSig algorithm.

Schnorr

procedure PreSig(sk, m, Y)
  $r \leftarrow_\$ \mathbb{Z}_q, R := g^r$
  $e := H(\mathsf{pk} \| R \cdot Y \| m)$
  $\boxed{\hat{s} := r - e \cdot \mathsf{sk} \bmod q}$
  return $\hat{\sigma} := (e, \hat{s})$

IAS

procedure PreSig(sk, m, E_Y)
  $r \leftarrow_\$ \mathrm{Cl}(\mathcal{O}), E_R := [r]E_0$
  $e := H(\mathsf{pk} \| E_R \cdot E_Y \| m)$
  $\boxed{\hat{s} := r - e \cdot \mathsf{sk} \bmod N}$
  return $\hat{\sigma} := (e, \hat{s})$

Isogeny Adaptor Signature (IAS)

- We can construct an adaptor signature from CSI-FiSh [BKV19] using GAIP as the hard relation (for simplicity we consider the base scheme with challenge space $\{0, 1\}$).
- The main technical challenge appears in PreSig algorithm.

Schnorr

procedure PreSig(sk, m, $\boxed{Y}$)
  $r \leftarrow_{\$} \mathbb{Z}_q, R := g^r$
  $e := H(\mathrm{pk} \| \boxed{R \cdot Y} \| m)$
  $\hat{s} := r - e \cdot \mathrm{sk} \bmod q$
  return $\hat{\sigma} := (e, \hat{s})$

IAS

procedure PreSig(sk, m, $\boxed{E_Y}$)
  $r \leftarrow_{\$} \mathrm{Cl}(\mathcal{O}), E_R := [r]E_0$
  $e := H(\mathrm{pk} \| \boxed{E_R \cdot E_Y} \| m)$
  $\hat{s} := r - e \cdot \mathrm{sk} \bmod N$
  return $\hat{\sigma} := (e, \hat{s})$

Isogeny Adaptor Signature (IAS)

- We can construct an adaptor signature from CSI-FiSh [BKV19] using GAIP as the hard relation (for simplicity we consider the base scheme with challenge space $\{0, 1\}$).
- The main technical challenge appears in PreSig algorithm.

Schnorr

procedure PreSig(sk, m, Y)

$r \leftarrow_{\$} \mathbb{Z}_q, R := g^r$

$e := H(\mathrm{pk} \| R \cdot Y \| m)$

$\hat{s} := r - e \cdot \mathrm{sk} \bmod q$

return $\hat{\sigma} := (e, \hat{s})$

IAS

procedure PreSig(sk, m, $\boxed{E_Y}$)

$r \leftarrow_{\$} \mathrm{Cl}(\mathcal{O}), E_R := [r]E_0$

$e := H(\mathrm{pk} \| \boxed{E_R \cdot E_Y} \| m)$

$\hat{s} := r - e \cdot \mathrm{sk} \bmod N$

return $\hat{\sigma} := (e, \hat{s})$

Problem

We cannot combine E_R and E_Y as there is no meaningful operation between two elliptic curves in isogeny-based group action.

Solution

Randomize the statement E_Y with the group action of E_R and prove the relation between E_R and E_Y in ZK (i.e., a DH-tuple proof for isogenies [CS20]).

Solution

Randomize the statement E_Y with the group action of E_R and prove the relation between E_R and E_Y in ZK (i.e., a DH-tuple proof for isogenies [CS20]).

procedure PreSig(sk, m, E_Y)

$r \leftarrow_\$ \mathrm{Cl}(\mathcal{O})$; $E_R := [r]E_0$

$\hat{E}_R := [r]E_Y = [r][y]E_0 = [r+y]E_0$

Set $x := \{r \mid E_R = [r]E_0 \ \wedge\ \hat{E}_R = [r]E_Y\}$

$\pi \leftarrow \mathsf{P}_{\mathsf{NIZK}}(x, r)$

$e := H(\mathsf{pk} \| \hat{E}_R \| m)$

$\hat{s} := r - e \cdot \mathsf{sk} \bmod N$

return $\hat{\sigma} := (e, \hat{s}, \hat{E}_R, \pi)$

Solution

Randomize the statement E_Y with the group action of E_R and prove the relation between E_R and E_Y in ZK (i.e., a DH-tuple proof for isogenies [CS20]).

procedure $\mathsf{PreSig}(\mathsf{sk}, m, E_Y)$

$r \leftarrow_{\$} \mathrm{Cl}(\mathcal{O}); E_R := [r]E_0$

$\boxed{\hat{E}_R := [r]E_Y = [r][y]E_0 = [r+y]E_0}$

Set $x := \{r \mid E_R = [r]E_0 \ \wedge \ \hat{E}_R = [r]E_Y\}$

$\pi \leftarrow \mathsf{P}_{\mathsf{NIZK}}(x, r)$

$e := H(\mathsf{pk} \| \boxed{\hat{E}_R} \| m)$

$\hat{s} := r - e \cdot \mathsf{sk} \bmod N$

return $\hat{\sigma} := (e, \hat{s}, \hat{E}_R, \pi)$

IAS

Solution

Randomize the statement E_Y with the group action of E_R and prove the relation between E_R and E_Y in ZK (i.e., a DH-tuple proof for isogenies [CS20]).

procedure $\mathsf{PreSig}(\mathsf{sk}, m, E_Y)$
$r \leftarrow_{\$} \mathsf{Cl}(\mathcal{O})$; $E_R := [r]E_0$
$\hat{E}_R := [r]E_Y = [r][y]E_0 = [r+y]E_0$
Set $x := \{r \mid E_R = [r]E_0 \wedge \hat{E}_R = [r]E_Y\}$
$\pi \leftarrow \mathsf{P}_{\mathsf{NIZK}}(x, r)$
$e := H(\mathsf{pk}||\hat{E}_R||m)$
$\hat{s} := r - e \cdot \mathsf{sk} \bmod N$
return $\hat{\sigma} := (e, \hat{s}, \hat{E}_R, \pi)$

procedure $\mathsf{PreVer}(\mathsf{pk}, m, E_Y, \hat{\sigma})$
Parse pk as (E_0, E_1)
Parse $\hat{\sigma}$ as $(e, \hat{s}, \hat{E}_R, \pi)$
$E_R := [\hat{s}]\mathsf{pk}_e$
Set $x := \{r \mid E_R = [r]E_0 \wedge \hat{E}_R = [r]E_Y\}$
if $\pi \leftarrow \mathsf{V}_{\mathsf{NIZK}}(x, \pi) \neq 1$ **then**
  return 0
$e' = H(\mathsf{pk}||\hat{E}_R||m)$
return $(e = e')$

Adapt and Ext algorithms are analogous to Schnorr-based adaptor signature construction from [AEE+20].

procedure Adapt($\hat{\sigma}, y$)
  Parse $\hat{\sigma}$ as $(e, \hat{s}, \hat{E}_R, \pi)$
  $s := \hat{s} + y \bmod N$
  return $\sigma := (e, s)$

procedure Ext($\sigma, \hat{\sigma}, E_Y$)
  Parse σ as (e, s) and $\hat{\sigma}$ as $(e, \hat{s}, \hat{E}_R, \pi)$
  $y' := s - \hat{s} \bmod N$
  if $(E_Y, y') \in R$ **return** y'
  else return $\perp$

IAS

The drawbacks of the lattice adaptor signature (LAS) [EEE20] are:

- the extracted witnesses do not guarantee adaptability (i.e., imperfect correctness),
- privacy issues in off-chain applications that require several concatenated instances of pre-signatures of the form

$$\mathsf{PreSig}(\mathsf{sk}_1, m_1, Y_1) \rightarrow \cdots \rightarrow \mathsf{PreSig}(\mathsf{sk}_n, m_n, Y_n),$$

  with interleaved statements Y_i.

IAS

The drawbacks of the lattice adaptor signature (LAS) [EEE20] are:

- the extracted witnesses do not guarantee adaptability (i.e., imperfect correctness),
- privacy issues in off-chain applications that require several concatenated instances of pre-signatures of the form

$$\mathsf{PreSig}(\mathsf{sk}_1, m_1, Y_1) \to \cdots \to \mathsf{PreSig}(\mathsf{sk}_n, m_n, Y_n),$$

  with interleaved statements Y_i.

The culprit in both of these drawbacks is the noisy nature of lattice-based schemes, which causes a knowledge-gap and increases the norm of the vectors. Our isogeny-based construction overcomes these issues due to the underlying **group action** structure.

- C implementation, parallelized with OpenMP, and benchmarked on 2.0GHz AMD EPYC 7702 processor with 16 cores and 32GB RAM (time in seconds, size in bytes)
- Source code: `https://github.com/etairi/Adaptor-CSI-FiSh`

S	t	$\|sk\|$	$\|pk\|$	$\|\hat{\sigma}\|$	$\|\sigma\|$	KGen	Sig	Ver	PreSig	PreVer	Ext	Adapt
2^1	56	16	128	19944	1880	0.05	0.24	0.23	3.59	3.55	0.005	0.005
2^2	38	16	256	19672	1286	0.06	0.16	0.16	2.75	2.68	0.005	0.005
2^3	28	16	512	19020	956	0.07	0.13	0.14	2.21	2.15	0.005	0.005
2^4	23	16	1024	19338	791	0.07	0.11	0.11	1.99	1.94	0.005	0.005
2^6	16	16	4096	18624	560	0.29	0.08	0.09	1.61	1.56	0.005	0.005
2^8	13	16	16384	18330	461	1.00	0.08	0.08	1.50	1.44	0.005	0.005

Performance Evaluation

- C implementation, parallelized with OpenMP, and benchmarked on 2.0GHz AMD EPYC 7702 processor with 16 cores and 32GB RAM (time in seconds, size in bytes)
- Source code: `https://github.com/etairi/Adaptor-CSI-FiSh`

S	t	$\lvert sk \rvert$	$\lvert pk \rvert$	$\lvert \hat{\sigma} \rvert$	$\lvert \sigma \rvert$	KGen	Sig	Ver	PreSig	PreVer	Ext	Adapt
2^1	56	16	128	19944	1880	0.05	0.24	0.23	3.59	3.55	0.005	0.005
2^2	38	16	256	19672	1286	0.06	0.16	0.16	2.75	2.68	0.005	0.005
2^3	28	16	512	19020	956	0.07	0.13	0.14	2.21	2.15	0.005	0.005
2^4	23	16	1024	19338	791	0.07	0.11	0.11	1.99	1.94	0.005	0.005
2^6	16	16	4096	18624	560	0.29	0.08	0.09	1.61	1.56	0.005	0.005
2^8	13	16	16384	18330	461	1.00	0.08	0.08	1.50	1.44	0.005	0.005

- S (public keys) and t (iterations) are inversely related to each other and control the running time of KGen and Sig (along with public key and signature size).

- C implementation, parallelized with OpenMP, and benchmarked on 2.0GHz AMD EPYC 7702 processor with 16 cores and 32GB RAM (time in seconds, size in bytes)
- Source code: `https://github.com/etairi/Adaptor-CSI-FiSh`

S	t	$\|sk\|$	$\|pk\|$	$\|\hat{\sigma}\|$	$\|\sigma\|$	KGen	Sig	Ver	PreSig	PreVer	Ext	Adapt
2^1	56	16	128	19944	1880	0.05	0.24	0.23	3.59	3.55	0.005	0.005
2^2	38	16	256	19672	1286	0.06	0.16	0.16	2.75	2.68	0.005	0.005
2^3	28	16	512	19020	956	0.07	0.13	0.14	2.21	2.15	0.005	0.005
2^4	23	16	1024	19338	791	0.07	0.11	0.11	1.99	1.94	0.005	0.005
2^6	16	16	4096	18624	560	0.29	0.08	0.09	1.61	1.56	0.005	0.005
2^8	13	16	16384	18330	461	1.00	0.08	0.08	1.50	1.44	0.005	0.005

- S (public keys) and t (iterations) are inversely related to each other and control the running time of KGen and Sig (along with public key and signature size).
- The main bottleneck of the construction is the expensive ZK proof used in pre-signatures.

Conclusion

- IAS is an isogeny-based post-quantum adaptor signature based on CSI-FiSh [BKV19] and proven secure in QROM.

- IAS is an isogeny-based post-quantum adaptor signature based on CSI-FiSh [BKV19] and proven secure in QROM.
- Unlike LAS [EEE20], due to a group action structure we can achieve perfect correctness and do not hinder the privacy of the off-chain applications that use adaptor signatures.

Conclusion

- IAS is an isogeny-based post-quantum adaptor signature based on CSI-FiSh [BKV19] and proven secure in QROM.
- Unlike LAS [EEE20], due to a group action structure we can achieve perfect correctness and do not hinder the privacy of the off-chain applications that use adaptor signatures.
- Our technique to construct isogeny-based adaptor signature is generic enough to be applicable to other isogeny-based signature schemes (e.g., SQISign [DFKL+20]).

Conclusion

- IAS is an isogeny-based post-quantum adaptor signature based on CSI-FiSh [BKV19] and proven secure in QROM.
- Unlike LAS [EEE20], due to a group action structure we can achieve perfect correctness and do not hinder the privacy of the off-chain applications that use adaptor signatures.
- Our technique to construct isogeny-based adaptor signature is generic enough to be applicable to other isogeny-based signature schemes (e.g., SQISign [DFKL+20]).
- The future work is to improve the performance and obtain better security estimates [BS20, Pei20].

Thank you!

@erkantairi

References I

[ADFMP20] Navid Alamati, Luca De Feo, Hart Montgomery, and Sikhar Patranabis, Cryptographic group actions and applications, ASIACRYPT, 2020.

[AEE+20] Lukas Aumayr, Oguzhan Ersoy, Andreas Erwig, Sebastian Faust, Kristina Hostakova, Matteo Maffei, Pedro Moreno-Sanchez, and Siavash Riahi, Generalized bitcoin-compatible channels, Cryptology ePrint Archive, Report 2020/476, 2020.

[BKV19] Ward Beullens, Thorsten Kleinjung, and Frederik Vercauteren, Csi-fish: Efficient isogeny based signatures through class group computations, ASIACRYPT, 2019.

[BS20] Xavier Bonnetain and André Schrottenloher, Quantum security analysis of csidh, EUROCRYPT, 2020.

[CLM+18] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes, Csidh: An efficient post-quantum commutative group action, ASIACRYPT, 2018.

References II

[CS20] Daniele Cozzo and Nigel P. Smart, Sashimi: Cutting up csi-fish secret keys to produce an actively secure distributed signing protocol, PQCrypto, 2020.

[DFG19] Luca De Feo and Steven D. Galbraith, Seasign: Compact isogeny signatures from class group actions, EUROCRYPT, 2019.

[DFKL+20] Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski, Sqisign: Compact post-quantum signatures from quaternions and isogenies, ASIACRYPT, 2020.

[EEE20] Muhammed F. Esgin, Oguzhan Ersoy, and Zekeriya Erkin, Post-quantum adaptor signatures and payment channel networks, Cryptology ePrint Archive, Report 2020/845, 2020.

[ENS20] Muhammed F. Esgin, Ngoc Khanh Nguyen, and Gregor Seiler, Practical exact proofs from lattices: New techniques to exploit fully-splitting rings, Cryptology ePrint Archive, Report 2020/518, 2020.

[Kup05] Greg Kuperberg, A subexponential-time quantum algorithm for the dihedral hidden subgroup problem, SIAM J. Comput. **35** (2005), no. 1, 170–188.

[MMS+19] Giulio Malavolta, Pedro Moreno-Sanchez, Clara Schneidewind, Aniket Kate, and Matteo Maffei, Anonymous multi-hop locks for blockchain scalability and interoperability, NDSS, 2019.

[Pei20] Chris Peikert, He gives c-sieves on the csidh, EUROCRYPT, 2020.