

# Another look at some isogeny hardness assumptions

Simon-Philipp Merz<sup>1</sup>, Romy Minko<sup>2</sup> and Christophe Petit<sup>3</sup>

<sup>1</sup> Royal Holloway, University of London

<sup>2</sup> University of Oxford

<sup>3</sup> University of Birmingham

**Abstract.** The security proofs for isogeny-based undeniable signature schemes have been based primarily on the assumptions that the One-Sided Modified SSCDH problem and the One-More SSCDH problem are intractable. We challenge the validity of these assumptions, showing that both the decisional and computational variants of these problems can be solved in polynomial time. We further demonstrate an attack, applicable to two undeniable signature schemes, one of which was proposed at PQCrypto 2014. The attack allows to forge signatures in  $2^{4\lambda/5}$  steps on a classical computer. This is an improvement over the expected classical security of  $2^\lambda$ , where  $\lambda$  denotes the chosen security parameter.

**Keywords:** elliptic curves · isogenies · undeniable signatures

## 1 Introduction

Most currently deployed cryptographic schemes are based on mathematical problems that are assumed to be hard on classical computers, but can be solved in polynomial time using quantum algorithms. Continuous progress in quantum computing therefore requires the development of “post-quantum secure” cryptography relying on problems that will (at least to the best of our knowledge) remain hard for quantum algorithms. To achieve quantum resistance some directions currently being explored include lattice-based, multivariate, code-based, and hash-based cryptography and, most recently, cryptography based on isogeny problems. While the latter is appealing for relatively small key sizes compared to other candidates, it requires further optimization and scrutiny.

Isogeny-based cryptography was first proposed by Couveignes in 1997 in a seminar at the ENS [7], but he did not publish his ideas at the time. Almost a decade later Rostovtsev and Stolbunov rediscovered and further developed the same idea independently [20]. While these cryptosystems were based on “ordinary curves”, “supersingular curves” were first put to use in the construction of a hash function by Charles, Goren and Lauter [4]. Jao and De Feo introduced another cryptosystem in the supersingular case, the so called Supersingular Isogeny Diffie-Hellman (SIDH) [11]. Instead of using the action of the class group on certain isomorphism classes of ordinary elliptic curves like Couveignes, Rostovtsev

and Stolbunov, SIDH relies on the simple observation that it does not matter in which order we divide out two non-intersecting subgroups of an elliptic curve. One promising submission to NIST’s post-quantum standardization project [18] is the SIDH-based key exchange protocol called SIKE [1].

For a nice introduction to different computational problems in supersingular isogeny-based cryptography we refer to Galbraith and Vercauteren [10]. The template for isogeny-based cryptography is the general isogeny problem. That is, to find an isogeny  $\phi : E_1 \rightarrow E_2$ , for two randomly chosen isogenous curves  $E_1$  and  $E_2$ . A variant of this problem includes the additional information of the degree of  $\phi$ . This reduces the problem space from an infinite to a finite number of isogenies while simultaneously reducing the search space. Hence, it is not clear whether it makes the problem harder or easier. Another related problem is the computation of endomorphism rings of supersingular elliptic curves. Assume you know the endomorphism ring of a supersingular curve  $E_1$  and you want to compute the endomorphism ring of  $E_2$ . This is computationally broadly equivalent to computing an isogeny  $\phi : E_1 \rightarrow E_2$  [15, 16].

However, more practical supersingular isogeny constructions give more information to a potential attacker. For example, the SIDH protocol, which we will describe in Section 3 in more detail, reveals the image of certain torsion points under some secret isogenies in addition to the origin and image curves. It was observed that this additional information might make the problem *a priori* easier and a framework for a potential attack under additional assumptions was given by Petit [19].

Various other versions of isogeny problems have been suggested and conjectured to be hard by other authors to provide security proofs for their cryptographic constructions.

**Our contribution:** In this work, we will review some of the isogeny problems that have been suggested in the construction of isogeny-based undeniable signatures [12] published at PQCrypto 2014. While this construction has been used and extended by other authors [22], we show that the assumptions used to make the security proofs work are not valid and the proposed isogeny problems lack the conjectured hardness. This does not immediately lead to an attack on the signature scheme itself. However, we propose an attack on the cryptographic construction as well.

**Outline:** In Section 2 we recall some mathematical background on isogeny-based cryptography. In Section 3 we give the definitions of some isogeny problems that have been used in the literature and we give an attack on two of them. The following Section 4 describes how the problems have been used in the construction of isogeny-based undeniable signatures of [12]. We provide an attack on the signature scheme combining a near-collision search in the hash function and the attack on the underlying isogeny problem. Before concluding the paper, we mention other constructions that are affected by our attacks in Section 5.

## 2 Mathematical background

For a full treatment of background information on elliptic curves and a detailed introduction to isogeny-based cryptography we refer to Silverman [21] and De Feo [9], respectively.

Let  $\mathbb{F}_q$  be a finite field of characteristic  $p$ . In the following we assume  $p \geq 3$  and therefore an elliptic curve  $E$  over  $\mathbb{F}_q$  can be defined by its short Weierstrass form

$$E(\mathbb{F}_q) = \{(x, y) \in \mathbb{F}_q^2 \mid y^2 = x^3 + Ax + B\} \cup \{\mathcal{O}_E\}$$

where  $A, B \in \mathbb{F}_q$  and  $\mathcal{O}_E$  is the point  $(X : Y : Z) = (0 : 1 : 0)$  on the projective curve  $Y^2Z = X^3 + AXZ^2 + BZ^3$ . The set of points on an elliptic curve is an abelian group under the ‘‘chord and tangent rule’’ with  $\mathcal{O}_E$  being the identity element. The number of points on an elliptic curve is  $\#E(\mathbb{F}_q) = q + 1 - t$  for some integer  $t \leq 2\sqrt{q}$ . A curve  $E$  is called *supersingular* if  $p|t$  and *ordinary* otherwise. The *j-invariant* of an elliptic curve is

$$j(E) = 1728 \frac{4A^3}{4A^3 + 27B^2}$$

and there is an isomorphism  $f : E \rightarrow E'$  if and only if  $j(E) = j(E')$ .

Given two elliptic curves  $E_1$  and  $E_2$  over a finite field  $\mathbb{F}_q$ , an *isogeny* is a morphism  $\phi : E_1 \rightarrow E_2$  such that  $\phi(\mathcal{O}_{E_1}) = \mathcal{O}_{E_2}$ . One can show that isogenies are morphisms both in the sense of algebraic geometry and group theory. If there exists a non-constant isogeny between them, two curves are called *isogenous*. The *degree* of an isogeny  $\phi$  is its degree when treated as an algebraic map. This is equal to the size of the kernel of  $\phi$  if the isogeny is separable (which is always the case in this work).

Since an isogeny defines a group homomorphism  $E_1 \rightarrow E_2$ , its kernel is a subgroup of  $E_1$ . Conversely, any subgroup  $S \subset E_1$  determines a (separable) isogeny  $\phi : E_1 \rightarrow E_2$  with  $\ker(\phi) = S$  and  $E_2 = E_1/S$ . Since all isogenies in the following will have cyclic groups as kernels, knowledge of the isogeny and knowledge of the kernel of the isogeny are equivalent.

A basic example of an isogeny is the multiplication by  $n$  map on an elliptic curve  $[n] : E \rightarrow E$ . The kernel of the multiplication by  $n$  map over the algebraic closure  $\overline{\mathbb{F}_q}$  of  $\mathbb{F}_q$  is the  $n$ -torsion subgroup

$$E[n] = \{P \in E(\overline{\mathbb{F}_q}) : [n]P = \mathcal{O}_E\}.$$

Whenever  $n$  and  $q$  are relatively prime, the group  $E[n]$  is isomorphic to  $(\mathbb{Z}/n\mathbb{Z})^2$ .

Given any isogeny  $\phi : E_1 \rightarrow E_2$ , there exists another isogeny  $\hat{\phi}$ , called the *dual isogeny*, satisfying  $\phi \circ \hat{\phi} = \hat{\phi} \circ \phi = [\deg(\phi)]$ .

## 3 The one-more isogeny problem

We begin this section by recalling the SIDH protocol and a problem underlying its security. Then, we define and illustrate the somewhat more artificial isogeny

problems that were conjectured to be hard and that are used in the security proofs of [12, 22]. However, at the end of this section we present our polynomial time attack against these more artificial problems and show that no confidence in them is justified.

### 3.1 Problem statements

Even though we do not attack SIDH, it is useful to recall this fundamental key exchange protocol as it contains some ideas upon which the undeniable signature schemes we cryptanalyze are based.

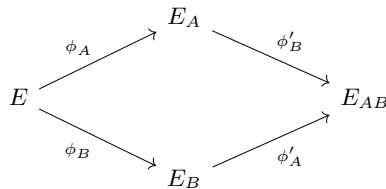
Let  $p$  be a prime of the form  $\ell_A^{e_A} \ell_B^{e_B} \cdot f \pm 1$  where  $\ell_A$  and  $\ell_B$  are small distinct primes,  $e_A$  and  $e_B$  are positive integers and  $f$  is some (usually small) cofactor. Moreover, we fix a supersingular elliptic curve  $E$  defined over  $\mathbb{F}_{p^2}$  together with bases  $\{P_A, Q_A\}$ ,  $\{P_B, Q_B\}$  of the  $\ell_A^{e_A}$  and  $\ell_B^{e_B}$  torsion of  $E$ ,  $E[\ell_A^{e_A}]$  and  $E[\ell_B^{e_B}]$ , respectively.

Suppose Alice and Bob wish to establish a shared secret. Alice's secret is an integer  $a \in \{0, \dots, \ell_A^{e_A} - 1\}$ , defining the subgroup  $A := \langle P_A + [a]Q_A \rangle$  of  $E[\ell_A^{e_A}]$ . Her public key is the curve  $E_A := E/A$  together with the images  $\phi_A(P_B), \phi_A(Q_B)$  of Bob's public basis under her secret isogeny  $\phi_A : E \rightarrow E/A$ . Analogously, Bob chooses his secret key  $b \in \{0, \dots, \ell_B^{e_B} - 1\}$  defining the cyclic subgroup  $B := \langle P_B + [b]Q_B \rangle \subset E[\ell_B^{e_B}]$ , and his public key is the tuple  $(E_B, \phi_B(P_A), \phi_B(Q_A))$ .

The key exchange proceeds as follows: Upon receipt of Bob's public key, Alice uses the points to push her secret  $A \subset E[\ell_A^{e_A}]$  to  $E/B$ , i.e. Alice computes an isogeny  $\phi'_A : E_B \rightarrow E_{AB}$  with kernel  $\langle \phi_B(P_A) + [a]\phi_B(Q_A) \rangle \subset E/B[\ell_A^{e_A}]$ . Bob proceeds *mutatis mutandis*. We have

$$E_{AB} = \phi'_A(\phi_B(E)) = \phi'_B(\phi_A(E)) = E/\langle P_A + [a]Q_A, P_B + [b]Q_B \rangle,$$

where the equality holds up to isomorphism. Since the  $j$ -invariant is the same for all curves in one isomorphism class, both Alice and Bob can compute the shared secret  $j(E_{AB})$ .



**Fig. 1.** The commutative diagram of the SIDH key exchange

The hardness of the following problem underlies the security of the SIDH protocol.

**Definition 1 (Supersingular Computational Diffie-Hellman (SSCDH) Problem).** Let  $m_A, n_A$  be chosen at random from  $\{0, \dots, \ell_A^{e_A} - 1\}$  not both divisible by  $\ell_A$ . Analogously, let  $m_B, n_B$  be randomly chosen from  $\{0, \dots, \ell_B^{e_B} - 1\}$  not both divisible by  $\ell_B$ . Furthermore, let  $\phi_A : E \rightarrow E_A$  and  $\phi_B : E \rightarrow E_B$  denote the isogenies with kernel  $\langle [m_A]P_A + [n_A]Q_A \rangle$  and  $\langle [m_B]P_B + [n_B]Q_B \rangle$  respectively.

Given the curves  $E_A, E_B$  and the points  $\phi_A(P_B), \phi_A(Q_B), \phi_B(P_A)$  and  $\phi_B(Q_A)$ , find the  $j$ -invariant of

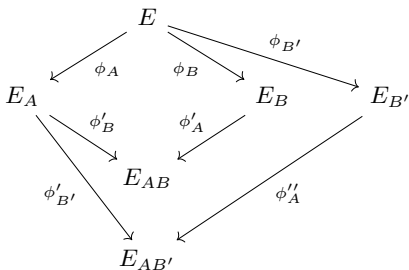
$$E_{AB} = E / \langle [m_A]P_A + [n_A]Q_A, [m_B]P_B + [n_B]Q_B \rangle.$$

For the following, we fix the notation of Definition 1.

**Definition 2 (Modified SSCDH (MSSCDH) Problem).** [12] Given  $E_A, E_B$  and  $\ker(\phi_B)$ , determine  $E_{AB}$  up to isomorphism, i.e.  $j(E_{AB})$ .

Note that knowledge of  $\ker(\phi_B)$  is equivalent to knowledge of  $\phi_B$ , but the lack of information on the auxiliary points in the image of  $\phi_A$  in the MSSCDH problem prevents to *shift*  $\ker(\phi_B)$  into  $E_A$ .

**Definition 3 (One-sided Modified SSCDH (OMSSCDH) Problem).** [12] For fixed  $E_A, E_B$ , given an oracle to solve MSSCDH for any  $E_A, E_{B'}, \ker(\phi_{B'})$  with  $E_{B'}$  not isomorphic to  $E_B$  and  $\ell_B^{e_B}$ -isogenous to  $E$ , solve MSSCDH for  $E_A, E_B$  and  $\ker(\phi_B)$ .



**Fig. 2.** The oracle provides  $E_{AB'}$  for any  $E_{B'}$  and  $\phi_{B'}$ , while  $E_{AB}$  needs to be found in OMSSCDH

While the OMSSCDH assumption seems somewhat more artificial, it arises naturally in the security analysis of undeniable signatures proposed in [12]. Moreover, the authors proposing the problem conjectured it to be computationally infeasible, in the sense that for any polynomial-time solver algorithm, the advantage of the algorithm is a negligible function in the security parameter  $\log p$ . However, we will see in the next subsection that a polynomial time attacker will have a non-negligible advantage to solve the OMSSCDH problem.

A decisional variant of this problem is also defined in [12]; our attack will apply to it in the obvious way as well.

Our results furthermore break other strongly related problems, such as the following slightly weaker problem used in the construction of undeniable blind signatures [22].

**Definition 4 (One-More SSCDH (1MSSCDH) Problem).** *Let  $E$  be some base curve of the form as in the SIDH protocol and let  $m_A, n_A$  be secret integers in  $\{0, \dots, \ell_A^{e_A} - 1\}$ .*

*Let a signing oracle respond  $E_{AB} \cong E_B / \langle [m_A]P_B + [n_A]Q_B \rangle$  upon receipt of a curve  $E_B$  isogenous to  $E$  and points  $P_B, Q_B$  spanning  $E_B[\ell_B^{e_B}]$ .*

*The 1MSSCDH problem is to produce at least  $q + 1$  distinct pairs of curves  $(E_{B_i}, E_{AB_i})$ , where  $E_{B_i}$  are  $\ell_B^{e_B}$ -isogenous to  $E$ ,  $P_{B_i}$  and  $Q_{B_i}$  span  $E_{B_i}[\ell_B^{e_B}]$  and  $E_{AB_i}$  is isomorphic to  $E_{B_i} / \langle [m_A]P_{B_i} + [n_A]Q_{B_i} \rangle$  for  $1 \leq i \leq q + 1$ , after  $q$  queries to the signing oracle.*

Compared to the OMSSCDH problem it leaves the choice of the additional MSSCDH instance which needs to be solved to the attacker. Both problems are somewhat natural variants of the SSCDH problem underlying the security of SIDH. However, variants of computational problems used in cryptography are not always as hard as the original problems themselves [13, 14].

### 3.2 Basic attack

Now, we describe our attacks on the OMSSCDH and 1MSSCDH problems.

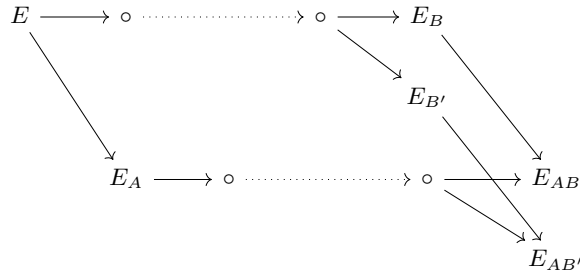
**Theorem 1.** *A solution to the OMSSCDH problem (Definition 3) can be guessed with probability  $\frac{1}{(\ell_B + 1)\ell_B}$  after a single query to the signing oracle.*

*Proof.* Assume an attacker wants to solve OMSSCDH given  $E_A, E_B$  and  $\ker(\phi_B)$ . Let  $E_{B'}$  be another curve  $\ell_B^2$ -isogenous to  $E_B$  and  $\ell_B^{e_B}$ -isogenous to  $E$ . That is, one gets from  $E_B$  to  $E_{B'}$  via backtracking the last  $\ell_B$ -isogeny step of  $\phi_B$ . Note, one could guess such an  $E_{B'}$  with probability  $\frac{\ell_B - 1}{(\ell_B + 1)\ell_B}$  even without knowing  $\phi_B$ .

Then, the attacker can query the oracle on  $E_{B'}$  to receive  $E_{AB'}$ . Now, any curve in the isomorphism class of  $E_{AB}$  is  $\ell_B^2$ -isogenous to  $E_{AB'}$  as depicted in Figure 3. Therefore an attacker can guess the isomorphism class of  $E_{AB}$  correctly with probability  $((\ell_B + 1)\ell_B)^{-1}$  finishing the proof.

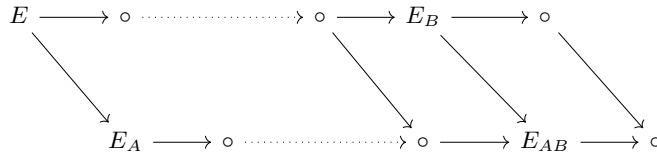
In practice the prime  $\ell_B$  is chosen to be small (usually 2 or 3) and thus Theorem 1 breaks the OMSSCDH problem completely.

*Remark 1.* Without the condition on the degree of the isogeny between the curves submitted to the OMSSCDH oracle and the base curve, the attack can be made even more efficient. Namely, an attacker can always solve this modified version of the OMSSCDH problem after two queries to the oracle as follows.



**Fig. 3.** Query of OMSSCDH oracle on  $\ell_B^2$ -isogenous curve via backtracking one step yields elliptic curve close to target curve

The attacker computes two curves  $E_{B_1}, E_{B_2}$  of different isomorphism classes that are  $\ell_B$ -isogenous to  $E_B$ . Knowing  $\ker(\phi_B)$  the attacker can compute  $\ker(\phi_{B_i})$  and they can query the oracle to solve MSSCDH for  $E_A, E_{B_i}$  and  $\ker(\phi_{B_i})$  for  $i = 1, 2$ . The oracle sends back  $E_{AB_i}$  which are  $\ell_B$ -isogenous to the unknown  $E_{AB}$  as shown in Figure 4. Listing all  $\ell_B + 1$  isomorphism classes which are  $\ell_B$ -isogenous to  $E_{AB_1}$  and  $E_{AB_2}$  respectively, we find the isomorphism class of  $E_{AB}$  as it is the only one appearing in both lists.



**Fig. 4.** Diagonal maps are the signing oracle sending  $\ell_B$ -isogenous curves of  $E_B$  to  $\ell_B$ -isogenous curves of target curve  $E_{AB}$

Clearly, the attack described in Theorem 1 can be generalised to OMSSDDH, the decisional variant of OMSSCDH. Furthermore, a solution to the OMSSCDH problem implies a solution to the 1MSSCDH problem which yields the following theorem.

**Theorem 2.** *A solution to the 1MSSCDH problem (Definition 4) can be guessed with probability  $\frac{1}{(\ell_B+1)\ell_B}$  after a single query to the signing oracle.*

### 4 Application to Jao-Soukharev’s construction

We now describe the application of our attack against Jao-Soukharev’s undeniable signature scheme [12]. For background knowledge on undeniable signature schemes we refer the reader to [5, 8, 17].

#### 4.1 Jao-Soukharev undeniable signatures

An undeniable signature scheme is a scheme in which signatures can only be verified with cooperation from the signer [5]. Upon receipt of a signature  $\sigma$  from a verifier, the signer engages in a zero-knowledge confirmation (or disavowal) protocol to prove the validity (or invalidity) of  $\sigma$ . The security properties required by an undeniable signature scheme are undeniability, unforgeability and invisibility. Undeniability ensures that a signer cannot repudiate a valid signature. Unforgeability is the notion that an adversary cannot compute a valid message-signature pair without knowledge of the signer's secret key. Invisibility requires that an adversary cannot distinguish between a valid signature and a signature produced by a simulator with non-negligible probability. We refer to Appendix A for a full definition of all security games for undeniable signature schemes.

The Jao-Soukharev protocol takes  $p$  as a prime of the form  $\ell_A^{e_A} \ell_B^{e_B} \ell_C^{e_C} \cdot f \pm 1$ . We fix a supersingular curve  $E$  over  $\mathbb{F}_{p^2}$  and bases  $\{P_A, Q_A\}$ ,  $\{P_B, Q_B\}$  and  $\{P_C, Q_C\}$  of the  $\ell_A^{e_A}$ ,  $\ell_B^{e_B}$  and  $\ell_C^{e_C}$  torsion of  $E$ ,  $E[\ell_A^{e_A}]$ ,  $E[\ell_B^{e_B}]$  and  $E[\ell_C^{e_C}]$ , respectively. The public parameters of the scheme are  $p$ ,  $E$  and the three torsion bases, together with a hash function  $H$ . The signer generates random integers  $m_A, n_A \in \mathbb{Z}/\ell_A^{e_A}$  and computes the isogeny  $\phi_A : E \rightarrow E_A$ , defined as in Problem 3.1. The public key consists of the curve  $E_A$  together with the points  $\{\phi_A(P_C), \phi_A(Q_C)\}$  and the integers  $m_A, n_A$  constitute the private key. Note that this is equivalent to taking  $\phi_A$  as the private key.

To sign a message  $M$ , the signer computes the hash  $h = H(M)$  of the message and the isogenies

$$\begin{aligned}\phi_B : E &\rightarrow E_B = E/\langle P_B + [h]Q_B \rangle \\ \phi_{AB} : E_A &\rightarrow E_{AB} = E_A/\langle \phi_A(P_B + [h]Q_B) \rangle \\ \phi_{BA} : E_B &\rightarrow E_{AB} = E_B/\langle \phi_B([m_A]P_A + [n_A]Q_A) \rangle.\end{aligned}$$

The signer then outputs  $E_{AB}$  in addition to the set of two auxiliary points  $\{\phi_{BA}(\phi_B(P_C)), \phi_{BA}(\phi_B(Q_C))\}$  as the signature.

Given a signature  $\sigma = (E, P, Q)$ , the first step in the confirmation and disavowal protocols is for the signer to select  $m_C, n_C \in \mathbb{Z}/\ell_C^{e_C}\mathbb{Z}$  and compute the curves  $E_C = E/\langle [m_C]P_C + [n_C]Q_C \rangle$ ,  $E_{BC} = E_B/\langle \phi_B([m_C]P_C + [n_C]Q_C) \rangle$ ,  $E_{AC} = E_A/\langle \phi_A([m_C]P_C + [n_C]Q_C) \rangle$  and  $E_{ABC} = E_{BC}/\langle \phi_B([m_A]P_A + [n_A]Q_A) \rangle$ . The signer outputs these curves and  $\ker(\phi_{CB})$  as the commitment, where  $\phi_{CB}$  is the isogeny from  $E_C$  to  $E_{BC}$ . In addition to the auxiliary points of the signature, this commitment gives the verifier enough information to compute  $E_{ABC}$  and  $E_{\sigma C} = E_\sigma/\langle [m_C]P + [n_C]Q \rangle$ , to check whether  $E_{\sigma C} = E_{ABC}$ . Further details of the confirmation and disavowal protocols can be found in [12].

In the Jao-Soukharev construction, the adversary knows  $E_A$  and can compute  $E_{B_i}$  and  $\ker(\phi_{B_i})$ , corresponding to message  $M_i$ , from  $H$ . The signing oracle then



essentially solves MSSCDH for any of the adversary’s input messages  $M_i$ . The paper claims that under the assumption that the confirmation and disavowal protocols of the signature scheme are zero-knowledge, the unforgeability game describes the OMSSCDH problem. We will show that this claim is incorrect.

#### 4.2 Another look at the security proof of [12]

In [12] the claim is made that forging a signature for this construction is equivalent to solving OMSSCDH, so one would expect our attack to directly break unforgeability. However, equivalence would only be true if an attacker had the freedom to submit arbitrary curves to the signing oracle. In the protocol, an adversary wishing to forge a signature can only query the signing oracle with messages,  $M_i$ . In the Jao-Soukharev signing protocol the curves  $E_{B_i}$  are computed from message hashes, rather than the messages themselves. Thus, an adversary would need to find a message mapping to some specific curve first for the scheme to be equivalent to OMSSCDH and thus an adversary would need to break the hash function. Forging messages seems therefore actually harder than breaking OMSSCDH.

As a consequence the attack of Section 3 applies to the hardness assumption but not the actual protocol in [12]. However, in this section we will demonstrate how a hybrid version of our attack on OMSSCDH and finding “near-collisions” in the hash function allows to reduce the security of the construction for the given parameters.

In accounting for the scheme’s loss of malleability due to the hash function we make use of the following Lemma.

**Lemma 1.** *Let  $E$  be a supersingular elliptic curve, let  $\ell$  be a prime, let  $e$  be an integer, and let  $\{P, Q\}$  be a basis for  $E[\ell^e]$ . Let  $n, m < \ell^e$  be positive integers congruent modulo  $\ell^k$  for some integer  $k < e$ . Then the  $\ell$ -isogeny paths from  $E$  to  $E_A = E/\langle P + [n]Q \rangle$  and  $E_B = E/\langle P + [m]Q \rangle$  are equal up to the  $k$ -th step.*

*Proof.* Let  $m = n + \alpha\ell^k$ , for some  $\alpha > 0$ . Let  $\phi_A : E \rightarrow E_A$  be a separable, cyclic isogeny of  $\deg(\phi_A) = \ell^e$  and  $\ker(\phi_A) = \langle P + [n]Q \rangle$ . We can express  $\phi_A$  as the composition of  $e$   $\ell$ -isogenies such that  $\phi_A = \phi_1^A \circ \dots \circ \phi_e^A$ . Likewise,  $\phi_B : E \rightarrow E_B$  can be expressed as  $\phi_B = \phi_1^B \circ \dots \circ \phi_e^B$ . The single  $\ell$ -isogenies correspond to the single steps in the  $\ell$ -isogeny graph. We will show that  $\phi_i^A = \phi_i^B$  for  $1 \leq i \leq k$ .

For  $i = 1, \dots, e$ , let  $\phi_i^A : E_{i-1} \rightarrow E_i$  be an isogeny with kernel  $\langle \ell^{e-i} S_{i-1}^A \rangle$ , where  $E_0 = E$ ,  $S_0^A = P + [n]Q$  and  $S_{i-1}^A = \phi_{i-1}^A(S_{i-2}^A)$ . Define the  $\phi_i^B$  similarly, with  $B$  substituted for  $A$  and  $m$  for  $n$ . A proof can be found in [6] that these are  $\ell$ -isogenies and that  $\phi_1^A \circ \dots \circ \phi_e^A = \phi_A$  up to composition with an automorphism on  $E_A$  (similarly for  $\phi_B$ ). We also have the recursion

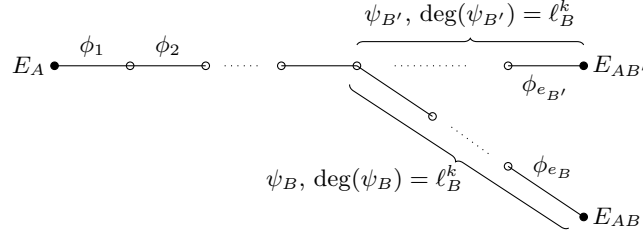
$$\ell^{e-i} S_{i-1}^A = \ell^{e-i} \phi_{i-1}^A(S_{i-2}^A) = \phi_{i-1}^A \circ \dots \circ \phi_1^A(\ell^{e-i} S_0^A)$$

with the analogous result for  $\ell^{e-i}S_{i-1}^B$ . For  $1 \leq i \leq k$ , we have  $e - i + k \geq e$  and so

$$\begin{aligned} \ell^{e-i}S_0^B &= \ell^{e-i}(P + [m]Q) \\ &= \ell^{e-i}(P + [n]Q) + \ell^{e-i+k}[\alpha]Q \\ &= \ell^{e-i}(P + [n]Q) \\ &= \ell^{e-i}S_0^A \end{aligned}$$

using that isogenies are group homomorphisms and  $Q \in E[\ell^e]$ . It follows that  $\phi_i^A = \phi_i^B$  for  $1 \leq i \leq k$ .

Let  $M$  be the message upon which the adversary wishes to forge a signature. Let  $H : \{0, 1\}^* \rightarrow \mathbb{Z}$  be the public hash function used in the signature scheme. The hash function determines a coefficient of a point in the  $E[\ell_i^{e_i}]$  torsion group and can therefore be treated as a function to a group of size  $2^{2\lambda}$  for classical security levels and  $2^{3\lambda}$  for quantum security levels. Let  $2^L$  denote the size of this group in the image.



**Fig. 5.** Isogeny paths between  $E_A$ ,  $E_{AB}$  and  $E_{AB'}$ . In our attack we use  $\psi = \psi_B \circ \hat{\psi}_{B'}$  and have  $\phi_{AB'} = \phi_{e_{B'}} \circ \phi_{e_{B'}-1} \circ \dots \circ \phi_1$ .

The attack proceeds as follows:

1. Build a near-collision on  $H$  with respect to the  $\ell_B$ -adic metric. More precisely, find two messages  $M$  and  $M'$  such that the difference between  $H(M)$  and  $H(M')$  is divisible by a large power of  $\ell_B$ , say a power of size roughly  $2^{L_1}$ .
2. Submit  $M'$  to the signing oracle to obtain the signature

$$\sigma' = (E_{AB'}, P_1 := \phi_{B'A}(\phi_{B'}(P_C)), P_2 := \phi_{B'A}(\phi_{B'}(Q_C))).$$

3. Guess the  $\ell_B^{2k}$ -isogeny  $\psi : E_{AB'} \rightarrow E_{AB}$ , where  $E_{AB}$  is the unknown curve corresponding to  $M$ . Let  $\psi = \psi_B \circ \hat{\psi}_{B'}$ , the composition of two degree  $\ell_B^k \approx 2^{L_2}$  isogenies with  $L_2 = L - L_1$ , where  $\hat{\psi}_{B'}$  corresponds to  $k$  backwards steps

on the isogeny path from  $E_{AB'}$  and  $\psi_B$  corresponds to  $k$  forward steps to  $E_{AB}$ . This is illustrated in Figure 5. The probability of correctly identifying  $\psi$  in a single guess is  $\frac{1}{(\ell_B+1)\ell_B^{2k-1}}$ .

4. Find  $s$  such that  $s\ell_B^k \equiv 1 \pmod{\ell_C^{e_C}}$ . Compute the auxiliary points of the signature as  $\{[s] \cdot \psi(P_1), [s] \cdot \psi(P_2)\}$ .
5. Output  $\sigma = (E_{AB}, [s] \cdot \psi(P_1), [s] \cdot \psi(P_2))$ .

**Theorem 3.** *Let  $s, \psi, P_1$  and  $P_2$  be defined as in our attack. Moreover, let  $\sigma$  be the signature  $(E_{AB}, [s] \cdot \psi(P_1), [s] \cdot \psi(P_2))$  computed in the attack. Assuming that  $E_{AB}$  is guessed correctly,  $\sigma$  is a valid signature.*

*Proof.* Since  $\psi$  takes points on  $E_{AB'}$  to points on  $E_{AB}$ , we have that  $\psi(P_1), \psi(P_2)$  both lie on the target curve. Moreover, as  $\psi(P_1) = \psi(\phi_{B'A}(\phi_{B'}(P_C)))$ , the point lies in the  $\ell_C^{e_C}$  torsion of  $E_{AB}$ ,  $E_{AB}[\ell_C^{e_C}]$ . The same holds for  $\psi(P_2)$ . Although these points would already pass the validation for the signature scheme, they can be easily distinguished from the honestly generated points. The factor  $[s]$  in our signature ensures that forged and honest signatures are identically distributed as described in the following.

Recall that  $\psi = \psi_B \circ \hat{\psi}_{B'}$  and  $P_1 = \phi_{B'A}(\phi_{B'}(P_C))$ . Since the order of  $P_C$  is coprime to  $\deg(\phi_{B'A})$  and  $\deg(\phi_{B'})$ , and the isogeny diagram is commutative, we can write  $P_1 = \phi_{AB'}(\phi_A(P_C))$ .

By expanding  $\phi_{AB'}$  we obtain

$$\begin{aligned} \hat{\psi}_{B'} \circ \phi_{AB'} &= \hat{\phi}_{e_{B'}-k} \circ \cdots \circ \hat{\phi}_{e_{B'}} \circ \phi_{e_{B'}} \circ \cdots \circ \phi_{e_{B'}-k} \circ \cdots \circ \phi_{e_B-k} \circ \cdots \circ \phi_1 \\ &= [\ell_B^k] \circ \phi_{e_{B'}-k-1} \circ \cdots \circ \phi_1. \end{aligned}$$

Since  $s$  is the multiplicative inverse of  $\ell_B^k$  modulo  $\ell_C^{e_C}$ , we have

$$[s] \cdot \psi(P_1) = \phi_{AB}(\phi_A(P_C)) \in E_{AB}[\ell_C^{e_C}].$$

Analogously, we have  $[s] \cdot \psi(P_2) = \phi_{AB}(\phi_A(Q_C)) \in E_{AB}[\ell_C^{e_C}]$ .

Let  $P = \phi_{BA}(\phi_B(P_C)) \in E_{AB}[\ell_C^{e_C}]$  and  $Q = \phi_{BA}(\phi_B(Q_C)) \in E_{AB}[\ell_C^{e_C}]$ . These are the points we expect in an honest signature. In both the confirmation and disavowal protocols of the Jao-Soukharev scheme, the verifier uses the auxiliary points to compute an isogeny from  $E_{AB}$  to a curve  $E_\sigma$  defined as  $E_{AB}/\langle [m_C \cdot s]\psi(P_1) + [n_C \cdot s]\psi(P_2) \rangle$ , where  $m_C, n_C \in \mathbb{Z}/\ell_C^{e_C}\mathbb{Z}$  are integers chosen by the signer. This curve is checked against  $E_{ABC} = E_{AB}/\langle [m_C]P + [n_C]Q \rangle$  to determine the validity of  $\sigma$ . The two points obtained in our attack span the subgroup  $E_{AB}[\ell_C^{e_C}]$ , and we have  $E_{AB}$  as the correct signature curve, so it follows that  $E_\sigma = E_{ABC}$  up to isomorphism and thus the signature is accepted as valid.

Finding a near-collision of  $L_1$  bits on  $H$  classically has cost  $2^{L_1/2}$ . In Step 3 we can then guess the correct isogeny and curve  $E_{AB}$  with probability approximately

$2^{-2L_2} = 2^{-2(L-L_1)}$ . Taking  $L_1 = 4L/5$  the attack then has a total classical cost of  $2^{2L/5}$ , as opposed to the expected  $2^{L/2}$ .

Assuming that we can find (near)-collisions of the hash function with lower quantum complexity [3], the first step of our attack costs  $2^{L_1/3}$  on a quantum computer. Taking  $L_1 = 6L/7$ , this could lower the complexity on a quantum computer to  $2^{2L/7}$ , as opposed to the expected  $2^{L/3}$ . However, it has been argued that quantum collision search might be inferior to classical collision search because of the expensive memory access and quantum memory. For a general discussion on the impracticality of known quantum algorithms for collision search we refer to Bernstein [2].

Clearly, our attack breaks the unforgeability property of the scheme. Moreover, we are also able to break invisibility, since any adversary with the ability to forge signatures with higher probability can simply check whether the challenge signature obtained in the invisibility game (see Appendix A) matches a potential forgery.

## 5 Srinath and Chandrasekaran undeniable blind signatures

Srinath and Chandrasekaran [22] extend the Jao-Soukharev construction to an undeniable *blind* signature scheme, introducing a third actor, the requestor, to the scheme. It is a four-prime variant of the original scheme, taking the prime  $p$  to be of the form  $\ell_A^{e_A} \ell_B^{e_B} \ell_C^{e_C} \ell_D^{e_D} \cdot f \pm 1$  and adding the public parameter  $\{P_D, Q_D\}$ , a basis for  $E[\ell_D^{e_D}]$ . The requestor computes the message curve  $E_B = E/\langle P_B + [H(m)]Q_B \rangle$  using the public hash function, as before. They then blind the curve by taking a random integer  $0 < d < \ell_D^{e_D}$  to compute  $E_{BD} = E_B/\langle \phi_B(P_D) + [d]\phi_B(Q_D) \rangle$ . The blinded curve is then sent to the signer. The **Sign** algorithm of the scheme functions in the same way as for the Jao-Soukharev construction. Upon receipt of the blinded signature curve  $E_{BDA}$ , the requestor uses an unblinding algorithm to obtain the unblinded signature  $E_{BA}$ . The resulting signature is the same as the Jao-Soukharev signature. Thus, signatures as in Srinath and Chandrasekaran are just Jao-Soukharev signatures shifted through another coprime isogeny graph and the scheme is vulnerable to our attack. As before, both unforgeability and invisibility can be broken.

## 6 Conclusion

In this paper, we investigate the hardness of some isogeny problems used in cryptography. In particular, we show that the OMSSCDH and 1MSSCDH problems can be solved with non-negligible probability by a polynomial time attacker. This contribution is particularly relevant to isogeny-based undeniable signature schemes, as the security proofs for unforgeability and invisibility of currently known schemes assume the hardness of these problems. We give basic attacks

against both OMSSCDH and 1MSSCDH, which are also applicable to their decisional variants.

Jao and Soukharev [12] proposed the first quantum-resistant undeniable isogeny-based signature scheme, which was extended to include blindness by Srinath and Chandrasekaran [22]. We present an attack against the unforgeability and invisibility properties of the Jao-Soukharev protocol, showing that an adversary with access to a signing oracle is able to forge arbitrary signatures at lower cost than expected for a given security parameter,  $\lambda$ . To summarise, this is achieved by computing a near-collision on the public hash function  $H$  and guessing an  $\ell_B^{2k}$ -isogeny between an honest signature produced by the oracle for one message to the target forgery curve. The classical cost for this attack is  $2^{4\lambda/5}$ , with the hash function length equal to  $2\lambda$ . We postulate that the quantum cost for this attack is  $2^{6\lambda/7}$ . These attacks imply that parameters should now be increased by 25% to achieve the same classical security level (17% for quantum security). Furthermore, we argue that the equivalence drawn in [12] between unforgeability and the OMSSCDH problem is incorrect, and hence that the security proofs in this paper are incorrect. We note that the inclusion of a hash function increases the difficulty of forgery, assuming the hash function is ‘cryptographically secure’, as the adversary is forced to search for a message that will result in a specific curve, rather than querying the oracle indiscriminately.

Finally, we review the Srinath-Chandrasekan signature scheme and show that our attack is applicable against it. We also note the same problem with the security proofs.

**Acknowledgements.** We thank David Jao for his comments on a preliminary version of this paper. Moreover, we thank Neal Koblitz and Alfred Menezes for their inspiring work [13, 14]. The work of all three authors was supported by the EPSRC and the UK government as part of the grants EP/P009301/1, EP/P00881X/1 and EP/S01361X/1 for the first, second and third author respectively.

## References

1. Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, A Jalali, D Jao, B Koziel, B LaMacchia, P Longa, et al. Supersingular isogeny key encapsulation. *Submission to the NIST Post-Quantum Standardization project*, 2017.
2. Daniel J Bernstein. Cost analysis of hash collisions: Will quantum computers make sharcs obsolete. *SHARCS*, 9:105, 2009.
3. Gilles Brassard, Peter Hoyer, and Alain Tapp. Quantum algorithm for the collision problem. *arXiv preprint quant-ph/9705002*, 1997.
4. Denis X Charles, Kristin E Lauter, and Eyal Z Goren. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113, 2009.
5. David Chaum and Hans Van Antwerpen. Undeniable signatures. In *Conference on the Theory and Application of Cryptology*, pages 212–216. Springer, 1989.

6. Anamaria Costache, Brooke Feigon, Kristin Lauter, Maike Massierer, and Anna Puskás. Ramanujan graphs in cryptography. *arXiv preprint arXiv:1806.05709*, 2018.
7. Jean Marc Couveignes. Hard homogeneous spaces. *IACR Cryptology ePrint Archive*, 2006:291, 2006.
8. Ivan Damgård and Torben Pedersen. New convertible undeniable signature schemes. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 372–386. Springer, 1996.
9. Luca De Feo. Mathematics of isogeny based cryptography. *arXiv preprint arXiv:1711.04062*, 2017.
10. Steven D Galbraith and Frederik Vercauteren. Computational problems in supersingular elliptic curve isogenies. *Quantum Information Processing*, 17(10):265, 2018.
11. David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *International Workshop on Post-Quantum Cryptography*, pages 19–34. Springer, 2011.
12. David Jao and Vladimir Soukharev. Isogeny-based quantum-resistant undeniable signatures. In *International Workshop on Post-Quantum Cryptography*, pages 160–179. Springer, 2014.
13. Neal Koblitz and Alfred Menezes. Another look at "provable security". *IACR Cryptology ePrint Archive*, 2004:152, 2004.
14. Neal Koblitz and Alfred Menezes. Critical perspectives on provable security: Fifteen years of "another look" papers. *Adv. in Math. of Comm.*, 13(4):517–558, 2019.
15. David Kohel. *Endomorphism rings of elliptic curves over finite fields*. PhD thesis, University of California, Berkeley, 1996.
16. David Kohel, Kristin Lauter, Christophe Petit, and Jean-Pierre Tignol. On the quaternion  $\ell$ -isogeny path problem. *LMS Journal of Computation and Mathematics*, 17(A):418–432, 2014.
17. Kaoru Kurosawa and Jun Furukawa. Universally composable undeniable signature. In *International Colloquium on Automata, Languages, and Programming*, pages 524–535. Springer, 2008.
18. National Institute for Standards and Technology (NIST). Post-quantum crypto standardization (2016), <https://csrc.nist.gov/projects/post-quantum-cryptography>.
19. Christophe Petit. Faster algorithms for isogeny problems using torsion point images. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 330–353. Springer, 2017.
20. Alexander Rostovtsev and Anton Stolbunov. Public-key cryptosystem based on isogenies. *IACR Cryptology ePrint Archive*, 2006:145, 2006.
21. Joseph H Silverman. *The arithmetic of elliptic curves*, volume 106. Springer Science & Business Media, 2009.
22. M Seshadri Srinath and V Chandrasekaran. Isogeny-based quantum-resistant undeniable blind signature scheme. *International Journal of Network Security*, 20(1):9–18, 2018.

## A Undeniable (Blind) Signature Schemes

Undeniable signature schemes were introduced by Chaum and van Antwerpen [5], differing from traditional signature schemes in that verification of a signature

cannot be completed without cooperation from the signer. Following the notation of [17] we denote an undeniable signature scheme  $\Sigma$  by

$$\Sigma = \{\text{KeyGen}, \text{Sign}, \text{Check}, \text{Sim}, \pi_{con}, \pi_{dis}\}.$$

**KeyGen** is the PPT (probabalistic polynomial time) key generation algorithm, which outputs  $(vk, sk)$  - a verification and signing key, respectively. **Sign** is the PPT signing algorithm, taking a message  $m$  and  $sk$  as input to generate a signature  $\sigma$ . **Check** is a deterministic validity checking algorithm, such that  $\text{Check}((vk, m, \sigma), sk)$  returns 1 if  $(m, \sigma)$  is a valid message-pair and 0 if not. **Sim** is a PPT algorithm outputting a simulated signature  $\sigma'$  on input of  $vk$  and  $m$ . Finally,  $\pi_{con}$  and  $\pi_{dis}$  are confirmation and disavowal protocols, respectively, with which the signer can prove the validity (or invalidity) of a signature to the verifier. These are zero-knowledge interactive protocols.

An undeniable signature scheme must satisfy undeniability, unforgeability and invisibility. We use the definitions as stated in [8, 5, 17]. An undeniable blind signature scheme must also satisfy blindness, as defined in [22].

**Undeniability** requires that a signer cannot use the disavowal protocol to deny a valid signature. A signer is also unable to convince the verifier that an invalid signature is valid.

**Unforgeability** is the notion that an adversary cannot compute a valid message-signature pair with non-negligible probability. It is defined using the following security game:

1. The challenger generates a key pair, giving the verification key to the adversary.
2. The adversary is given access to a signing oracle and makes queries adaptively with messages  $m_i$ , for  $i = 1, 2, \dots, k$ , for some  $k$ , receiving corresponding signatures  $\sigma_i$ .
  - (a) The adversary additionally has access to a confirmation/disavowal oracle for the protocol, which they can query adaptively with message-signature pairs throughout step 2.
3. The adversary outputs a pair  $(m, \sigma)$ .

The adversary wins the game (i.e. successfully forges a signature) if  $(m, \sigma)$  is a valid message-signature pair and  $m \neq m_i$  for any  $i = 1, 2, \dots, k$ . A signature scheme is *unforgeable* if any PPT adversary wins with only negligible probability.

**Invisibility** requires that an adversary cannot distinguish between a valid signature and a simulated signature with non-negligible probability. It is defined by the following security game:

1. The challenger generates a a key pair, giving the verification key to the adversary.

2. The adversary is given access to a signing oracle and makes queries adaptively with messages  $m_i$ , for  $i = 1, 2, \dots, k$ , for some  $k$ , receiving corresponding signatures  $\sigma_i$ .
  - (a) The adversary additionally has access to a confirmation/disavowal oracle for the protocol, which they can query adaptively with message-signature pairs throughout step 2.
3. The adversary sends a new message  $m_j$  to the challenger.
4. The challenger computes a random bit  $b$ . If  $b = 1$ , the challenger computes  $\sigma = \mathbf{Sign}(m_j, sk)$ . If  $b = 0$  the challenger computes  $\sigma = \mathbf{Sim}(m_j, vk)$ . The challenger sends  $\sigma$  to the adversary.
5. The adversary is able to query the signing oracle again, with access to the confirmation/disavowal oracles. They cannot submit  $(m_j, \sigma)$  to either oracle.
6. The adversary outputs a bit  $b^*$ .

The adversary wins the game if  $b^* = b$ . An undeniable signature scheme is *invisible* if  $|\Pr(b = b^*) - 1/2|$  is negligible.

**Blindness** requires that an adversary cannot relate message-signature pairs with their associated blind versions with non-negligible probability. It is defined by the following security game:

1. The adversary generates a key pair  $(sk, vk)$ .
2. The adversary chooses two messages,  $m_0$  and  $m_1$ , and sends them to the challenger.
3. The challenger computes a random bit  $b$  and reorders the messages as  $(m_b, m_{b-1})$ .
4. The challenger blinds the messages and sends them to the adversary.
5. The adversary signs the blinded messages, generating the signatures  $\sigma_b^{blind}$  and  $\sigma_{b-1}^{blind}$ , which are returned to the challenger.
6. The challenger applies an unblinding algorithm to  $\sigma_b^{blind}$  and  $\sigma_{b-1}^{blind}$  and reveals the unblinded signatures,  $\sigma_b$  and  $\sigma_{b-1}$ , to the adversary.
7. The adversary outputs a bit  $b'$ .

The adversary wins if  $b' = b$ . A signatures scheme is *blind* if  $|\Pr(b = b') - 1/2|$  is negligible.