AN EFFICIENT KEY RECOVERY ATTACK ON SIDH (PRELIMINARY VERSION)

WOUTER CASTRYCK AND THOMAS DECRU

imec-COSIC, KU Leuven

Abstract. We present an efficient key recovery attack on the Supersingular Isogeny Diffie–Hellman protocol (SIDH), based on a "glue-and-split" theorem due to Kani. Our attack exploits the existence of a small non-scalar endomorphism on the starting curve, and it also relies on the auxiliary torsion point information that Alice and Bob share during the protocol. Our Magma implementation breaks the instantiation `SIKEp434`, which aims at security level 1 of the Post-Quantum Cryptography standardization process currently ran by NIST, in about one hour on a single core. This is a preliminary version of a longer article in preparation.

1. Set-up

We present a new and powerful key recovery attack on the Supersingular Isogeny Diffie–Hellman key exchange protocol (SIDH) [16] and its instantiation SIKE [15] that recently advanced to the fourth round of NIST's ongoing Post-Quantum Cryptography standardization process. It is based on a "glue-and-split" theorem from 1997 due to Ernst Kani [17, Thm. 2.6] and heavily outperforms previous attack strategies, such as the ones discussed in [21], [7, §5], [9].

We target Bob's private key, which is obtained by pushing 2^a-torsion points through a secret 3^b-isogeny. This case allows for the easiest and fastest implementation, but the method can also be used to recover Alice's key, and more generally works for arbitrary choices for ℓ_{Alice} and ℓ_{Bob} instead of just $\ell_{\text{Alice}} = 2$ and $\ell_{\text{Bob}} = 3$. The attack also generalizes to arbitrary (smooth and coprime) torsion choices for Alice and Bob, as used in for example B-SIDH [5]. Ran on a single core, the appended Magma code breaks the Microsoft SIKE challenges `$IKEp182` and `$IKEp217` in about 4 minutes and 6 minutes, respectively. A run on the `SIKEp434` parameters, previously believed to meet NIST's quantum security level 1, took about 62 minutes, again on a single core. We also ran the code on random instances of `SIKEp503` (level 2), `SIKEp610` (level 3) and `SIKEp751` (level 5), which took about 2h19m, 8h15m and 20h37m, respectively.

Concretely, we present an algorithm which, upon input of

(i) a prime p of the form $2^a 3^b f - 1$ for integers $a \geq 2$, $b, f \geq 1$ with $2^a \approx 3^b$,
(ii) an elliptic curve $E_0/\mathbb{F}_{p^2}$ with $\#E_0(\mathbb{F}_{p^2}) = (p+1)^2$,
(iii) generators P_0, Q_0 of $E_0[2^a]$,
(iv) a 3^β-isogeny $\tau : E_0 \to E_{\text{start}}$ for some $\beta \geq 0$, where

$$E_{\text{start}} : y^2 = x^3 + x \qquad \text{or} \qquad E_{\text{start}} : y^2 = x^3 + 6x^2 + x$$

is one of the two commonly chosen base curves in SIDH/SIKE, with respective j-invariants 1728 and 287496,

(v) the codomain $E/\mathbb{F}_{p^2}$ of a secret cyclic 3^b-isogeny $\varphi : E_0 \to E$,

(vi) the generators $P = \varphi(P_0)$ and $Q = \varphi(Q_0)$ of $E[2^a]$,

returns the isogeny φ; for simplicity we assume that φ is uniquely determined, which is true with overwhelming probability. A note on input (iv): when attacking SIKE, at the initial stage we will have $\beta = 0$ and $E_0 = E_{\text{start}}$, so the reader can keep this setting in mind for now. But our attack will involve a recursion during which the value of β will grow, whence this more general formulation. Moreover, we will also need to cope with larger values of β when discussing other base curves E_0 than these two standard choices (see Section 8.2).

Modulo the factorization of polynomially many natural numbers of size $O(2^a)$, which only depend on a and b and can therefore be handled during a precomputation phase, the attack runs in heuristic polynomial time (on a classical computer) and, as the reader can tell from the above timings, is very efficient in practice. The heuristics behind this complexity claim will be discussed in the full version of our article.

In light of the work by Kohel–Lauter–Petit–Tignol [18] and Love–Boneh [20], all known ways to generate a supersingular base curve $E_0/\mathbb{F}_{p^2}$ in a trustless manner reveal an isogeny of the form (iv). Therefore, with the current state of affairs, SIDH appears to be fully broken for any publicly generated base curve. At first sight, it seems possible to thwart our attack by using a trusted set-up, or by having the base curve generated by Alice, as suggested in [6, §8] (in the threat model of SIKE, there is no incentive for Alice to mess up with this procedure, and she will learn Bob's ephemeral key in any case). However, as explained in Section 8.3, even in the absence of a known path to E_{start}, the glue-and-split method has attack potential that may lower the security. This should be investigated further before jumping to conclusions.

Acknowledgements. We thank Craig Costello and Frederik Vercauteren for helpful questions and suggestions, and we have also benefited indirectly from discussions with Luciano Maino. We acknowledge support by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 101020788 – Adv-ERC-ISOCRYPT) and also by CyberSecurity Research Flanders with reference number VR20192203.

2. Decision via gluing and splitting

For the moment, let us concentrate on a decision variant: we assume to be given (i), (ii), (iii) and an elliptic curve $E/\mathbb{F}_{p^2}$ satisfying $\#E(\mathbb{F}_{p^2}) = (p+1)^2$, along with generators P, Q of $E[2^a]$. The goal is to decide whether or not

(1) there exists a 3^b-isogeny $\varphi : E_0 \to E$ such that $\varphi(P_0) = P$ and $\varphi(Q_0) = Q$.

2.1. Temporary assumptions.

We impose two technical conditions that will be discussed in more detail later on:

- We suppose that $2^a > 3^b$.
- Let $c = 2^a - 3^b$. We assume that we can compute the images $P_c = \gamma(P_0)$ and $Q_c = \gamma(Q_0)$ under an arbitrary cyclic c-isogeny $\gamma : E_0 \to C$ to some codomain curve C.

Let $x \in \mathbb{Z}$ be a multiplicative inverse of 3^b modulo 2^a. Note that $-x$ is then a multiplicative inverse of c modulo 2^a.

2.2. **Kani's theorem.** If (1) holds then we can consider the isogeny

$$\psi = [-1] \circ \varphi \circ \hat{\gamma} : C \to E,$$

where we note that $\psi(P_c) = -cP$ and $\psi(Q_c) = -cQ$. For all $R, S \in C[2^a]$ we have that

$$e_{2^a}(x\psi(R), x\psi(S)) = e_{2^a}(R,S)^{x^2 c 3^b} = e_{2^a}(R,S)^{-1}$$

or in other words the group homomorphism

$$[x] \circ \psi|_{C[2^a]} : C[2^a] \to E[2^a]$$

is a so-called "anti-isometry" with respect to the 2^a-Weil pairing. This implies that the group

$$\langle (P_c, x\psi(P_c)), (Q_c, x\psi(Q_c)) \rangle = \langle (P_c, -xcP), (Q_c, -xcQ) \rangle = \langle (P_c, P), (Q_c, Q) \rangle \quad (2)$$

is maximally isotropic with respect to the 2^a-Weil pairing on the product $C \times E$ (equipped with the product polarization). Indeed, the Weil pairing on $C \times E$ is just the product of the Weil pairings of the corresponding components.

So it concerns the kernel of a $(2^a, 2^a)$-isogeny, i.e., a length-a chain of $(2,2)$-isogenies. This is a walk in the $(2,2)$-isogeny graph of superspecial principally polarized abelian surfaces over $\overline{\mathbb{F}}_p$, all of whose vertices are defined over $\mathbb{F}_{p^2}$. These vertices come in two types: about $p^2/288$ products of supersingular elliptic curves and about $p^3/2880$ Jacobians of superspecial genus-2 curves, see e.g. [1]. Therefore it is to be expected that most isogenies in the chain are between Jacobians of genus-2 curves, and such isogenies can be computed efficiently using classical formulae due to Richelot [22]. But the first step is clearly an exception to this: with overwhelming probability, this is a "gluing" step, mapping the product $C \times E$ to a Jacobian (more precisely, by Theorem 1 below this can only fail if $C \cong E$). Formulae for this gluing step were derived in [14] and are recalled in Section 6.

What is the role of the isogeny γ in all this? Its aim is to force us into the exceptional situation where the *last* step of the chain is split, i.e., the codomain of our $(2^a, 2^a)$-isogeny is again a product of elliptic curves. In that case the anti-isometry $x\psi|_{C[2^a]}$ and the group (2) are called "reducible". This event is characterized by the theorem of Kani [17, Thm. 2.6]:

Definition 1. Let C, E be two elliptic curves and $N \geq 2$ an integer. Let $\psi : C \to E$ be a separable isogeny and let $H_1, H_2 \subset \ker \psi$ be subgroups such that $H_1 \cap H_2 = \{0\}$, $\#H_1 \cdot \#H_2 = \deg \psi$ and $\#H_1 + \#H_2 = N$. Then the triplet (ψ, H_1, H_2) is called an *isogeny diamond configuration of order* N between C and E.

Theorem 1. *Let (ψ, H_1, H_2) be an isogeny diamond configuration of order $N \geq 2$ between two elliptic curves C and E. Let $d = \gcd(\#H_1, \#H_2)$, let $n = N/d$ and let $k_i = \#H_i/d$ for $i = 1, 2$. Then ψ factors uniquely over $[d]$, i.e. $\psi = \psi' \circ [d]$ and there is a unique reducible anti-isometry $\iota : C[N] \to E[N]$ such that*

$$\iota(k_1 R_1 + k_2 R_2) = \psi'(R_2 - R_1) \text{ for all } R_i \in [n]^{-1} H_i \ (i = 1, 2). \quad (3)$$

Moreover, every reducible anti-isometry $C[N] \to E[N]$ is of this form.

In our case, the kernel of ψ is a cyclic group of order $c3^b$, so it admits two (unique) cyclic subgroups H_1, H_2 of respective orders c and 3^b. We clearly have that $H_1 \cap H_2 = \{0\}$ and

$$\#H_1 + \#H_2 = 2^a, \quad \#H_1 \cdot \#H_2 = \deg \psi,$$

so the triplet (ψ, H_1, H_2) is an isogeny diamond configuration of order 2^a. Then Kani's theorem implies that our anti-isometry $x\psi|_{C[2^a]}$ is reducible. Indeed, let us check condition (3) explicitly: we need to verify that

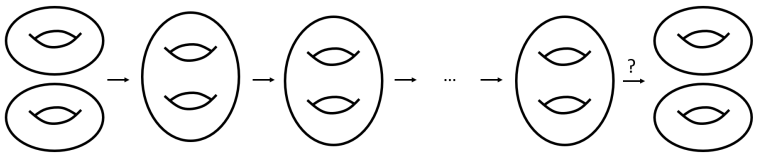
$$x\psi(cR_1 + 3^bR_2) = \psi(R_2 - R_1)$$

for all points R_1, R_2 such that $2^aR_1 \in H_1$ and $2^aR_2 \in H_2$ (note that $d = 1$ in our case). But this is easy: since $\psi(R_1)$ and $\psi(R_2)$ are 2^a-torsion points, we can rewrite the left hand side as

$$xc\psi(R_1)+x3^b\psi(R_2) = 3^{-b}(2^a-3^b)\psi(R_1)+3^{-b}3^b\psi(R_2) = \psi(R_2)-\psi(R_1) = \psi(S-R_1)$$

as wanted.

2.3. **Decision strategy.** Our decision strategy amounts to testing whether or not quotienting out $C \times E$ by (2) takes us to a product of elliptic curves. As we have just argued, if (1) holds, then we pass the test. For now, we content ourselves with the



loose heuristic that if (1) does not hold, then the test should fail with overwhelming probability because the proportion of products of elliptic curves among all vertices in the graph is only about $10/p$. We can actually be more precise about this heuristic in the cases that are relevant for our attack, namely the "wrong guesses" in our search-to-decision reduction from Section 4; this uses the converse implication in Kani's theorem and will be elaborated in the full version of our article.

3. Constructing and evaluating the auxiliary isogeny γ

3.1. **Construction.** The assumption that we can (efficiently) compute the image points P_c and Q_c under a degree-c isogeny is non-trivial, and this is where we need the factorization of an integer of size $O(2^a)$. It is also here that we rely on the special nature of E_{start}: both options come with an endomorphism $2\mathbf{i}$ satisfying $(2\mathbf{i})^2 = -4$. Indeed, on $E_{\text{start}} : y^2 = x^3 + x$ we have the automorphism $\mathbf{i} : (x, y) \mapsto (-x, \sqrt{-1}y)$ and we simply let $2\mathbf{i} = [2] \circ \mathbf{i}$. For $E_{\text{start}} : y^2 = x^3 + 6x^2 + x$ we can obtain $2\mathbf{i}$ as the composition of its outgoing 2-isogeny to $y^2 = x^3 + x$, the automorphism $\mathbf{i}$ on the latter curve, and the dual of the said 2-isogeny.

There is a reasonable chance that the prime factorization of c only involves prime factors that are congruent to 1 mod 4; this chance is roughly $1/\sqrt{a}$. As far as we are aware, the only known way to find out is by factoring c explicitly. Once this factorization is done and all prime factors are indeed congruent to 1 mod 4, we can efficiently write $c = u^2 + 4v^2 = (u + 2\mathbf{i}v)(u - 2\mathbf{i}v)$. Then

$$\gamma_{\text{start}} = [u] + [v] \circ 2\mathbf{i}$$

is an easy-to-evaluate degree-c endomorphism of E_{start}. Moreover, we can choose u, v such that this endomorphism is cyclic; this is automatic in the (likely) event that c is squarefree.

Remark 1. The method for finding u and v is classical: e.g., in the squarefree case, one computes

$$\prod_{\text{primes } \ell | c} \gcd(z_\ell + \mathbf{i}, \ell)$$

using Euclid's algorithm over the Gaussian integers; here z_ℓ is any integer such that $z_\ell^2 \equiv -1 \bmod \ell$. The outcome is among $\pm(u + 2\mathbf{i}v), \pm\mathbf{i}(u + 2\mathbf{i}v)$.

Then in order to find γ, we use the isogeny τ from input (iv). Let $\tilde{\tau} : E_{\text{start}} \to C$ be the isogeny with kernel $\gamma_{\text{start}}(\tau(E_0[3^\beta])) = \gamma_{\text{start}}(\ker \hat{\tau})$. Then $\tilde{\tau} \circ \gamma_{\text{start}} \circ \tau : E_0 \to C$ is a $3^{2\beta}c$-isogeny vanishing on $E_0[3^\beta]$, so it factors over $[3^\beta]$ and we can let

$$\gamma = \frac{\tilde{\tau} \circ \gamma_{\text{start}} \circ \tau}{3^\beta}.$$

It remains to see that γ is easy to evaluate on our 2^a-torsion points P_0 and Q_0. For this, we first discuss a special case.

3.2. **Evaluation: case $\beta \leq b$.** This is the only relevant case when attacking SIDH with base curve $E_0 = E_{\text{start}}$, as in the case of SIKE: while β will grow during our search-to-decision reduction, it will never grow beyond b. But then we always have that $\ker \hat{\tau} \subset E_0[3^b] \subset E(\mathbb{F}_{p^2})$. So we can explicitly write down a generator $T \in E_0(\mathbb{F}_{p^2})$ of $\ker \hat{\tau}$ and compute the isogeny $\tilde{\tau}$ with kernel $\langle \gamma_{\text{start}}(T) \rangle$. Evaluating γ in our 2^a-torsion points P_0 and Q_0 is then simply done by feeding them to $\tilde{\tau} \circ \gamma_{\text{start}} \circ \tau$ and scalar-multiplying the outcome with a multiplicative inverse of 3^β modulo 2^a. (In fact, this evaluation will naturally simplify in the context of our search-to-decision reduction below.)

3.3. **Evaluation: general case.** If $\beta > b$ then we cannot simply evaluate γ_{start} in a generator of $\ker \hat{\tau}$, unless we base change to a potentially very large and costly extension of $\mathbb{F}_{p^2}$. But note that the isogeny $\tilde{\tau}$ is precisely the pushforward isogeny $[\gamma_{\text{start}}]_* \hat{\tau}$ that was studied in [8, §4]. This suggests the following alternative method for computing $\tilde{\tau}$, which we will discuss in more detail in the full version of this article. Note that the specific choice of E_{start} comes with an explicit isomorphism

$$\iota : \text{End}(E_{\text{start}}) \to \mathcal{O}_{\text{start}}$$

where $\mathcal{O}_{\text{start}}$ is a maximal order in the quaternion algebra $B_{p,\infty} = \langle 1, \mathbf{i}, \mathbf{j}, \mathbf{ij} \rangle_{\mathbb{Q}}$ with $\mathbf{i^2} = -1$ and $\mathbf{j}^2 = -p$. Then:

(1) First, one converts the isogeny $\hat{\tau} : E_{\text{start}} \to E_0$ into a left ideal $I_{\hat{\tau}} \subset \mathcal{O}_{\text{start}}$ of norm 3^β, e.g. following [12, Alg. 3]. In fact, in the main use cases of this general method, a large component of the isogeny $\hat{\tau}$ will arise *from* its corresponding left $\mathcal{O}_{\text{start}}$-ideal; so in those cases this step can be simplified.
(2) Next, one computes the left ideal $I_{\tilde{\tau}} = [(\iota(\gamma_{\text{start}}))]_* I_{\hat{\tau}}$ using the formula from [8, Lem. 3]; this ideal again has norm 3^β.
(3) Finally, one converts the ideal $I_{\tilde{\tau}}$ into a length-β chain of 3-isogenies emanating from E_{start}, e.g. using [12, Alg. 2]. Then $\tilde{\tau}$ is the composition of these 3-isogenies.

Then, here too, evaluating γ in P_0 and Q_0 is done by applying $\tilde{\tau} \circ \gamma_{\text{start}} \circ \tau$ and scalar-multiplying with an inverse of 3^β modulo 2^a.

Remark 2. There are many other candidate-ways for constructing the isogeny γ. Just to give one similar example, decompositions of the form $c = u^2 + 3v^2$ are useful as soon as one knows an explicit path to $y^2 = x^3 + 1$, because this curve comes equipped with an endomorphism ω such that $\omega^2 = -3$. A different type of example is the case where c is very smooth: in that case one can construct the desired c-isogeny $\gamma : E_0 \to C$ as a composition of small degree isogenies *without* knowing a path to some special-featured curve; see Section 8.3 for further discussion. Unfortunately/fortunately, this event is unlikely.

4. Key recovery algorithm: basic version

We resume with the set-up from Section 1. The previous sections suggest the following iterative approach to full key recovery. We assume for simplicity that $\beta = 0$, so that the base curve E_0 coincides with E_{start}. Recall that this is the case in SIKE. In the general case, one should just replace the maps $\hat{\kappa}_1 : E_1 \to E_0$, $\widehat{\kappa_2\kappa_1} : E_2 \to E_0, \ldots$ below with their compositions with τ.

4.1. **Iteration.** For the first iteration, choose $\beta_1 \geq 1$ minimal such that there exists some $\alpha_1 \geq 0$ for which

$$c_1 = 2^{a-\alpha_1} - 3^{b-\beta_1}$$

is positive and only has prime factors congruent to 1 mod 4. Write $\varphi = \varphi_1 \circ \kappa_1$ with κ_1 a 3^{β_1}-isogeny. To an attacker, there are a priori 3^{β_1} options for κ_1 (this assumes knowledge of an "incoming isogeny", otherwise there are $4 \cdot 3^{\beta_1 - 1}$ options). For each of these options, we can run our decision algorithm on

(ii) the curve $E_1 = \kappa_1(E_0)$,
(iii) the generators $P_1 = \kappa_1(2^{\alpha_1}P_0)$ and $Q_1 = \kappa_1(2^{\alpha_1}Q_0)$ of $E_1[2^{a-\alpha_1}]$,
(iv) the 3^{β_1}-isogeny $\hat{\kappa}_1 : E_1 \to E_0$,
(v) the codomain E; if the guess is correct then it is connected to E_1 via the unknown isogeny φ_1 of degree $3^{b-\beta_1}$,
(vi) the generators $2^{\alpha_1}P, 2^{\alpha_1}Q$ of $E[2^{a-\alpha_1}]$

where the numbering (ii)-(vi) is chosen to be consistent with that of Section 1. According to our heuristic assumption discussed at the end of Section 2, we expect that only the correct guess for κ_1 will pass the test.

Let us discuss in more detail what "running the test" amounts to in this case. First, one must compute the images P_{c_1}, Q_{c_1} of P_1, Q_1 under the isogeny

$$\gamma_1 = \frac{\tilde{\hat{\kappa}}_1 \circ \gamma_{\text{start}} \circ \hat{\kappa}_1}{3^{\beta_1}}$$

where $\tilde{\hat{\kappa}}_1 : E_{\text{start}} \to C_1$ is the isogeny with kernel $\gamma_{\text{start}}(\ker \kappa_1)$. Observe that this simplifies: all one should do is compute

$$P_{c_1} = 2^{\alpha_1}\tilde{\hat{\kappa}}_1\gamma_{\text{start}}(P_0), \quad Q_{c_1} = 2^{\alpha_1}\tilde{\hat{\kappa}}_1\gamma_{\text{start}}(Q_0). \tag{4}$$

Once these points have been computed, one checks whether the quotient of $C_1 \times E$ by the $(2^{a-\alpha_1}, 2^{a-\alpha_1})$-subgroup

$$\langle (P_{c_1}, 2^{\alpha_1}P), (Q_{c_1}, 2^{\alpha_1}Q) \rangle \tag{5}$$

is again a product of elliptic curves. This is done by computing the corresponding chain of $(2,2)$-isogenies. With overwhelming probability, the first $a - \alpha_1 - 1$ steps in this chain amount to one gluing step followed by $a - \alpha_1 - 2$ Richelot isogenies

between Jacobians of genus-2 curves. An easy "$\delta = 0$ test" then checks whether or not the last step splits. See Section 6 for more algorithmic details.

If the test fails, then we try again with a different guess for κ_1. We remark that, even in the case of a wrong guess, the subgroup (5) is always maximally isotropic with respect to the Weil pairing, so this is *not* the way in which one can detect having taken the wrong direction: one really has to perform the gluing and its successive Richelot walk. (The failure of detecting wrong steps using the Weil pairing is well-known, see e.g. [13, §7.2]; with some imagination, our attack can be viewed as a refinement of this approach.)

If the test passes, then we have found the correct instance of κ_1 and we continue from E_1. That is, we let $\beta_2 > \beta_1$ be minimal such that there is some $\alpha_2 \geq 0$ for which $c_2 = 2^{a-\alpha_2} - 3^{b-\beta_2}$ is positive and all its prime factors are congruent to 1 mod 4. Now one tries to recover the $3^{\beta_2-\beta_1}$-component $\kappa_2 : E_1 \to E_2$ such that $\varphi_1 = \varphi_2 \circ \kappa_2$. In this case, for each guess for κ_2 one computes

$$P_{c_2} = 2^{\alpha_2}\widetilde{\kappa_2\kappa_1}\gamma_{\text{start}}(P_0), \quad Q_{c_2} = 2^{\alpha_2}\widetilde{\kappa_2\kappa_1}\gamma_{\text{start}}(Q_0)$$

with $\widetilde{\kappa_2\kappa_1} : E_{\text{start}} \to C_2$ the isogeny with kernel $\gamma_{\text{start}}(\ker \kappa_2\kappa_1)$. One then checks whether

$$\langle (P_{c_2}, 2^{\alpha_2}P), (Q_{c_2}, 2^{\alpha_2}Q) \rangle \ \subset \ C_2 \times E$$

is reducible or not. By continuing in this way, one eventually retrieves all of φ.

4.2. **Step sizes.** The gaps between the consecutive integers $0, \beta_1, \beta_2, \beta_3, \ldots, \beta_r = b$ should be as small as possible, because this reduces the number of possible guesses in each iteration. More concretely, the expected number of $(2,2)$-chains that need to be computed is about

$$\frac{1}{2}\left(3^{\beta_1} + 3^{\beta_2-\beta_1} + 3^{\beta_3-\beta_2} + \ldots + 3^{b-\beta_{r-1}}\right). \tag{6}$$

A necessary condition on each β_i is that $b - \beta_i$ is odd, except in the last iteration where we have $\beta_r = b$. Indeed, if $b - \beta_i > 0$ is even then

$$c_i = 2^{a-\alpha_i} - 3^{b-\beta_i} \equiv 3 \bmod 4$$

must admit at least one prime factor that is congruent to 3 mod 4. Therefore the best we can hope is that the sequence grows by steps of two, in which case the estimate (6) becomes about $9b/4$. Experiment shows that this optimal estimate lies close to reality, with the only exceptions corresponding to small β_i. This makes sense: as β_i grows, the amount of leeway (i.e., the number of candidate α_i's) grows as well, and moreover the probability of success increases as c_i is allowed to get smaller. Example: for the parameters of `SIKEp434` where we have $a = 216$ and $b = 137$, one quickly finds suitable α_i for every even β_i in $\{0, 1, \ldots, b\} \setminus \{4\}$.

4.3. **Rephrasing in terms of Bob's secret key.** In practice, SIDH comes with public generators $P_{\text{Bob}}, Q_{\text{Bob}}$ of $E_0[3^b]$ and Bob's secret isogeny φ is encoded as the integer

$$\text{sk}_{\text{Bob}} \in [0, 3^b)$$

for which $\ker \varphi = \langle P_{\text{Bob}} + \text{sk}_{\text{Bob}}\, Q_{\text{Bob}} \rangle$. Upon expanding

$$\text{sk}_{\text{Bob}} = k_1 + k_2 3^{\beta_1} + \ldots + k_r 3^{\beta_{r-1}}, \qquad k_i \in [0, 3^{\beta_i - \beta_{i-1}} - 1)$$

(where we let $\beta_0 = 0$), we observe that

$$\ker \kappa_1 = \langle 3^{b-\beta_1} P_{\text{Bob}} + k_1 3^{b-\beta_1} Q_{\text{Bob}} \rangle. \tag{7}$$

So the first iteration amounts to

- guessing k_1,
- determining the 3^{β_1}-isogeny $\tilde{\tilde{\kappa}}_1 : E_{\text{start}} \to C_1$ with kernel $\gamma_{\text{start}}(\ker \kappa_1)$, with $\ker \kappa_1$ as in (7),
- computing the points $P_{c_1}, Q_{c_1} \in C_1$ as in (4),
- checking whether or not the subgroup (5) is reducible.

After finding k_1, we proceed with

$$\ker \kappa_2 = \langle 3^{b-\beta_2} P_{\text{Bob}} + (k_1 + k_2 3^{\beta_1}) 3^{b-\beta_2} Q_{\text{Bob}} \rangle$$

in order to determine k_2, and so on. So the attack determines sk_{Bob} digit by digit. If all the gaps are of size two, then this amounts to determining one base-9 digit of sk_{Bob} at a time.

5. Some speed-ups

5.1. **Take α_i as large as possible.** If for a given β_i there indeed exists some $\alpha_i \geq 0$ such that $c_i = 2^{a-\alpha_i} - 3^{b-\beta_i}$ is positive and free of prime factors congruent to 3 mod 4, then usually α_i is not the unique integer with that property, so there is some freedom. The larger we choose α_i, the smaller will be the length $a - \alpha_i$ of our chain of $(2,2)$-isogenies. Therefore, it is more efficient to take larger α_i's.

5.2. **Use a precomputed table.** We have precomputed a table which for all $s \in \{1, 3, 5, \ldots, 239\}$ stores the smallest integer $t(s)$ such that $2^{t(s)} - 3^s$ is a product of primes congruent to 1 modulo 4. It also stores corresponding values for u and v. The table is available as `uvtable.m` and can be used as follows: for every candidate-β_i such that $b - \beta_i$ is odd, one checks whether or not $t(b - \beta_i) \leq a$. If not, then we proceed to the next candidate. If yes, then we can use this instance of β_i, and we choose $a - t(b - \beta_i)$ as a corresponding value for α_i. This makes sure that α_i is as large as possible, and moreover we have u, v readily available, without the need for factoring. Our table is sufficiently large to be used for each of the proposed parameter sets for SIKE, up to `SIKEp751` targeting NIST's security level 5.

5.3. **Extend Bob's secret isogeny where useful.** Imagine that some candidate-β_i does not admit an integer $\alpha_i \geq 0$ such that $2^{a-\alpha_i} - 3^{b-\beta_i}$ is a product of primes congruent to 1 mod 4 (e.g., because $b - \beta_i > 0$ is even). But imagine that $\beta_i - 1$ does. Then one can prolong Bob's secret isogeny with an arbitrary 3-isogeny φ' and let $P' = \varphi'(P)$ and $Q' = \varphi'(Q)$. Treating $\varphi' \circ \varphi$ as the new secret isogeny, the relevant expression now becomes $2^{a-\alpha_i} - 3^{b+1-\beta_i}$, and we know that there exists some $\alpha_i \geq 0$ for which this *is* a product of primes congruent to 1 mod 4. We can now use our attack to determine Bob's secret key modulo 3^{β_i} and proceed.

In practice, this means that most step sizes drop from 2 to 1, or in other words that we are determining one base-3 digit of sk_{Bob} at a time. The only possibly larger step occurs at the beginning of the iteration. For instance, in the case of `SIKEp751`, the smallest β_1 such that $2^a - 3^{b-\beta_1} > 0$ is $\beta_1 = 6$, so we cannot hope for a smaller first gap. This implies a rather costly start of the algorithm: of the 20.6 hours that we spent on breaking `SIKEp751`, about 14 hours were needed for determining the first 6 out of 239 ternary digits of sk_{Bob}.

Remark 3. If 2^a is considerably smaller than 3^b, then it probably makes more sense to attack Alice's private key instead of Bob's, using chains of $(3,3)$-isogenies; see

Section 8.1. Of course, if 2^a gets much smaller than 3^b, then one enters the regime of the torsion-point attack from [9].

Remark 4. There is a 1/4 probability that the random isogeny φ' matches with the dual of the last degree-3 component of φ. In this case, the wrong guesses are also at distance $3^{b-\beta_i}$ from E, so this creates false positives, leaving us clueless about which is the correct guess. However, this is easy to fix: if multiple guesses pass the test, then all one needs to do is change φ', and then we have identified the dual direction once and for all. If this happens, then it will be discovered when trying to determine the ternary digit at position $\beta_2 = \beta_1 + 1$ (and this does not affect the correctness of the first β_1 digits, as these were determined without the use of φ').

6. Computing chains of (2,2)-isogenies

In this section we explain how to determine whether or not a $(2^a, 2^a)$-subgroup $\langle (P_c, P), (Q_c, Q)\rangle$ of a product of elliptic curves $C \times E$ is reducible. Throughout, we avoid dealing with certain exceptional cases, e.g. every genus-2 curve $H : y^2 = h(x) = c_6x^6 + c_5x^5 + \ldots + c_0$ encountered is assumed to satisfy $c_6 \neq 0$, so that it has two places ∞_1, ∞_2 at infinity, and all points on its Jacobian J_H that we deal with are assumed to be representable as $(\alpha_1, \beta_1) + (\alpha_2, \beta_2) - \infty_1 - \infty_2$ with $\alpha_1 \neq \alpha_2$, so that they have a Mumford representation of the form $[x^2 + u_1x + u_0, v_1x + v_0]$. Moreover, all our chains of $(2, 2)$-isogenies are assumed to start off by gluing $C \times E$ into a Jacobian, after which we never run into a product of elliptic curves again, except possibly at the a-th and last step. The exceptions to these assumptions are expected to occur with probability $O(p^{-1})$, so we see no need to discuss nor implement them.

6.1. **Gluing elliptic curves into a Jacobian.** In the first step we want to glue the curves C and E into the Jacobian of a genus-2 curve H via the $(2, 2)$-subgroup $\langle (2^{a-1}P_c, 2^{a-1}P), (2^{a-1}Q_c, 2^{a-1}Q)\rangle$. We also need to push the points (P_c, P), (Q_c, Q) through the corresponding isogeny. The relevant equations are as follows. We refer to [14, Prop. 4] and its proof for further details.

Proposition 1. *Let $C/K : y^2 = (x - \alpha_1)(x - \alpha_2)(x - \alpha_3)$ and $E : y^2 = (x - \beta_1)(x - \beta_2)(x - \beta_3)$ be elliptic curves over a field K of characteristic different from two. Write Δ_α for the discriminant of $(x - \alpha_1)(x - \alpha_2)(x - \alpha_3)$ and Δ_β for the discriminant of $(x - \beta_1)(x - \beta_2)(x - \beta_3)$. Furthermore, define*

$$\begin{aligned}
a_1 &= (\alpha_3 - \alpha_2)^2/(\beta_3 - \beta_2) + (\alpha_2 - \alpha_1)^2/(\beta_2 - \beta_1) + (\alpha_1 - \alpha_3)^2/(\beta_1 - \beta_3),\\
b_1 &= (\beta_3 - \beta_2)^2/(\alpha_3 - \alpha_2) + (\beta_2 - \beta_1)^2/(\alpha_2 - \alpha_1) + (\beta_1 - \beta_3)^2/(\alpha_1 - \alpha_3),\\
a_2 &= \alpha_1(\beta_3 - \beta_2) + \alpha_2(\beta_1 - \beta_3) + \alpha_3(\beta_2 - \beta_1),\\
b_2 &= \beta_1(\alpha_3 - \alpha_2) + \beta_2(\alpha_1 - \alpha_3) + \beta_3(\alpha_2 - \alpha_1),\\
A &= \Delta_\beta a_1/a_2, \quad B = \Delta_\alpha b_1/b_2,\\
h(x) &= -\big(A(\alpha_2 - \alpha_1)(\alpha_1 - \alpha_3)x^2 + B(\beta_2 - \beta_1)(\beta_1 - \beta_3)\big)\\
&\quad \cdot \big(A(\alpha_3 - \alpha_2)(\alpha_2 - \alpha_1)x^2 + B(\beta_3 - \beta_2)(\beta_2 - \beta_1)\big)\\
&\quad \cdot \big(A(\alpha_1 - \alpha_3)(\alpha_3 - \alpha_2)x^2 + B(\beta_1 - \beta_3)(\beta_3 - \beta_2)\big)\,.
\end{aligned}$$

Then the $(2, 2)$-isogeny with domain $C \times E$ and kernel

$$\big\langle ((\alpha_1, 0), (\beta_1, 0)),\ ((\alpha_2, 0), (\beta_2, 0))\big\rangle$$

has as codomain the Jacobian of a genus-2 curve H defined by $y^2 = h(x)$. The degree-2 morphisms of the dual isogeny are given by

$$\begin{aligned}
&\varphi_1 : H \to C \\
&\quad (x, y) \mapsto (s_1/x^2 + s_2, (\Delta_\beta/A^3)(y/x^3)), \\
&\varphi_2 : H \to E \\
&\quad (x, y) \mapsto (t_1 x^2 + t_2, (\Delta_\alpha/B^3)y),
\end{aligned}$$

where

$$\begin{aligned}
s_1 &= -(B/A)(a_2/a_1), \\
s_2 &= \frac{1}{a_1}\left(\frac{\alpha_1(\alpha_3-\alpha_2)^2}{\beta_3-\beta_2} + \frac{\alpha_2(\alpha_1-\alpha_3)^2}{\beta_1-\beta_3} + \frac{\alpha_3(\alpha_2-\alpha_1)^2}{\beta_2-\beta_1}\right), \\
t_1 &= -(A/B)(b_2/b_1), \\
t_2 &= \frac{1}{b_1}\left(\frac{\beta_1(\beta_3-\beta_2)^2}{\alpha_3-\alpha_2} + \frac{\beta_2(\beta_1-\beta_3)^2}{\alpha_1-\alpha_3} + \frac{\beta_3(\beta_2-\beta_1)^2}{\alpha_2-\alpha_1}\right).
\end{aligned}$$

The morphisms φ_i extend to the Jacobian J_H by mapping

$$[\sum_j P_j] \to \sum_j \varphi(P_j)$$

and they combine into a $(2,2)$-isogeny $\Phi : J_H \to C \times E$, the dual of which is our isogeny of interest. To compute the image of a point $(P_c, P) \in C \times E$ under this dual isogeny, it suffices to compute some $[D] \in \Phi^{-1}\{(P_c, P)\} \subset J_H$ and then double it. Indeed, then we have

$$2[D] = \hat{\Phi}\Phi([D]) = \hat{\Phi}(P_c, P)$$

as wanted.

Let $D = P_H + Q_H - \infty_1 - \infty_2$ represent a point on J_H. As mentioned, we assume that its Mumford representation is of the form $[x^2 + u_1 x + u_0, v_1 x + v_0]$. To avoid the need for field extensions, let us express $\varphi_i(P_H + Q_H)$ for $i = 1, 2$ directly in terms of u_0, u_1, v_0, v_1. Note that the divisor $\infty_1 + \infty_2$ maps to ∞, both under φ_1 and under φ_2, so it suffices to concentrate on $P_H + Q_H$.

The calculation is easiest for φ_2, where the line connecting $\varphi_2(P_H)$ and $\varphi_2(Q_H)$ has slope

$$\lambda_2 = -\frac{(\Delta_\alpha/B^3)v_1}{t_1 u_1}$$

and then $\varphi_2(P_H + Q_H)$ is

$$\left(\lambda_2^2 + \sum_{i=1}^{3}\beta_i - t_1(u_1^2 - 2u_0) - 2t_2\,,\; -\lambda_2\left(\cdots - t_2 + (u_0 v_1 - u_1 v_0)\frac{t_1}{v_1}\right)\right) \tag{8}$$

with $\cdots$ denoting a copy of the first coordinate. To derive formulae for φ_1, note that this map is of a very similar kind, except for the transformation

$$\tilde{\cdot} : (x, y) \mapsto (1/x, y/x^3)$$

by which it is preceded. Let $\tilde{u_0}, \tilde{u}_1, \tilde{v_0}, \tilde{v}_1$ be the Mumford coordinates of $\tilde{P}_H + \tilde{Q}_H$, then an easy calculation shows:

$$\tilde{u}_0 = \frac{1}{u_0}, \quad \tilde{u}_1 = \frac{u_1}{u_0}, \quad \tilde{v}_0 = \frac{u_1 v_0 - u_0 v_1}{u_0^2}, \quad \tilde{v}_1 = \frac{u_1^2 v_0 - u_0 v_0 - u_0 u_1 v_1}{u_0^2}.$$

Thus the formulae for the coordinates of $\varphi_1(P_H+Q_H)$ are the same as in (8), except for swapping the α_i's and the β_i's and for substituting $\tilde{u_0}, \tilde{u}_1, \tilde{v_0}, \tilde{v}_1$ for u_0, u_1, v_0, v_1.

This gives us 4 equations in the unknowns u_0, u_1, v_0, v_1:

$$\begin{cases} x(\varphi_1(P_H+Q_H)) = x(P_c), \\ y(\varphi_1(P_H+Q_H)) = y(P_c), \\ x(\varphi_2(P_H+Q_H)) = x(P), \\ y(\varphi_2(P_H+Q_H)) = y(P). \end{cases} \tag{9}$$

Together with the equation

$$\begin{aligned} 2v_0^2 - 2v_0v_1u_1 + v_1^2(u_1^2 - 2u_0) = {} & 2c_0 + (-u_1)c_1 + (u_1^2 - 2u_0)c_2 \\ & + (-u_1^3 + 3u_0u_1)c_3 + (u_1^4 - 4u_1^2u_0 + 2u_0^2)c_4 \\ & + (-u_1^5 + 5u_1^3u_0 - 5u_1u_0^2)c_5 \\ & + (u_1^6 - 6u_1^4u_0 + 9u_1^2u_0^2 - 2u_0^3)c_6, \end{aligned}$$

expressing that $[D] \in J_H$, this system is expected to have 4 solutions, all of which are defined over $\mathbb{F}_{p^2}$. (In practice, we found these solutions by clearing denominators in (9), running a Gröbner basis computation, and discarding solutions having zeroes among their coordinates, because they are most likely fake solutions that were created when clearing denominators.) Taking any of these solutions and doubling the corresponding point on J_H produces the desired image of (P_c, P).

6.2. **Richelot isogenies.** By assumption, the next $a-2$ steps are $(2,2)$-isogenies between Jacobians of genus-2 curves. Such maps are called Richelot isogenies and they are classical; for a contemporary exposition, including explicit formulae, we refer to Smith's thesis [22, Ch. 8]. Starting from a hyperelliptic curve $H : y^2 = h(x)$ and a $(2,2)$-subgroup

$$\big\langle [g_1(x), 0], [g_2(x), 0] \big\rangle, \quad g_1(x) = x^2 + g_{11}x + g_{10}, \quad g_2(x) = x^2 + g_{21}x + g_{20}$$

of its Jacobian, one lets $g_3(x) = h(x)/(g_1(x)g_2(x)) = g_{32}x^2 + g_{31}x + g_{30}$. One then computes

$$\delta = \det \begin{pmatrix} g_{10} & g_{11} & 1 \\ g_{20} & g_{21} & 1 \\ g_{30} & g_{31} & g_{32} \end{pmatrix}$$

and $h'(x) = g_1'(x)g_2'(x)g_3'(x)$ where

$$g_i'(x) = \delta^{-1} \left(\frac{dg_j}{dx} g_k - g_j \frac{dg_k}{dx} \right) \text{ for } (i,j,k) = (1,2,3),\ (2,3,1),\ (3,1,2).$$

Then the codomain of our Richelot isogeny is the Jacobian of $H' : \mathbf{y}^2 = h'(\mathbf{x})$. We use different notation for the coordinates because pushing a point through this isogeny is done via the "Richelot correspondence", which is the curve $X \subset H \times H'$ defined by

$$X : g_1(x)g_1'(\mathbf{x}) + g_2(x)g_2'(\mathbf{x}) = y\mathbf{y} - g_1(x)g_1'(\mathbf{x})(x - \mathbf{x}) = 0.$$

It naturally comes equipped with two projection maps $\pi : X \to H$, $\pi' : X \to H'$. The isogeny is then

$$J_H \to J_{H'} : [D] \mapsto [\pi'_* \pi^* D] \quad \text{(pullback along } \pi \text{ and pushforward along } \pi').$$

This means that in order to compute the image of a point $[x^2+u_1x+u_0, v_1x+v_0] \in J_H$, one should eliminate the variables x, y from the system

$$\begin{cases} x^2 + u_1x + u_0 = 0, \\ y = v_1x + v_0, \\ y^2 = h(x), \\ g_1(x)g_1'(\mathbf{x}) + g_2(x)g_2'(\mathbf{x}) = 0, \\ y\mathbf{y} = g_1(x)g_1'(\mathbf{x})(x - \mathbf{x}). \end{cases}$$

We expect the last two equations of its reduced Gröbner basis (with respect to the lexicographic order with $\mathbf{x} \prec \mathbf{y} \prec y \prec x$) to be of the form

$$\mathbf{y} = v_3'\mathbf{x}^3 + v_2'\mathbf{x}^2 + v_1'\mathbf{x} + v_0', \quad \mathbf{x}^4 + u_3'\mathbf{x}^3 + u_2'\mathbf{x}^2 + u_1'\mathbf{x} + u_0' = 0$$

and then $[\mathbf{x}^4 + u_3'\mathbf{x}^3 + u_2'\mathbf{x}^2 + u_1'\mathbf{x} + u_0', v_3'\mathbf{x}^3 + v_2'\mathbf{x}^2 + v_1'\mathbf{x} + v_0']$ are non-reduced Mumford coordinates for the image on $J_{H'}$.

6.3. **Split or not?** We now want to check whether or not the a-th $(2,2)$-isogeny takes us back to a product of elliptic curves. This is easy: we proceed as if we are dealing with a Richelot isogeny (just the codomain computation, no points need be pushed through anymore). It can be shown that the determinant δ vanishes if and only if the codomain is a product of elliptic curves instead of the Jacobian of a genus-2 curve. Therefore the final and deciding step in our computation simply amounts to verifying whether or not $\delta = 0$.

7. Magma code

This paper comes with the following Magma files. They are available at `https://homes.esat.kuleuven.be/~wcastryc`:

- `richelot_aux.m` contains auxiliary functions, mainly for computing chains of $(2,2)$-isogenies, where the functions `FromProdtoJac` and `FromJactoJac` are implementations of the methods described in Section 6,
- `uvtable.m` contains precomputed values of u and v as described in Section 5.2,
- a run of `SIKE_challenge.m` loads the first two files and breaks `$IKEp217` by running the algorithm from Section 4, incorporating the speed-ups from Section 5,
- a run of `SIKEp434.m` generates random input for the `SIKEp434` parameters and runs the algorithm from Section 4, again incorporating the speed-ups from Section 5; to attack `SIKEp503`, `SIKEp610` and `SIKEp751` one simply replaces the line `a := 216; b := 137;` by `a := 250; b := 159;`, `a := 305; b := 192;`, `a := 372; b := 239;`, respectively.

The reader can run these files in order to confirm the approximate timings mentioned in Section 1. We ran them on an Intel Xeon CPU E5-2630v2 at 2.60GHz.

8. Generalizations

8.1. **Arbitrary torsion.** There is no theoretical obstruction to attacking Alice's public key instead of Bob's. In this case one will end up computing a chain of $(3,3)$-isogenies, which is slightly more convoluted, but still doable using the machinery from [2]; see also [11]. The formulae are still practical and recovering Alice's private key can then be done bit ber bit (except possibly for some offset of the kind discussed in Section 5.3). Altogether, we expect having to compute approximately a chains

of $(3,3)$-isogenies of length at most b in order to retrieve Alice's private key. The expression Δ in the formulae from [2] plays a similar role as δ in the Richelot isogeny formulae, in the sense that $\Delta = 0$ occurs if and only if the codomain of the $(3,3)$-isogeny is the product of two elliptic curves, see [3]. Therefore, verifying whether the final $(3,3)$-isogeny splits is just as easy.

More generally, one can attack SIDH when set up using arbitrary small primes $\ell_{\text{Alice}}, \ell_{\text{Bob}}$ instead of just $2, 3$, or even more general smooth torsion as in B-SIDH. Inherently, this changes nothing to our attack, except that now one must compute (ℓ,ℓ)-isogenies for primes $\ell \geq 5$. For isogenies between Jacobians of genus-2 curves, we refer to the work of Cosset and Robert [4], whose formulae are a lot more involved than those to compute $(2,2)$- and $(3,3)$-isogenies, but they are polynomial in ℓ and likely practical enough to complete the attack. The gluing of elliptic curves and splitting of Jacobians is succinctly explained by Kuhn in [19]; for a more elaborate and practical exposition, see also [10, §1.4]. Away from $\ell = 2, 3$ we are not aware of a straightforward decision algorithm to verify whether an (ℓ,ℓ)-subgroup of a given Jacobian of a genus-2 curve results in a product of elliptic curves: the easiest way seems to try and compute an (ℓ,ℓ)-isogeny to a Jacobian as in [4] and see if the theta constants fail to create a genus-2 curve. Alternatively, one can write down a system of equations expressing that our Jacobian is "(ℓ,ℓ)-split" (i.e., (ℓ,ℓ)-isogenous to a product of elliptic curves) via our given subgroup, and verify whether this system is consistent, see [10].

8.2. **Other base curves with a known path to E_{start}.** All current instantiations of SIDH/SIKE have as base curve $E_0 = E_{\text{start}}$, where

$$E_{\text{start}} : y^2 = x^3 + x \qquad \text{or} \qquad E_{\text{start}} : y^2 = x^3 + 6x^2 + x$$

is one of the two options listed in Section 1. However, with the currently known methods for generating supersingular elliptic curves, every publicly generated alterative to E_0 comes with a known path to E_{start}, in view of the work of Love and Boneh [20]. The KLPT algorithm from [18] can convert this path into an isogeny $\tau : E_0 \to E_{\text{start}}$ of degree 3^β for some $\beta \geq 0$. Thus we are in business for running our glue-and-split attack, where the auxiliary isogenies γ should now be evaluated as explained in Section 3.3. In conclusion, using another publicly generated base curve does not thwart the attack.

8.3. **Base curves without a known path to E_{start}.** We now discuss the scenario where no path to E_{start} is known. As indicated in Remark 2, if $c = 2^a - 3^b$ is smooth then it remains possible to construct the auxiliary isogeny γ. In fact, if we no longer exploit special features of E_0, then it makes more sense to let γ emanate from E rather than E_0, leading us to considering $\gamma \circ \varphi : E_0 \to C$. This isogeny has degree $c3^b$ and can again be used to decide whether or not assumption (1) holds: this should be the case if and only if the subgroup $\langle (P_0, x\gamma(P)), (Q_0, x\gamma(Q)) \rangle \subset E_0 \times C$ is reducible, with x a multiplicative inverse of 3^b modulo 2^a.

Remark 5. Computing γ works as follows. Write c as a product of small primes $\ell_1 \ell_2 \cdots \ell_s$ and for each $i = 1, \ldots, s$ let r_i denote the multiplicative order of $-p$ modulo ℓ_i. Because p^2-Frobenius acts as $[-p]$, we can find a non-trivial point in $E_0[\ell_1] \subset E_0(\mathbb{F}_{p^{2r_1}})$ and the subgroup it generates is defined over $\mathbb{F}_{p^2}$. So this is the kernel of an $\mathbb{F}_{p^2}$-rational degree-ℓ_1 isogeny $\gamma_1 : E_0 \to C_1$ that can be computed and evaluated using formulae of Vélu type. By repeating this construction, we

eventually obtain γ as a composition $\gamma_s \circ \gamma_{s-1} \circ \ldots \circ \gamma_1$ where each γ_i is an $\mathbb{F}_{p^2}$-rational ℓ_i-isogeny.

Turning this decision method into a key recovery algorithm works along the lines of Section 4. First, we look for the smallest $\beta \geq 1$ for which there exists an integer $\alpha \geq 0$ such that

$$c = 2^{a-\alpha} - 3^{b-\beta} \tag{10}$$

is smooth (this is an optimistic goal!). Then, for each guess for the first degree-3^β-component κ_1 of φ, we run our test to see whether or not there exists a degree-$3^{b-\beta}$-isogeny $\kappa_1(E_0) \to E$ mapping $2^\alpha\kappa_1(P_0)$ to $2^\alpha P$ and $2^\alpha\kappa_1(Q_0)$ to $2^\alpha Q$. There are 3^β possible guesses, so clearly β should be small enough for this to be feasible.

Once κ_1 is found, we can proceed by steps of degree 3 as in Section 5.3. Since smoothness is such a rare event, it actually makes sense to recycle the expression (10) all along. Then we can also recycle our auxiliary isogeny γ, i.e., it only has to be computed once, including pushing through torsion points. Concretely: when guessing κ_2, we extend γ with an extra degree-3 isogeny $\varphi' : C \to E'$ and we test if we took the right direction by checking whether or not there is a degree $c3^{b-\beta}$-isogeny mapping $2^\alpha\kappa_2\kappa_1(P_0)$ to $2^\alpha\varphi'\gamma(P)$ and $2^\alpha\kappa_2\kappa_1(Q_0)$ to $2^\alpha\varphi'\gamma(Q)$. Iterating this process will recreate the entire isogeny chain.

In summary: as soon as we can find a small $\beta \geq 1$ with a corresponding $\alpha \geq 0$ such that (10) is smooth, then our attack applies. The likeliness of finding a smooth c of this form is very small, so this is not expected to lead to a practical attack, but it might lower the security level of certain parameter sets. Moreover, there are at least two ways to create more leeway:

- We can extend Bob's secret isogeny $\varphi : E_0 \to E$ by an arbitrary isogeny $\varepsilon : E \to F$ of some smooth degree e and work with $\varepsilon \circ \varphi$ instead of φ. This allows us to look for a smooth integer of the form $c = 2^{a-\alpha} - e3^{b-\beta}$ and construct a corresponding degree-c isogeny $\gamma : F \to C$.
- A second tweak can be obtained by any algorithm that can efficiently solve the following problem for a fixed d:
  - Let $H/\mathbb{F}_{p^2}$ be a genus-2 curve with superspecial Jacobian J, and $d > 1$ an integer. Is there a (d, d)-isogeny $\Psi : J \to A$ such that A is a product of elliptic curves?

  Indeed, this allows us to work with expressions of the form $c = d2^{a-\alpha} - e3^{b-\beta}$. Each test then amounts to computing a $(2^{a-\alpha}, 2^{a-\alpha})$-isogeny, using the torsion point data as before, and then checking if the resulting Jacobian is (d, d)-split. Verifying whether a given Jacobian is (d, d)-split is likely to be most efficient by means of a computation similar to those in [10, 19].

E.g., consider $a = 110$ and $b = 67$ as in `$IKEp217`, along with the identity

$$59 \cdot 67 \cdot 107 \cdot 443^2 \cdot 487 \cdot 1049 \cdot 2711 \cdot 8297 = 109 \cdot 2^{110-35} - 119 \cdot 3^{67-20}.$$

Assuming that we do not know a path from E_0 to E_{start}, we could still try to recover Bob's key by computing

- one-time isogenies $E \xrightarrow{\varepsilon} F \xrightarrow{\gamma} C$, dominated in cost by a 2711-isogeny and a 8297-isogeny over extension fields of respective degrees 1355 and 8297,
- computing all 3^{20}-isogenous neighbours of the base curve, gluing them together by means of a $(2^{75}, 2^{75})$-isogeny and checking which one of the resulting Jacobians is $(109, 109)$-split.

The second step immediately reveals the first 20 ternary digits of Bob's secret key and we can then easily find the remaining digits as explained above.

References

[1] Bradley Brock, *Superspecial curves of genera two and three*, Ph.D. thesis, Princeton University (1994)

[2] Nils Bruin, E. Victor Flynn, Damiano Testa, *Descent via (3,3)-isogeny on Jacobians of genus 2 curves*, Acta Arithmetica **165**(3), pp. 201-223 (2014)

[3] Wouter Castryck, Thomas Decru, *Multiradical isogenies*, Proceedings of AGC^{2}T18, Contemporary Mathematics **779**, pp. 57-89 (2022)

[4] Romain Cosset, Damien Robert, *Computing (ℓ, ℓ)-isogenies in polynomial time on Jacobians of genus 2 curves*, Mathematics of Computation **84**(294), pp. 1953-1975 (2015)

[5] Craig Costello, *B-SIDH: Supersingular isogeny Diffie-Hellman using twisted torsion*, Proceedings of Asiacrypt (2), Lecture Notes in Computer Science **12492**, pp. 440-463 (2020)

[6] Craig Costello, *The case for SIKE: a decade of the supersingular isogeny problem*, available at `https://eprint.iacr.org/2021/543` (2021)

[7] Luca De Feo, David Jao, Jérôme Plût, *Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies*, Journal of Mathematical Cryptology **8**, pp. 209-247 (2014)

[8] Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, Benjamin Wesolowski, *SQISign: compact post-quantum signatures from quaternions and isogenies*, Proceedings of Asiacrypt (1), Lecture Notes in Computer Science **12491**, pp. 64-93 (2020)

[9] Victoria de Quehen, Péter Kutas, Chris Leonardi, Chloe Martindale, Lorenz Panny, Christophe Petit, Katherine E. Stange, *Improved torsion-point attacks on SIDH variants*, Proceedings of Crypto (3), Lecture Notes in Computer Science **12827**, pp. 432-470 (2021)

[10] Martin Djukanovic, *Split Jacobians and lower bounds on heights*, Ph.D. thesis, Univ. Bordeaux (2017)

[11] E. Victor Flynn, Yan Bo Ti, *Genus two isogeny cryptography*, Proceedings of PQCrypto, Lecture Notes in Computer Science **11505**, pp. 286-306 (2019)

[12] Steven D. Galbraith, Christophe Petit, Javier Silva, *Identification protocols and signature schemes based on supersingular isogeny problems*, Proceedings of Asiacrypt (1), Lecture Notes in Computer Science **10624**, pp. 3-33 (2017)

[13] Steven D. Galbraith, Frederik Vercauteren, *Computational problems in supersingular elliptic curve isogenies*, Quantum Information Processing **17**(10), article no. 265, 22 pp. (2018)

[14] Everett W. Howe, Franck Leprévost, Bjorn Poonen, *Large torsion subgroups of split Jacobians of curves of genus two or three*, Forum Mathematicum **12**, pp. 315-364 (2000)

[15] David Jao, Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Aaron Hutchinson, Amir Jalali, Koray Karabina, Brian Koziel, Brian LaMacchia, Patrick Longa, Michael Naehrig, Geovandro Pereira, Joost Renes, Vladimir Soukharev, David Urbanik, *Supersingular Isogeny Key Encapsulation*, available at `https://sike.org/files/SIDH-spec.pdf`

[16] David Jao, Luca De Feo, *Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies*, PQCrypto, Lecture Notes in Computer Science **7071**, pp. 19-34 (2011)

[17] Ernst Kani, *The number of curves of genus two with elliptic differentials*, Journal für die reine und angewandte Mathematik **485**, pp. 93–121 (1997), available at `https://www.mast.queensu.ca/~kani/papers/numgenl.pdf`

[18] David Kohel, Kristin Lauter, Christophe Petit, Jean-Pierre Tignol, *On the quaternion ℓ-isogeny path problem*, LMS Journal of Computation and Mathematics **17**, pp. 418-432 (2014)

[19] Robert M. Kuhn, *Curves of genus 2 with split Jacobian*, Transactions of the American Mathematical Society **307**(1), pp. 41-49 (1988)

[20] Jonathan Love, Dan Boneh, *Supersingular curves with small non-integer endomorphisms*, Proceedings of ANTS-XIV, MSP Open Book Series **4**, pp. 7-22 (2020)

[21] Chloe Martindale, Lorenz Panny, *How to not break SIDH*, CFAIL 2019, available at `https://ia.cr/2019/558`

[22] Benjamin Smith, *Explicit endomorphisms and correspondences*, Ph.D. thesis, University of Sydney (2006)