A Note on a Static SIDH Protocol

Samuel Dobson and Trey Li and Lukas Zobernig

Department of Mathematics, University of Auckland, New Zealand

Abstract

It is well known, due to the adaptive attack by Galbraith, Petit, Shani, and Ti (GPST), that plain SIDH is insecure in the static setting. Recently, Kayacan's preprint *A Note on the Static-Static Key Agreement Protocol from Supersingular Isogenies*, ePrint 2019/815, presented two possible fixes. Protocol A (also known as 2-SIDH, a low-degree instantiation of the more general k-SIDH) has been broken by Dobson, Galbraith, LeGrow, Ti, and Zobernig. In this short note we will show how to break Protocol B in one oracle query per private key bit and $O(1)$ local complexity.

We will assume the readers to be familiar with the GPST attack [GPST16] on Supersingular Isogeny Diffie–Hellman (SIDH) [JF11, FJP14]. Kayacan proposed two possible countermeasures in [Kay19], called Protocol A and Protocol B. Protocol A is a specialisation of 2-SIDH [AJL17] and has recently been broken, see [DGL⁺19].

Protocol B is as follows. Choose a prime $p = \ell_A^n \cdot \ell_B^m \cdot f \pm 1$ (here f is some small cofactor), a supersingular elliptic curve $E/\mathbb{F}_{p^2}$, and generators P_A, Q_A and P_B, Q_B of $E[\ell_A^n]$ and $E[\ell_B^m]$, respectively. Split $n = f_A + g_A$ and $m = f_B + g_B$ such that $f_A \approx g_A$, $f_B \approx g_B$. Alice and Bob choose secrets $\alpha \in \mathbb{Z}/\ell_A^{f_A}\mathbb{Z}$ and $\beta \in \mathbb{Z}/\ell_B^{f_B}\mathbb{Z}$, respectively. The idea is that Alice and Bob complete the following commutative diagram.

$$\begin{tikzcd}
E \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & E_A \arrow[r, "\phi'_A"] \arrow[d, "\psi_B"] & E'_A \arrow[dd, "\Psi_B"] \\
E_B \arrow[r, "\psi_A"'] \arrow[d, "\phi'_B"'] & E_{AB} & \\
E'_B \arrow[rr, "\Psi_A"'] & & E'_{AB}
\end{tikzcd}$$

Here ϕ_A is the $\ell_A^{f_A}$-isogeny $\phi_A : E \to E_A = E/\langle [\ell_A^{g_A}](P_A + [\alpha]Q_A)\rangle$ which Alice uses to publish $K_A = \phi_A(P_A + [\ell_A^{n-1} + \alpha]Q_A)$, $R_B = \phi_A(P_B)$, and $S_B = \phi_A(Q_B)$, along with E_A. As always, Bob computes and publishes the mirrored information.

To finalize the key agreement, first Alice completes the inner diagram like in plain SIDH, to obtain E_{BA}. She then computes $\phi'_B : E_B \to E'_B = E_B/\langle K_B\rangle$, $U_A = \phi'_B(R_A)$, $V_A = \phi'_B(S_A)$, and $\Psi_A : E'_B \to E'_{BA} = E'_B/\langle U_A + [\ell_A^{n-1} + \alpha]V_A\rangle$. The shared key is given by $h = H(j(E_{BA})||j(E'_{BA}))$.

[Kay19] claims that this protocol is secure against an adversary that has access to an oracle that outputs the value $H(j(E_{BA})||j(E'_{BA}))$ upon completion of the protocol. We will now show that even an oracle that outputs 0 or 1 depending on whether both sides computed the same shared key h (Oracle_1 from [Kay19]) is sufficient to break the protocol in the static setting - that is, when Alice keeps her secret α fixed over multiple rounds.

In the following discussion we shall assume Bob is malicious and attempting to learn Alice's static secret key, and that $\ell_A = 2$, but the attack holds for either party and any choice of ℓ_A. The key observation is the following. If a malicious Bob modifies $R_A = \phi_B(P_A)$ and $S_A = \phi_B(Q_A)$ in his public key by adding torsion points, say X and Y, of small enough order, then this modification leaves the inner diagram unchanged. Formally, if $X, Y \in E[2^{g_A}]$ then

$$\langle [2^{g_A}](\phi_B(P_A + X) + [\alpha]\phi_B(Q_A + Y))\rangle = \langle [2^{g_A}](\phi_B(P_A) + [\alpha]\phi_B(Q_A))\rangle. \tag{1}$$

Let α_i denote the i-th bit (starting from 0) of a key α. We define the *i-th partial key* K_i of α as $K_i = \sum_{k=0}^{i-1} \alpha_k 2^k$ with which α can be written as $\alpha = K_i + \alpha' 2^i$ for some α'.

Suppose we have recovered the first i bits of the secret α. We proceed to learn the $(i+1)$-th bit. Bob generates $\phi_B : E \to E_B$ and K_B as per the protocol. He then calculates

$$\begin{aligned} R_A &= \phi_B(P_A - [K_i \cdot 2^{n-(i+1)}]Q_A) \\ S_A &= \phi_B(Q_A + [2^{n-(i+1)}]Q_A) \end{aligned}$$

and sends the public key (E_B, K_B, R_A, S_A) to Alice. Upon receipt of Alice's public key (E_A, K_A, R_B, S_B), Bob will complete his side of the protocol honestly to obtain the shared secret $h = H(j(E_{AB})||j(E'_{AB}))$.

Alice computes

$$\psi_A : E_B \to E_{BA} = E_B / \langle [2^{g_A}](R_A + \alpha S_A)\rangle$$

which completes the inner diagram. By Equation (1) ψ_A remains unchanged regardless of the value of α_{i+1}, so does E_{BA}. Alice proceeds to compute

$$\begin{aligned} U_A &= \phi'_B(R_A) \\ V_A &= \phi'_B(S_A) \\ \Psi_A &: E'_B \to E'_{BA} = E'_B / \langle U_A + [2^{n-1} + \alpha]V_A\rangle \end{aligned}$$

and eventually the shared secret $h' = H(j(E_{BA})||j(E'_{BA}))$. Note that whether $E'_{AB} \cong E'_{BA}$ (and hence whether $h = h'$) depends on whether α_{i+1} is 0 or 1:

$$\begin{aligned} U_A + [2^{n-1} + \alpha]V_A &= \phi'_B(R_A) + [2^{n-1} + \alpha]\phi'_B(S_A) \\ &= \phi'_B(R_A + [2^{n-1} + \alpha]S_A) \\ &= \phi'_B(\phi_B(P_A - [K_i \cdot 2^{n-(i+1)}]Q_A + [2^{n-1} + \alpha](Q_A + [2^{n-(i+1)}]Q_A))) \\ &= \phi'_B(\phi_B(P_A + [2^{n-1} + \alpha]Q_A + [2^{n-(i+1)}][\alpha - K_i]Q_A)) \\ &= \begin{cases} \phi'_B(\phi_B(P_A + [2^{n-1} + \alpha]Q_A)) & \text{if } \alpha_{i+1} = 0. \\ \phi'_B(\phi_B(P_A + [\alpha]Q_A)) & \text{if } \alpha_{i+1} = 1. \end{cases} \end{aligned}$$

This holds since Q_A has order 2^n. Thus if the shared secret Alice computes is equal to the one Bob computed ($h = h'$), the bit is 0, else it is 1. So Bob can learn this bit by querying Oracle_1.

Scaling to avoid pairing-based detection can be done as in GPST. The number of oracle queries Bob needs to learn the whole secret α of Alice is the bit-length f_A of α.

References

[AJL17] Reza Azarderakhsh, David Jao, and Christopher Leonardi, *Post-quantum static-static key agreement using multiple protocol instances*, SAC 2017, Springer, 2017, pp. 45–63.

[DGL+19] Samuel Dobson, Steven D. Galbraith, Jason LeGrow, Yan Bo Ti, and Lukas Zobernig, *An adaptive attack on 2-SIDH*, Cryptology ePrint Archive, Report 2019/890, 2019, https://eprint.iacr.org/2019/890.

[FJP14] Luca De Feo, David Jao, and Jérôme Plût, *Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies*, J. Mathematical Cryptology **8** (2014), no. 3, 209–247.

[GPST16] Steven D. Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti, *On the security of supersingular isogeny cryptosystems*, ASIACRYPT 2016, Springer, 2016, pp. 63–91.

[JF11] David Jao and Luca De Feo, *Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies*, PQCrypto 2011, Springer, 2011, pp. 19–34.

[Kay19] Selçuk Kayacan, *A note on the static-static key agreement protocol from supersingular isogenies*, Cryptology ePrint Archive, Report 2019/815, 2019, https://eprint.iacr.org/2019/815.