Supersingular Isogeny Oblivious Transfer

Paulo Barreto[1], Gláucio Oliveira[2], and Waldyr Benits[3]

[1] University of Washington, Tacoma, USA
pbarreto@uw.edu
http://directory.tacoma.uw.edu/employee/pbarreto
[2] Institute of Mathematics and Statistics, University of São Paulo, Brazil
glaucioaorj@gmail.com
[3] Naval Systems Analysis Center, Brazilian Navy, Brazil
wbenits@yahoo.com.br

Abstract We present an oblivious transfer (OT) protocol that combines the OT scheme of Chou and Orlandi [5] together with the supersingular isogeny Diffie-Hellman (SIDH) primitive of De Feo, Jao, and Plût [15]. Our construction is a candidate for post-quantum secure OT and demonstrates that SIDH naturally supports OT functionality. We consider the protocol in the simplest configuration of $\binom{2}{1}$-OT and analyze the protocol to verify its security.

Keywords: supersingular elliptic curves, isogenies, supersingular isogeny Diffie-Hellman, oblivious transfer.

1 Introduction

Most, if not all, of cryptography can be based on the notion of *Oblivious Transfer* (OT), under the assumption that an efficient such scheme is available. Efficient OT protocols are known in a *quantum*-susceptible scenario [5][4], where the underlying security assumption is the hardness of computing discrete logarithms or factoring integers. Additionally, many papers have introduced OT in the context of *quantum* cryptography [2,8,9,23,24,33,38], where the legitimate users manipulate quantum states. The *post-quantum* OT research has gradually increased over time. Thus, one of the other examples that could be cited is the work done by Kazmi [21].

In general, in an OT protocol, the sender wants to transfer one of possibly many pieces of information to a receiver, giving the receiver the choice of which information is transferred while remaining oblivious as to what information has been transferred. For instance, a sender sends two messages, say m_a and m_b, and the receiver chooses only one of them (for instance, the receiver chooses

[4] Not long ago, there was a supposed attack against scheme from [5]. However, a technical analysis by Claudio Orlandi seems to have not validated such attack. This analysis can be found in http://eprint.iacr.org/forum/read.php?18,962.

m_a). At the end of the protocol, the sender does not know which of the messages was chosen, and also the receiver learns nothing about the other message (in this case, m_b). Most actual OT proposals use a hybrid protocol similar to hybrid encryption in the public-key cryptography setting, in that a public-key cryptosystem, in practice, is needed only to bootstrap the initial transmission of a small piece of data.

This paper is organized as follows. In the section 2 describes our supersingular isogeny oblivious transfer (SIOT) proposal. Moreover, in Section 3 we discuss some analysis about security aspects from SIOT and finally we conclude this work in section 4. It should be noted that in appendix A, we introduce some crucial background used in this paper such as elliptic curve, torsion points, isogenies, distorsion map, modified Weil pairing and a basic concept of OT. In appendix B we show some definitions about the process that determines linearly independent points used by proposed protocol. Furthermore, In appendix C, we see a simplified form of the OT protocol from [5]. In appendix D, there is the possibility of applying a symmetric pairing in the security analysis of the SIOT protocol. Finally, in appendix E we will verify that the proposed protocol is able to share certain points that allow to execute the OT functionality.

Our contribuition. According to Hazay and Lindell [18], OT is one of the most important building blocks in cryptography and advantageous for constructing secure protocols. In addition, protocols with OT characteristics can be used in electronic auction environments and contract signing [13]. Thus, we implemented the OT functionality in the established SIDH protocol from [15], providing greater privacy between sender and recipient on a communication channel. Our main aim is to develop a post-quantum OT protocol to achieve quantum resistance.

2 The $\binom{2}{1}$ - SIOT protocol

In this section, we will see a new scheme called Supersingular Isogeny Oblivious Transfer (SIOT) protocol. It is fundamentally inspired on schemes from [5] and [15].

2.1 Notations

We use the cryptographic primitives of the Supersingular Isogeny Diffie-Hellman Key Exchange Protocol (SIDH) from [15]. In this way, the following notations below will be used.

i. $p \rightarrow$ A prime such that $p = 3 \bmod 4$;
ii. $\mathbb{F}_{p^2} \rightarrow$ A quadratic extension of $\mathbb{F}_p$, where $\mathbb{F}_{p^2} = \mathbb{F}_p[i]/\langle i^2 + 1\rangle$;
iii. $E_0(\mathbb{F}_{p^2}) \rightarrow$ A supersingular elliptic curve over $\mathbb{F}_{p^2}$;
iv. $\mathbb{Z}/\ell\mathbb{Z} \rightarrow$ A field of integers modulo ℓ, where ℓ is prime and $\ell \nmid p$;
v. $P_A, Q_A \rightarrow$ Points over the supersingular elliptic curve $E_A(\mathbb{F}_{p^2})$;
vi. $P_B, Q_B \rightarrow$ Points over the supersingular elliptic curve $E_B(\mathbb{F}_{p^2})$;

vii. $\phi_A, \phi_B \rightarrow$ Isogenies between E_0 and E_A, E_0 and E_B, respectively;
viii. $\phi_A^{'}, \phi_B^{'} \rightarrow$ Isogenies between E_B and E_{BA}, E_A and E_{AB}, respectively;
ix. $G_A, H_A \rightarrow$ Images of P_B and Q_B under Alice's private isogeny ϕ_A;
x. $G_B, H_B \rightarrow$ Images of P_A and Q_A under Bob's private isogeny ϕ_B;
xi. $j(E_{AB}) \rightarrow j - invariant$ of supersingular elliptic curve E_{AB};
xii. $r_A, r_B \rightarrow$ Points from $\mathbb{Z}/\ell_A\mathbb{Z}$ and $\mathbb{Z}/\ell_B\mathbb{Z}$, respectively;

2.2 Protocol

2.2.1 Public parameters

Let E_0 be a supersingular curve elliptic defined over $\mathbb{F}_{p^2}$. For convenience, assume a prime p of form[5] $p = \ell_A^{e_A}\ell_B^{e_B} - 1$ with $\ell_A = 2$ and $e_A \geqslant 4$ (and $f = 1$), or $p = 4\ell_A^{e_A}\ell_B^{e_B} - 1$, where both ℓ_A and ℓ_B are odd primes (and $f = 4$). Hence, either of these choices yield $p = 3 \pmod 4$, enabling the representation[6] $\mathbb{F}_{p^2} = \mathbb{F}_p[i]/\langle i^2 + 1\rangle$ and ensuring that the curve $E_0(\mathbb{F}_{p^2}) : y^2 = x^3 + x$ is supersingular, with group order $(\ell_A^{e_A}\ell_B^{e_B} f)^2$. Furthermore, let P_A, $Q_A \in E_0(\mathbb{F}_{p^2})$ be linearly independent points and a basis which generate $E_0[\ell_A^{e_A}]$. Similarly, this statement holds for other P_B, Q_B points which generate $E_0[\ell_B^{e_B}]$. In other words, $E_0[\ell_A^{e_A}]$ and $E_0[\ell_B^{e_B}]$ are generated by kernel $\langle P_A, Q_A\rangle$ and $\langle P_B, Q_B\rangle$, respectively. Finally, the appendix B presents some definitions used to compute such linearly dependent points. Before describing the proposed protocol, let's consider some premises in the next section.

2.2.2 Premises

As usual let us call "Alice" the sender and "Bob" the receiver, from now on.

1. $\mathcal{M}$ be a set of all plaintexts with binary strings of fixed length and $(x_0, x_1) \in \mathcal{M}$;
2. $\mathcal{C}$ be a set of all ciphertexts with binary strings of fixed length and $(c_0, c_1) \in \mathcal{C}$;
3. $Enc(j; x)$ denote a symmetric encryption scheme taking a shared symmetric key j (presumably the j-invariant of some shared supersingular elliptic curve) and a plaintext x;
4. Alice wants to send Bob two messages x_0, x_1 of her choice. Bob gets to choose one of them, but does not want Alice to know his choice;
5. Alice and Bob use a secure coin-flipping protocol to agree on a uniformly random bit string w that is unique for each session (this ensures that neither Alice not Bob can guess beforehand or control the value of w). This can be achieved with e.g. Wagner's bit commitment protocol [35];
6. They also agree on a generic scheme to encode w as a pair of points U, $V \in E_B[\ell_A^{e_A}]$ using the agreed upon deterministic technique.[7]

[5] Suppose ℓ_A and ℓ_B are small primes, and f is a cofactor such that p is prime. In SIDH protocol, the primes allow the curve to have smooth order so that the isogenies can be computed quickly. See [27] that reports a deep research about the choice of SIDH-Friendly Primes.

[6] For more details about this type of representation, see [19].

[7] Since the discrete logarithm problem is easy (even classically) in the relevant elliptic groups, a sensible method to attain this is to simply hash w to the coefficients of the

2.2.3 Abstract view of information exchange from $\binom{2}{1}$ - SIOT protocol

Alice		**Bob**
Input: $x_0, x_1 \in \mathcal{M}$		Input: $b \in \{0,1\}$
Output: None		Output: x_b
$r_A \in \mathbb{Z}/l_A^{e_A}\mathbb{Z}$		$r_B \in \mathbb{Z}/l_B^{e_B}\mathbb{Z}$
$\phi_A : E_0 \to E_A$		$\phi_B : E_0 \to E_B$
$G_A \leftarrow \phi_A(P_B); H_A \leftarrow \phi_A(Q_B)$		$G_B \leftarrow \phi_B(P_A); H_B \leftarrow \phi_B(Q_A)$
$\ker(\phi_A) = \langle P_A + r_A Q_A \rangle$		$\ker(\phi_B) = \langle P_B + r_B Q_B \rangle$
$\mathsf{pk}_A \leftarrow (E_A, G_A, H_A)$	$\xrightarrow{\mathsf{pk}_A}$	If $G_A, H_A \notin E_A[l_B^{e_B}]$, then abort.
		$U, V \leftarrow_{\$} E_B[l_A^{e_A}]$ then,
		$\hat{G}_B \leftarrow (G_B - bU)$
		$\hat{H}_B \leftarrow (H_B - bV)$
If $\hat{G}_B, \hat{H}_B \notin E_B[l_A^{e_A}]$, then abort.	$\xleftarrow{\hat{\mathsf{pk}}_B}$	$\hat{\mathsf{pk}}_B \leftarrow (E_B, \hat{G}_B, \hat{H}_B)$
$\forall i \in \{0,1\}$ and $U, V \leftarrow_{\$} E_B[l_A^{e_A}]$ then,		
$\ker(\phi'_{A_i}) = \langle (\hat{G}_B + iU) + r_A(\hat{H}_B + iV) \rangle$		
$\phi'_{A_i} : E_B \to E_{BA_i}$		
$j_i \leftarrow j(E_{BA_i})$		
$c_i \leftarrow \mathsf{Enc}(j_i, x_i)$	$\xrightarrow{(c_0, c_1)}$	$\ker(\phi'_B) = \langle G_A + r_B H_A \rangle$
		$\phi'_B : E_A \to E_{AB}$
		$j_b \leftarrow j(E_{AB})$
		$x_b \leftarrow \mathsf{Enc}^{-1}(j_b, c_b);$

Figure 1. $\binom{2}{1}$ - SIOT protocol.

chosen elliptic group basis (rather than resorting to, say, the scheme of Shallue and van de Woestijne [29]).

2.2.4 Generation of key pairs

2.2.4.1 Setup - Sender

1. Alice secretly chooses a value $r_A \in Z/\ell_A^{e_A}Z$;
2. Computes:
   (a) $\phi_A : E_0 \to E_A$;
   (b) $\phi_A(P_B) = G_A$; $\phi_A(Q_B) = H_A$;
   (c) $Ker(\phi_A) = \langle P_A + r_A Q_A \rangle$.
3. Alice creates a key pair $sk_A = (\phi_A, r_A)$; $pk_A = (E_A, G_A, H_A)$;
4. Alice sends to Bob $pk_A = (E_A, G_A, H_A)$. He checks if $G_A, H_A \in E_A[\ell_B^{e_B}]$. If so, he accepts the received public key. Otherwise, the public key is refused.

Remark 2.2.4.1. Recalling that $E_A = E_0/\langle P_A + r_A Q_A \rangle$ and $|ker(\phi_A)| = |\langle P_A + r_A Q_A \rangle| = \ell_A^{e_A}$, *i.e*, degree-$\ell_A^{e_A}$ isogeny ϕ_A.

Remark 2.2.4.2. Alice computes the projection $\{\phi_A(P_B), \phi_A(Q_B)\} \subset E_A$ of the basis $\{P_B, Q_B\} \subset E_0[\ell_B^{e_B}]$ under her secret isogeny ϕ_A. Moreover, notice that $\ell_B^{e_B} G_A = \ell_B^{e_B} H_A = \mathcal{O}_A \in E_A$.

2.2.4.2 Setup - Receiver

1. Bob secretly chooses a value $r_B \in Z/\ell_B^{e_B}Z$;
2. Computes:
   (a) $\phi_B : E_0 \to E_B$
   (b) $\phi_B(P_A) = G_B$; $\phi_B(Q_A) = H_B$;
   (c) $Ker(\phi_B) = \langle P_B + r_B Q_B \rangle$.
3. Computes: $\hat{G}_B = G_B - bU$; $\hat{H}_B = H_B - bV$;
4. Bob creates a public key $\hat{pk}_A = (E_B, \hat{G}_B, \hat{H}_B)$ and sends to Sender.

Remark 2.2.4.3. Recalling that $E_B = E_0/\langle P_B + r_B Q_B \rangle$ and $|ker(\phi_B)| = |\langle P_B + r_B Q_B \rangle| = \ell_B^{e_B}$, *i.e*, degree-$\ell_B^{e_B}$ isogeny ϕ_B.

Remark 2.2.4.4. Bob computes his projection $\{\phi_B(P_A), \phi_B(Q_A)\} \subset E_B$ of the basis $\{P_A, Q_A\} \subset E_0[\ell_A^{e_A}]$ under his secret isogeny ϕ_B. Moreover, notice also that $\ell_A^{e_A} G_B = \ell_A^{e_A} H_B = \mathcal{O}_B \in E_B$.

Remark 2.2.4.5. $\forall b \in \{0, 1\}$, it should be noted that $b = 0$ then $\hat{G}_B = G_B$; $\hat{H}_B = H_B$ and $\hat{G}_B = (G_B - U)$; $\hat{H}_B = (H_B - V)$ if $b = 1$.

Remark 2.2.4.6. It should be noted that $U, V, G_B, H_B \in E_B[\ell_A^{e_A}]$ then, they can be written as a linear combination, i.e, $U = \alpha G_B + \beta H_B$ and $V = \gamma G_B + \delta H_B$ for unique $\alpha, \beta, \gamma, \delta \in Z/\ell_A^{e_A}Z$. In the implementation of the proposed protocol, a priori, we are not worried about implementing a subprotocol originating from [29] or [35], as mentioned earlier on section 2.2.2. In other words, we have implemented a simpler deterministic technique for getting α, β, δ and γ values.

Remark 2.2.4.7. Alice and Bob encode the shared bit string w as a pair of points $U, V \in E_B[\ell_A^{e_A}]$ using the agreed upon deterministic technique. This ensures that the pair of points $(G_B + U, H_B + V)$ and $(G_B - U, H_B - V)$ are generated by $E_B[\ell_A^{e_A}]$. If they do not, restarts the whole protocol by choosing a different string w.

2.2.5 Generation of secret keys $j_i = E_{BA_i}$ such that $i \in \{0, 1\}$

2.2.5.1 Setup - Sender

1. Upon reception of $(E_B, \hat{G}_B, \hat{H}_B)$, Alice checks if $\hat{G}_B, \hat{H}_B \in E_B[\ell_A^{e_A}]$. If so, she accepts the received public key. Otherwise, the public key is refused;
2. Alice computes:
   (a) $\forall i \in \{0, 1\}\ \phi'_{A_i} : E_B \to E_{BA_i}$, $Ker(\phi'_{A_i}) = \langle(\hat{G}_B + iU) + r_A(\hat{H}_B + iV)\rangle$;
   (b) $j_i \leftarrow j{-}invariant(E_{BA_i})$. (1)

2.2.5.2 Setup - Receiver

1. Bob computes:
   (a) $\phi'_B : E_B \to E_{AB}, Ker(\phi'_B)\langle G_A + r_B H_A\rangle$;
   (b) $j_b \leftarrow j{-}invariant(E_{AB})$ such that $b \in \{0, 1\}$. (2)

2.2.6 Encryption and Decryption

1. Alice encrypts $c_i \leftarrow Enc(j_i, x_i)$ such that $i \in \{0, 1\}$ and sends to Bob $c = (c_0, c_1)$;
2. Bob decrypts and gets $x_b = Enc^{-1}(j_b, c_b)$ such that $b \in \{0, 1\}$.

Remark 2.2.6.1. From equations (1) and (2) above, it is verified that a key $j(E_{BA_i}) = j(E_{AB})$, if $b = i$ and if $b \neq i$ then $j(E_{BA_i}) \neq j(E_{AB})$ such that $b, i \in \{0, 1\}$. Hence, at the end of the protocol if both parts are honest then we have that $j_b = j_i$. Therefore, we can conclude that if the Bob chooses a unique b, then it will share a unique secret key with Alice.

3 Analysis of the SIOT protocol

3.1 Supersingular Isogeny Problems

In this section, we will see some cases of computational problems from supersingular elliptic curves that were adapted by [15]. Such problems reinforce the security of the proposed protocol. Therefore, let a supersingular curve E_0 over $\mathbb{F}_{p^2}$ together with independent bases $\{P_A, Q_A\}$ and $\{P_B, Q_B\}$ of $E_0[\ell_A^{e_A}]$ and $E_0[\ell_B^{e_B}]$, respectively. Furthermore, recall p be a prime of the form defined on section 2.2.1.

Problem 1 (Decisional Supersingular Isogeny (DSSI) problem). Let $E_B(\mathbb{F}_{p^2})$ be another supersingular curve. Decide whether E_B is $\ell_B^{e_B}$-isogenous to E_0. In this case, we could suppose that even if we knew that E_0 and E_B had the same cardinality by Tate's theorem [32], this would not be enough to decide correctly if there is an isogeny of degree $\ell_B^{e_B}$ between them.

Problem 2 (Computational Supersingular Isogeny (CSSI) problem). Let $\phi_A : E_0 \to E_A$ be an isogeny whose kernel is $\langle P_A + [r_A]Q_A \rangle$ for some $r_A \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$. Given E_A and the values $\phi_A(P_B)$, $\phi_A(Q_B)$ then, find a generator R_A of $\langle P_A + [r_A]Q_A \rangle$. In this case, we could suppose that given only a public key $(E_A, \phi_A(P_B), \phi_A(Q_B))$ then, we won't be to able to determine any value from kernel because we don't know the corresponding private key (ϕ_A, r_A).

Problem 3 (Supersingular Computational Diffie-Helmann (SSCDH) problem). Let $\phi_A : E_0 \to E_A$ be an isogeny whose kernel is $\langle P_A + [r_A]Q_A \rangle$ for some $r_A \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ and let $\phi_B : E_0 \to E_B$ be an isogeny whose kernel is $\langle P_B + [r_B]Q_B \rangle$ for some $r_B \in \mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$. Given E_A, E_B and the values G_A, H_A,G_B, H_B then, find the $j - invariant$ of $E_0/\langle P_A + [r_A]Q_A, P_B + [r_B]Q_B \rangle$. In this case, we could assume that given only the two public keys (E_A, G_A, H_A) and (E_B, G_B, H_B) and not knowing any of their respective private keys (ϕ_A, r_A) and (ϕ_B, r_B) then, it will be impracticable to compute the value of a $j - invariant$.

In the next sections, we will present some notations and definitions for security analysis of proposed protocol. After that, we will check some minimal requirements that are important to make such protocol secure.

3.2 Notations

In our analysis of the proposed protocol, the following notations below will be used.

i. Application of Vélu's formula[8] $\to V\acute{e}lu's\ formula\{\langle G_A + r_B H_A \rangle, E_A\}$, where parameters E_A, G_A, H_A and r_B are denoted in section 2.1;

ii. The view of Alice in an execution, for a two-party protocol with parties Sender (Alice) and Receiver (Bob) $\to \{VIEW_{Alice}(Alice(1^n, \tau), Bob(1^n, b))\}$, where Alice has input τ, Bob has input b, and the security parameter is 1^n;

Remark 3.2.0.1. Similarly, we denote the view of Bob by $\{VIEW_{Bob}(Alice(1^n, \tau), Bob(1^n, b))\}$.

iii. Both Alice and Bob dishonest $\to Alice^*$, Bob^*, respectively.

3.3 Preliminaries

Definition 3.3.0.1. *A function $\epsilon(\cdot)$ is negligible in n, or just negligible, if for every positive polynomial $p(\cdot)$ and all sufficiently large ns it holds that $\epsilon(\cdot) < 1/p(\cdot)$.*

Remark 3.3.0.1. For example, the functions $2^{-\sqrt{n}}$ and $n^{-\log_2 n}$ are negligible as function in n. For more details, see [17].

[8] See proposition A.1.5.2

Definition 3.3.0.2. *A probability ensemble* $\mathcal{X} = \{\mathcal{X}(n,a)\}$ *is an infinite sequence of random variables indexed by* $n \in \mathbb{N}$ *and* $a \in \{0,1\}^*$. *The value* n *will represent the security parameter and* a *will represent the parties' inputs.*

Definition 3.3.0.3. *Two distribution ensembles* $\mathcal{X} = \{\mathcal{X}(n,a)\}$ *and* $\mathcal{Y} = \{\mathcal{Y}(n,a)\}$ *are said to be computationally indistinguishable, denoted by* $\mathcal{X} \overset{\mathbf{c}}{\equiv} \mathcal{Y}$, *if for every non-uniform polynomial-time algorithm* $\mathcal{D}$ *there exists a negligible function* $\epsilon(\cdot)$ *such that for every* $n \in \mathbb{N}$ *and* $a \in \{0,1\}^*$,

$$|\Pr[\mathcal{D}(\mathcal{X}(n,a)) = 1] - \Pr[\mathcal{D}(\mathcal{Y}(n,a)) = 1]| \leqslant \epsilon(n)$$

Remark 3.3.0.2. All parties are assumed to run in time that is polynomial in the security parameter which value 1^n is written.

3.4 Some analysis requirements

A priori, secure protocols should resist any adversarial attack. Even, Goldreich and Lempel [13] developed a form $\binom{2}{1}$- OT to build protocols for secure multiparty computation. Thus, in order to formally prove that a protocol is secure, say an OT protocol, Hazay and Lindell [18] mentions some required properties such as *privacy*, *correctness*, *independence of inputs*, *guaranteed output delivery* and *fairness*. Evidently, such list does not constitute a rigorous definition of security, but rather a set of requirements that should hold for any secure protocol. Moreover, they also state that the two most important requirements in any secure computing protocol are privacy and *correctness*. Thus, we will see below these requirements for the security analysis of the protocol proposed in this work.

3.4.1 Correctness

Each party should have a guarantee that the output he or she receives from the protocol is correct . We assume that both parties, say Alice and Bob, should be honests, then it will be possible to compute a shared secret in the end of protocol. In other words, only one of two messages (x_0 or x_1) should be decrypted, say, $x_b \leftarrow \mathsf{Enc}^{-1}(j_b, c_b)$ without loss of generality. Therefore, it should be true that $j(E_{BA_i}) = j(E_{AB})$, *i.e*, a $j - invariant$ computed by Alice will be exactly the same as that computed by Bob. In figure 2, the pseudocode shows that this holds, since if the value b, secretly chosen by Bob, is equal to the value i from Alice then, there will be a secret value shared between Alice and Bob. Otherwise, there will be no shared secret.

3.4.2 Privacy

Informally speaking, each party should only learn its intended output from the protocol and nothing else. In the protocol, the choices of Bob shouldn't be known to Alice. At the end of the protocol, Bob can't get any knowledge about the message that he did not decrypt. Formally, we shall see below a suitable and analogous definition from Hazay and Lindell [18] applied to the proposed protocol.

for $b, i \in \{0,1\}$ **do**
  if $b = i$ **then**
    $j(E_{BA_i}) \;=\; j(E_B/\langle G_B + r_A H_B\rangle) \;=\; j(E_{AB}) = j(E_A/\langle G_A + r_B H_A\rangle)$;
    return "*Shared secret*"
  elseif $b \neq i$ **then**
    $j(E_{BA_i}) \;=\; j(E_B/\langle (G_B - bU + iU) + r_A(H_B - bV + iV)\rangle) \;\neq\; j(E_{AB}) =$
    $j(E_A/\langle G_A + r_B H_A\rangle)$;
    return "*No shared secret*"
  endfor

Figure 2. Pseudocode of information exchange between two parties of the SIOT protocol.

Definition 3.4.2.1. *A two-message two-party probabilistic polynomial - time protocol (Alice, Bob) is said to be a private oblivious transfer if the following holds:*

i. ***Non-triviality****: If Alice and Bob follow the protocol then after an execution in which Alice has for input any x_0, $x_1 \in \mathcal{M}$[9] and Bob has input bit $b \in \{0,1\}$, the output of Bob is x_b. In other words, Bob receives pk_A and the pair (c_0, c_1) from Alice. Recalling that $pk_A \leftarrow (E_A, G_A, H_A)$ and $c_b \leftarrow Enc(j_b, x_b)$ are well defined. Thus, non-triviality follows from the fact that*

$$V\acute{e}lu's\, formula\{\langle G_A + r_B H_A\rangle, E_A\} \Rightarrow E_{AB} \therefore$$
$$j(E_{AB}) \Rightarrow j_{AB}.$$

Therefore, $x_b \leftarrow Enc^{-1}(j_{AB}, c_b)$ such that $j_{AB} = j_b$ and b is an unique value secretly chosen by Bob.

Remark 3.4.2.1. Upon receiving pk_A, say Alice's public key, Bob will not be able to compute her private key, say (ϕ_A, r_A). If that could be possible, there would be a violation of the CSSI problem (problem 2).

ii. ***Privacy in the case of a dishonest Alice:*** *Note that this requirement is that Alice's view when Bob has input 0 is indistinguishable from its view when Bob has input 1. In other words, the view of a supposed adversarial, say Alice, consists merely of public key $\hat{pk}_B$. Hence, recalling figure 1, we can see that:*

Lemma 1. *Alice on input $\hat{pk}_B$ cannot guess b with probability greater than $1/2 + \epsilon(n)$, for some negligible function $\epsilon(n)$ and $\forall\; n \in \mathbb{N}$*

[9] Recall $\mathcal{M}$ be the set of all plaintexts with binary strings of fixed length.

Proof. We can assume that on receiving $\hat{pk}_B$ *and not knowing the value of Bob's bit b, Alice cannot distinguish the pairs of tuples* $\{(E_B, G_B, H_B)\}$ *and* $\{(E_B, (G_B - U), (H_B - V)\}$*, i.e, for some* $\hat{G}_B, \hat{H}_B \in E_B[\ell_A^{e_A}]$ *such that* $\Pr[(E_B, G_B, H_B) = (E_B, \hat{G}_B, \hat{H}_B)] = \Pr[(E_B, G_B - U, H_B - V) = (E_B, \hat{G}_B, \hat{H}_B))]$ *which is independent of b . Thus, this case is similar to the decisional Diffie-Hellman (DDH) assumption. Hence, analogously considering such assumption, we have that:*

$$\{(E_B, G_B, H_B)\} \stackrel{\mathbf{c}}{\equiv} \{E_B, \mathring{G}_B, \mathring{H}_B\}^{10}$$

Now, assume by contradiction that there exists a probabilistic polynomial-time distinguisher[11] $\mathcal{D}$ *and a non-negligible function* ϵ *such that for every* n

$$|P_r[\mathcal{D}(E_B, G_B, H_B) = 1] - P_r[\mathcal{D}(E_B, \mathring{G}_B, \mathring{H}_B) = 1]| \geqslant \epsilon(n)$$

where $G_B, H_B \in E_B[\ell_A^{e_A}]$*. Then, by subtracting and adding*

$$P_r[\mathcal{D}(E_B, \tilde{G}_B, \tilde{H}_B) = 1]^{12}$$

We have

$$\begin{gathered}|P_r[\mathcal{D}(E_B, G_B, H_B) = 1] - P_r[\mathcal{D}((E_B, \mathring{G}_B, \mathring{H}_B) = 1]| \leqslant \\ |P_r[\mathcal{D}(E_B, G_B, H_B) = 1] - P_r[\mathcal{D}(E_B, \tilde{G}_B, \tilde{H}_B) = \\ 1]| + |P_r[\mathcal{D}(E_B, \tilde{G}_B, \tilde{H}_B) = 1] - P_r[\mathcal{D}(E_B, \mathring{G}_B, \mathring{H}_B) = 1]|\end{gathered}$$

Thus, by the contradicting assumption,

$$|P_r[\mathcal{D}(E_B, G_B, H_B) = 1] - P_r[\mathcal{D}(E_B, \tilde{G}_B, \tilde{H}_B) = 1]| \geqslant \frac{\epsilon(n)}{2} \tag{3.1}$$

or

$$|P_r[\mathcal{D}(E_B, \tilde{G}_B, \tilde{H}_B) = 1] - P_r[\mathcal{D}(E_B, \mathring{G}_B, \mathring{H}_B) = 1]| \geqslant \frac{\epsilon(n)}{2} \tag{3.2}$$

Let that (4.1) holds. Thus, we can construct a distinguisher $\tilde{\mathcal{D}}$ *for the DDH assumption that works as follow. Upon input* $\hat{pk}_B = \{(E_B, \mathring{G}_B, \mathring{H}_B)\}$*, the*

[10] Let $\mathring{G}_B = (G_B - U)$ and $\mathring{H}_B = (H_B - V)$, where $U, V \in E_B[\ell_A^{e_A}]$.

[11] Informally, a distinguisher is an algorithm that describes an adversary's advantage. In addition, [28] explains that a distinguisher is just an algorithm, possibly a probabilistic one, equipped with way to interact with its environment.

[12] Let $\tilde{G}_B = (\mathring{G}_B - R)$ and $\tilde{H}_B = (\mathring{H}_B - S)$, where $R, S \in E_B[\ell_A^{e_A}]$.

distinguisher $\tilde{\mathcal{D}}$ randomly chooses the pair of points $R, S \in E_B[\ell_A^{e_A}]$. Then, $\hat{pk}'_B = \{(E_B, \tilde{G}_B, \tilde{H}_B)\}$. On the other hand, if $\hat{pk}_B = \{(E_B, G_B, H_B)\}$ then $\hat{pk}'_B = \{E_B, (G_B - R), (H_B - S)\}$. Note that the pairs of points U and V are not used in the last tuple $\hat{pk}'_B$. However, these points as points R and S from to the same elliptic group $E_B[\ell_A^{e_A}]$ and could also be randomly chosen by $\tilde{\mathcal{D}}$, say U and V. Thus, we have that $\hat{pk}'_B = \{(E_B, G_B, H_B)\}$ and

$$|P_r[\tilde{\mathcal{D}}(E_B, G_B, H_B) = 1] - P_r[\tilde{\mathcal{D}}(E_B, \mathring{G}_B, \mathring{H}_B) = 1]| =$$
$$|P_r[\mathcal{D}(E_B, G_B, H_B) = 1] - P_r[\mathcal{D}(E_B, \tilde{G}_B, \tilde{H}_B) = 1]| \geqslant \frac{\epsilon(n)}{2}$$

in contradiction to the DDH assumption. An analogous analysis follows in the case where (4.2) holds. The proof of Bob's privacy is concluded by noting that $\{(E_B, G_B, H_B)\}$, $\{(E_B, \mathring{G}_B, \mathring{H}_B)\}$, regardless of the value of b, are indistinguishable in Alice's view. In other words, let $\tau \in \{0,1\}^*$ be an auxiliary input. Thus,

$\{VIEW_{Alice^*}(Alice^*(1^n, \tau), Bob(1^n, 0))\} \overset{\mathbf{c}}{\equiv} \{VIEW_{Alice^*}(Alice^*(1^n, \tau), Bob(1^n, 1))\}$

Therefore, the privacy of Bob follows from analogous DDH assumption over the elliptic group $E_B[\ell_A^{e_A}]$. □

iii. ***Privacy in the case of a dishonest Bob:*** *Let $\hat{pk}_B \leftarrow (E_B, \hat{G}_B, \hat{H}_B)$ denotes Bob's public key sent to Alice. Recall that $\hat{G}_B \leftarrow G_B$, $\hat{H}_B \leftarrow H_B$, if $b = 0$ and $\hat{G}_B \leftarrow (G_B - U)$, $\hat{H}_B \leftarrow (H_B - V)$, if $b = 1$. Moreover, the Alice's public key, say $pk_A := (E_A, G_A, H_A)$, sent to Bob and an unique $j-invariant$ $j_b = j(V\acute{e}lu's\ formula\{\langle G_A + r_B H_A \rangle, E_A\})$ computed by him upon receiving pk_A are well defined. After that, $\forall\ i \in \{0,1\}$, Alice will compute $j_i = j(V\acute{e}lu's\ formula\{\langle(\hat{G}_B + iU) + r_A(\hat{H}_B + iV)\rangle, E_B\})$, i.e, j_0 and j_1. In addition, recalling the correctness requirement from subsection 3.4.1, Alice will share an unique secret key with Bob. Therefore, the Alice's privacy is based on the following lemma:*

Lemma 2. *Bob can't compute two $j-invariants$ j_0 and j_1 $(j_0 \neq j_1)$ whether SSCDH problem is hard.*

Proof. If Bob could compute j_0 and j_1 then it would be a violation of the SSCDH problem (problem 3). In other words, Bob will be able to compute just an unique $j-invariant$, depending on the chosen value of b. □

Complementing the lemma proof above, let $b \in \{0,1\}$ be an auxiliary input and every triple of inputs x_0, x_1, $x \in \mathcal{M}$[13]. Thus, another way to view the Alice's privacy is that Bob's first message, denoted by Bob $(1^n, b)$, determines whether it should receive x_0 or x_1. For example, if it determines that*

[13] Recall $\mathcal{M}$ be the set of all plaintexts with binary strings of fixed length.

it should receive x_0, then its view when Alice's input is (x_0, x_1) is indistinguishable from its view when Alice's input is (x_0, x). Evidently, this implies that Bob cannot learn anything about x_1 when it receives x_0 and vice versa. Hence,

$$\{VIEW_{Bob^*}(Alice(1^n,(x_0,x_1));Bob^*(1^n,b))\}_{n\in\mathbb{N}} \overset{\mathbf{c}}{\equiv} \{VIEW_{Bob^*}(Alice(1^n, (x_0,x));Bob^*(1^n,b))\}_{n\in\mathbb{N}}$$

or

$$\{VIEW_{Bob^*}(Alice(1^n,(x_0,x_1));Bob^*(1^n,b))\}_{n\in\mathbb{N}} \overset{\mathbf{c}}{\equiv} \{VIEW_{Bob^*}(Alice(1^n, (x,x_1));Bob^*(1^n,b))\}_{n\in\mathbb{N}}$$ □

After analyzing all the requirements from definition 3.4.2.1 on the SIOT protocol and recalling p be a prime of the form defined on section 2.2.1, we can formulate the following theorem,

Theorem 3.4.1. *Assume that an analogous decisional Diffe-Hellman (DDH) assumption and a Supersingular Computational Diffe-Hellman (SSCDH) problem are hard in an elliptic group $E(F_{p^2})$. Then, SIOT protocol is a private oblivious transfer as in definition 3.4.2.1.*

3.5 Further distinguisher and other analyzes

So far, we have demonstrated that SIOT protocol guarantees privacy between sender (Alice) and receiver (Bob). In addition, it should be recalled that the proposed protocol is based on the arithmetic of supersingular elliptic curves from [15], *i.e*, the structure of The SIOT protocol inherits the security features from SIDH protocol. However, we still consider it necessary to analyze the security of SIOT protocol. Thus, considering the case of a dishonest Alice, she will use a pairing-based distinguisher for trying to find out the secret value b from honest Bob. In the second situation, the roles will be inverted, *i.e*, Alice will be considered an honest sender and Bob a dishonest receiver. In the latter case, an analysis is performed in such a way that some algebraic conditions must be obeyed so that Bob is not able to decipher both Alice's messages.

3.5.1 Preventing a pairing-based distinguisher from a possible Alice's dishonesty

Considering the situation where Alice (the dishonest sender) receiving the information $(E_B, \hat{G}_B, \hat{H}_B)$ from Bob, a priori, does not know whether to receive (E_B, G_B, H_B) or $(E_B, G_B - U, H_B - V)$. Alice might consider using the Weil pairing to distinguish between these two values.

In what follows, all pairings have order $\ell_A^{e_A}$. After all, the correct points $G_B = \phi_B(P_A)$ and $H_B = \phi_B(Q_A)$ are known to satisfy $e(G_B, H_B) = e(P_A, Q_A)^{\ell_A^{e_A}}$: if this relation does not hold for both of $(\hat{G}_B, \hat{H}_B)$ or $(\hat{G}_B + U, \hat{H}_B + V)$, it would reveal which key Bob chose. More generally, because Alice can add any multiple

of (U, V) to $(\hat{G}_B, \hat{H}_B)$ and look for such a mismatch, one must have $e(\hat{G}_B + \lambda U, \hat{H}_B + \lambda V) = e(G_B, H_B) = e(P_A, Q_A)^{\ell_A^{e_A}}$ for any $\lambda \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$. Recalling[14] that $U = \alpha G_B + \beta H_B$, $V = \gamma G_B + \delta H_B$, this condition means that:

$$\begin{aligned}
e(G_B + \lambda U, H_B + \lambda V) &= e(G_B + \lambda\alpha G_B + \lambda\beta H_B, H_B + \lambda\gamma G_B + \lambda\delta H_B) \\
&= e((1+\lambda\alpha)G_B + \lambda\beta H_B, \lambda\gamma G_B + (1+\lambda\delta)H_B) \\
&= e((1+\lambda\alpha)G_B, \lambda\gamma G_B) \\
&\cdot\ e((1+\lambda\alpha)G_B, (1+\lambda\delta)H_B) \\
&\cdot\ e(\lambda\beta H_B, \lambda\gamma G_B) \\
&\cdot\ e(\lambda\beta H_B, (1+\lambda\delta)H_B) \\
&= 1 \\
&\cdot\ e(G_B, H_B)^{(1+\lambda\alpha)(1+\lambda\delta)} \\
&\cdot\ e(H_B, G_B)^{\lambda\beta\lambda\gamma} \\
&\cdot\ 1 \\
&= e(G_B, H_B)^{(1+\lambda\alpha)(1+\lambda\delta)-\lambda^2\beta\gamma} \\
&= e(G_B, H_B),
\end{aligned}$$

hence it is necessary that $(1+\lambda\alpha)(1+\lambda\delta)-\lambda^2\beta\gamma = 1 \pmod{\ell_A^{e_A}}$, or equivalently $\lambda(\alpha+\delta) + \lambda^2(\alpha\delta - \beta\gamma) = 0 \pmod{\ell_A^{e_A}}$. This must hold for *any* choice of λ, in particular those that are invertible mod $\ell_A^{e_A}$, and hence it must hold that $\lambda(\alpha\delta - \beta\gamma) = -(\alpha+\delta) \pmod{\ell_A^{e_A}}$. Once more, this can only hold for *any* λ if $\alpha\delta - \beta\gamma = 0 \pmod{\ell_A^{e_A}}$ and $\alpha + \delta = 0 \pmod{\ell_A^{e_A}}$, or equivalently, $\delta = -\alpha \pmod{\ell_A^{e_A}}$ and $\alpha^2 + \beta\gamma = 0 \pmod{\ell_A^{e_A}}$. Therefore, in principle, such conditions should be obeyed to avoid Alice finds out Bob's choice. □

3.5.2 Possible decryptions from a possible Bob's dishonesty

Recalling $U, V \in E_B[\ell_A^{e_A}]$ be linearly independent points, and write $U = \alpha G_B + \beta H_B$, $V = \gamma G_B + \delta H_B$. Suppose Alice receives an information $(E_B, \hat{G}_B, \hat{H}_B)$ from Bob. Then, Alice will compute actually the degree-$\ell_A^{e_A}$ isogeny $\phi'_{A_0} : E_B \to E_{BA_0}$ whose kernel is $ker(\phi'_{A_0}) = \langle G_B + r_A H_B \rangle$ and $\phi'_{A_1} : E_B \to E_{AB_1}$ whose kernel is $ker(\phi_{A_1}) = \langle (G_B + U) + r_A(H_B + V) \rangle$. It should be noted[15] that if $ker(\phi'_{A_0}) \subseteq ker(\phi_{A_1})$ then, E_{BA_0} is isomorphic to E_{BA_1} *i.e.*, $E_{BA_0} \cong E_{BA_1}$. Moreover, if ϕ_{A_1} is separable then there is a unique isogeny $\hat{\phi_A} : E_{BA_0} \to E_{BA_1}$. Now $(G_B + U) + r_A(H_B + V) = (G_B + \alpha G_B + \beta H_B) + r_A(H_B + \gamma G_B + \delta H_B) = (1 + \alpha + \gamma r_A)G_B + (r_A + \beta + \delta r_A)H_B$. Hence, by inspection, this point can only be in $\langle G_B + r_A H_B \rangle$ with the following conditions:

1. $(1 + \alpha + \gamma r_A)$ is invertible mod $\ell_A^{e_A}$ (i.e. if $\ell_A \nmid 1 + \alpha + \gamma r_A$);

[14] Appendix A.1.1

[15] See theorem 9.6.18 from [16] and proposition 12.12 from [36].

2. $(r_A + \beta + \delta r_A)/(1 + \alpha + \gamma r_A) = r_A \pmod{\ell_A^{e_A}}$, which means $\gamma r_A^2 + (\alpha + \delta) r_A - \beta = 0 \pmod{\ell_A^{e_A}}$ and hence $\gamma r_A^2 + (\alpha - \delta) r_A - \beta = 0 \pmod{\ell_A}$. Thus, a simple constraint on the coefficients ensures that the last equation has no solution then, just force $\ell_A \mid \gamma$ and $\ell_A \mid (\alpha - \delta)$, but $\ell_A \nmid \beta$.

Therefore, it is important that this equation has no solution because, otherwise, if Alice and Bob cannot control the coefficients α, β, γ, δ apart from ensuring conditions as above, Bob could be able to decrypt both messages from Alice. □

3.5.3 Summing up the conditions

In this section we will consider the three conditions on α, β, γ, and δ based on the equations obtained in sections 3.5.1 and 3.5.2 to ensure SIOT protocol security in a scenario where Alice and Bob are dishonest parties. Thus, conditions on α, β, γ, and δ are obtained that guarantee that Alice will not be able to get the secret choice of Bob's bit b and he will not be able to decipher both pairs c_0 and c_1 sent by Alice. Combining these relations yields $\gamma = -\alpha^2/\beta \pmod{\ell_A^{e_A}}$ since β is certainly invertible mod $\ell_A^{e_A}$. In particular, this means $V = -(\alpha/\beta)U$. □

Additionally, in the appendix D we will see the application of a symmetric pairing to analyze other possible conditions relative to the coefficients of the points U and V. Moreover, the appendix E shows the process of sharing of these last points.

4 Conclusion

We introduce a hybrid protocol called SIOT using a *post-quantum* protocol called SIDH, whose security is based on the difficulty of an adversary to compute isogenies between supersingular elliptic curves, and an OT protocol whose security feature is based on the privacy between a sender and receiver on a communication channel. For the security analysis of the proposed protocol, an evaluation of the correctness and privacy properties based on the DDH problem was performed, considering a supposed scenario with a dishonest sender and an honest receiver and *vice versa*. Thus, this analysis verified the guarantee of privacy between the parties involved in the communication channel. In addition, considering the same scenario mentioned above, an algebraic analysis was performed using *Weil* pairing. This analysis formulated some necessary conditions for choosing the values of the coefficients α, β, δ and γ such that both sender and receiver cannot violate the security of the proposed protocol. It should be noted that the SIOT protocol inherits the conjectures of the isogenies computational problems from SIDH protocol. Finally, it was also considered the possibility of applying symmetric pairing in the security analysis of SIOT protocol taking into account also conditions in choosing the values of the coefficients mentioned above.

References

1. A.Rostovtsev and A.Stolbunov. Public-key cryptosystem based on isogenies. Cryptology ePrint Archive, Report 2006/145., 2006.
2. Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélène Skubiszewska. Practical quantum oblivious transfer. In Joan Feigenbaum, editor, *Advances in Cryptology — CRYPTO '91: Proceedings*, pages 351–366, Berlin, Heidelberg, 1992. Springer Berlin Heidelberg.
3. D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In *Lecture Notes in Computer Science.*, volume 2139. of *Advances in Cryptology — CRYPTO 2001.* CRYPTO 2001., 2001.
4. Andrew Byrne Brian Baldwin, Richard Maloney and Gary MacGuire. A hardware analysis of twisted edwards curves for an elliptic curve cryptossistem,. Cryptology ePrint Archive, Report 2009/001, 2009.
5. Tung Chou and Claudio Orlandi. The simplest protocol for oblivious transfer. In Kristin Lauter and Francisco Rodríguez-Henríquez, editors, *Progress in Cryptology – LATINCRYPT 2015: 4th International Conference on Cryptology and Information Security in Latin America, Guadalajara, Mexico, August 23-26, 2015, Proceedings*, pages 40–58, Cham, 2015. Springer International Publishing.
6. Henri Cohen and GerhardFrey. *Handbook of Elliptic and Hyperelliptic Curve Cryptography Handbook of Elliptic and Hyperelliptic Curve Cryptography.* Chapman & Hall/CRC, 2006.
7. Longa P. Costello, C. and M. Naehrig. Efficient algorithms for supersingular isogeny diffie-hellman. Cryptology ePrint Archive, Report 2016/413, 2016.
8. C. Crepeau and J. Kilian. Achieving oblivious transfer using weakened security assumptions. In *[Proceedings 1988] 29th Annual Symposium on Foundations of Computer Science*, pages 42–52, Oct 1988.
9. Ivan Damgård, Serge Fehr, Carolin Lunemann, Louis Salvail, and Christian Schaffner. Improving the security of quantum protocols via commit-and-open. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009: 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, pages 408–427, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.
10. M.Joye T.Lange D.Bernstein, P.Birkner and C.Peters. Twisted edwards curves. In *Lecture Notes in Computer Science 5023, Springer- Verlag, New York.*, pages 389–405, 2008.
11. A.M.Childs. D.Jao and V. Soukharev. Constructing elliptic curve isogenies in quantum subexponential time., 2014.
12. William A. Stein et al. *Sage Mathematics Software - Version 7.4.* The Sage Development Team, 2016.
13. Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. *Commun. ACM*, 28(6):637–647, June 1985.
14. Luca De Feo. *Algorithmes Rapides pour les Tours de Corps Finis et les Isogénies.* PhD thesis, Ecole Polytechnique X, December 2010.
15. Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.
16. Steven D. Galbraith. *Mathematics of public key cryptography.* Cambridge University Press, Cambridge, 2012.
17. Oded Goldreich. *Foundations of Cryptography.* Cambridge University Press, Cambridge, 2004.

18. Carmit Hazay and Yehuda Lindell. *Efficient Secure Two - Party Protocols - Techniques and Constructions.* Springer Berlin Heidelberg, 2010.
19. Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. *An introduction to mathematical cryptography.* Undergraduate Texts in Mathematics. Springer, New York, second edition, 2014.
20. A. Joux. A one round protocol for tripartite diffe-hellman, in algorithmic number theory. In *Lecture Notes in Computer Science, Springer, Berlin.*, volume 1838., pages 385–393., 2000.
21. Raza Ali Kazmi. Cryptography from post-quantum assumptions. Cryptology ePrint Archive, Report 2015/376, 2015. `http://eprint.iacr.org/2015/376`.
22. H.Kurumatani K.Okeya and K.Sakurai. Elliptic curves with the montgomery-form and their cryptography applications. In *Proceedings of the third internation workshop on practice and theory in Public Key Cryptography, Lectures Notes in Computer Science 1751, Springer-Verlag, London.*, pages 238–257., 2000.
23. D. Mayers and L. Salvail. Quantum oblivious transfer is secure against all individual measurements. In *Physics and Computation, 1994. PhysComp '94, Proceedings., Workshop on*, pages 69–77, Nov 1994.
24. Dominic Mayers. Quantum key distribution and string oblivious transfer in noisy channels. In Neal Koblitz, editor, *Advances in Cryptology — CRYPTO '96: 16th Annual International Cryptology Conference Santa Barbara, California, USA August 18–22, 1996 Proceedings*, pages 343–357, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg.
25. Barreto P.S.L.M. and Naehrig M. Pairing-friendly elliptic curves of prime order. In *Selected Areas in Cryptography. SAC 2005. Lecture Notes in Computer Science*, volume 3897, 2005.
26. Michael O. Rabin. How to exchange secrets with oblivious transfer, 2005. Harvard University Technical Report 81 talr@watson.ibm.com 12955 received 21 Jun 2005.
27. Amir Jalali Mehran Mozaffari Kermani Reza Azarderakhsh, Brian Koziel and David Jao. Neon-sidh: Efficient implementation of supersingular isogeny diehellman key exchange protocol on arm. Cryptology ePrint Archive, Report 2016/669., 2016.
28. Phillip Rogaway. On the role of definitions in and beyond cryptography.
29. Andrew Shallue and Christiaan E. van de Woestijne. Construction of rational points on elliptic curves over finite fields. In Florian Hess, Sebastian Pauli, and Michael Pohst, editors, *Algorithmic Number Theory: 7th International Symposium, ANTS-VII, Berlin, Germany, July 23-28, 2006. Proceedings*, pages 510–524, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
30. Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics.* Springer, Dordrecht, second edition, 2009.
31. Anton Stolbunov. *Cryptography Shemes based on Isogenies.* PhD thesis, Norwegian University of Science and Technology., 2012.
32. John Tate. Endomorphisms of abelian varieties over finite fields. *Invent. Math.*, 2:134–144, 1966.
33. Dominique Unruh. Universally composable quantum multi-party computation. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30 – June 3, 2010. Proceedings*, pages 486–505, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
34. Jacques Vélu. Isogénies entre courbes elliptiques. *C. R. Acad. Sci. Paris Sér. A-B*, 273:A238–A241, 1971.

35. David Wagner. Midterm solution, 2006. `http://www.cs.berkeley.edu/~daw/teaching/cs276-s06/mtsol.ps`.
36. Lawrence C. Washington. *Elliptic curves - Number Theory and Cryptography.* Taylor & Francis Group. LLC, second edition. edition, 2008.
37. S.Galbraith W.Castryck and R.Rezacian Farashahi. Efficient arithmetic on elliptic curves using a mixed edwards-montgomery representation,. Cryptology ePrint Archive, Report 2008/218, 2008.
38. Andrew Chi-Chih Yao. Security of quantum protocols against coherent measurements. In *Proceedings of the Twenty-seventh Annual ACM Symposium on Theory of Computing*, STOC '95, pages 67–75, New York, NY, USA, 1995. ACM.

A The Basic Theory

Throughout this appendix, unless otherwise noted, we are going to use the following notation:

- $\mathbb{F}_p \rightarrow$ A finite Field, where $p \geqq 3$ be a prime;
- $\bar{\mathbb{F}}_p \rightarrow$ A fixed algebraic closure of $\mathbb{F}_p$;
- $\mathbb{F}_{p^k} \rightarrow$ An extension of order k of finite field $\mathbb{F}_p$;
- E, $\tilde{E} \rightarrow$ Two fixed elliptic curves over $\mathbb{F}_p$;
- $E(\mathbb{F}_p)$, $E(\bar{\mathbb{F}}_p) \rightarrow$ The set of pairs (x, y) satisfying the Weierstrass equation of E where x and y are taken in $\mathbb{F}_p$ or $\bar{\mathbb{F}}_p$, respectively;
- $\#E \rightarrow$ Cardinality of the E group, *i.e*, the number of elements in an elliptic group ;
- ϕ or $\psi \rightarrow$ An algebraic map between E and $\tilde{E}$ and ;
- $\mathcal{O}_E \rightarrow$ A point at infinity on a curve E.

A.1 Elliptic Curves

Let $p \geqq 3$ be a prime. An elliptic curve E over $\mathbb{F}_p$, *i.e*, $E(\mathbb{F}_p)$, is an equation of the form $E(\mathbb{F}_p) : y^2 = x^3 + Ax + B$ with $A, B \in \mathbb{F}_p$ satisfying $\triangle = 4A^3 + 27B^2 \neq 0$. Equations of this type are called *Weierstrass equations*. The set of points on E with coordinates in $\mathbb{F}_p$ is the set $E(\mathbb{F}_p) = \{(x, y) : x, y \in \mathbb{F}_p \text{ satisfy } y^2 = x^3 + Ax + B\} \cup \{\mathcal{O}\}$[16]. The points of an elliptic curve E has addition properties[17] and hence, form an Abelian Group. The quantity $\triangle$ is called discriminant and there is a quantity j called $j - invariant$ of a *Weierstrass equations* defined by $j = -1728(4A)^3/\triangle$.

There are other types of elliptic curves such as *Montgomery Form* ($By^2 = x^3 + Ax^2 + x$) and *Edwards Form* ($\bar{x}^2 + \bar{y}^2 = 1 + d\bar{x}^2\bar{y}^2$). These special forms are well suited for certain computations and many authors have used them to improve the efficiency of diverse cryptographic applications[18]. Despite the recognized importance of these special curves, this work is focused on *Weierstrass equation* for the introduction of the concept of SIOT protocol. Lastly, for more details about all types of elliptic curves applied in cryptograph, see [16].

[16] The point $\mathcal{O}$ is a special point located "at infinity". It acts like zero for elliptic curve addition on finite field prime.
[17] See theorem 6.5 from [19].
[18] See, for instance, [4,7,10,15,22,37]

A.1.1 Points of finite order

The concept of points of finite order is important for a later and better understanding about the concept of *kernels* in the isogeny of elliptic curves. Thus, according to [19] let $\ell \geq 1$ be an integer and E be an elliptic curve over $\mathbb{F}_p$. A point $P \in E(\mathbb{F}_p)$ satisfying the identity[19] $\ell P = \mathcal{O}$ is called a point of order ℓ or ℓ-torsion point in the group $E(\mathbb{F}_p)$. The set of ℓ-torsion points is denoted by $E[\ell] = \{P \in E : [\ell]P = \mathcal{O}\}$ and $E[\ell]$ is a subgroup of $E(\mathbb{F}_p)$. For ℓ such that $p \nmid \ell$ then $E[\ell] \cong \mathbb{Z}/\ell\mathbb{Z} \times \mathbb{Z}/\ell\mathbb{Z}$. If ℓ is prime then, $E[\ell]$ can be view as a 2-dimensional vector space over the field $\mathbb{Z}/\ell\mathbb{Z}$,*i.e.*, given points $P, Q, R \in E[\ell]$ then, we can write a linear combination $P = \alpha Q + \beta R$ for unique $\alpha, \beta \in \mathbb{Z}/\ell\mathbb{Z}$.

A.1.2 Supersingular elliptic curve

As our proposed protocol is based on [15], now we are going to see some concepts to supersingular elliptic curves. According to [36] an elliptic curve $E(\mathbb{F}_p)$ is called supersingular if $E[p] = \mathcal{O}$. In other words, there are no points of order p, even with coordinates in an algebraically closed field. Thus, let $p \geqslant 3$ be prime then, the elliptic curve $E : y^2 = x^3 + x$ over $\mathbb{F}_p$ is supersingular if and only if $p = 3 \bmod 4$. Moreover, the theorem 9.11.2 from [16] lists other properties that satisfy a supersingular elliptic curve over $\mathbb{F}_p$. On the other hand, if a given elliptic curve does not satisfy any of the conditions in this theorem then this elliptic curve is called ordinary.

Supersingular curves can be defined over $\mathbb{F}_p$ or $\mathbb{F}_{p^2}$. The field with p^2 elements resembles $\mathbb{F}_{p^2} = \{a + bi : a, b \in \mathbb{F}_p\}$, where i satisfies $i^2 = -1$. For more details, see [19,30].

A.1.3 The Weil pairing

The Weil pairing has notorious applications to cryptography, for instance [3,20,25]. Now we recall some concepts and properties of this type of pairing that will be important to understand later its use in the security analysis of the protocol proposed in this work.

Definition A.1. *The Weil pairing e_ℓ takes as input a pair of points $P, Q \in E[\ell]$ and gives as output an ℓ^{th} root of unity $e_\ell(P, Q)$. In addition, e_ℓ has the followings properties:*

i. *The values of the Weil pairing satisfy $e_\ell(P, Q)^\ell = 1$ for all $P, Q \in E[\ell]$;*
ii. *The Weil pairing is bilinear then $e_\ell(P, Q_A + Q_B) = e_\ell(P, Q_A)e_\ell(P, Q_B)$ for all $P, Q_A, Q_B \in E[\ell]$ and $e_\ell(P_A + P_B, Q) = e_\ell(P_A, Q)e_\ell(P_B, Q)$ for all $P_A, P_B, Q \in E[\ell]$;*
iii. *The Weil pairing is alternating, which means that $e_\ell(P, P) = 1$ for all $P \in E[\ell]$. This implies that $e_\ell(P, Q) = e_\ell(Q, P)^{-1}$ for all $P, Q \in E[\ell]$. In addition, if $e_\ell(P, Q) = e_\ell(Q, P)$ then, the pairing is symmetric;*
iv. *The Weil pairing is nondegenerate, which means that if $e_\ell(P, Q) = 1$ for all $Q \in E[\ell]$, then $P = \mathcal{O}$.*

[19] ℓP means $\underbrace{P + P + P + \cdots + P}_{\ell \text{ times}}$.

A.1.4 Distortion map and modified Weil pairing

Definition A.1. *Let E be an elliptic curve, $\ell \geqq 3$ be a prime, $P \in E[\ell]$ be a point of order ℓ and $\psi : E \to E$ be a map from E to itself. Thus, ψ is an $\ell-$ distortion map for P if it has the following properties:*

i. $\psi(nP) = n\psi(P)\ \forall\ n \geqq 1$;
ii. $e_\ell(P, \psi(P))^r = 1$ *if and only if* $\ell | r$ *(i.e, r is a multiple of ℓ.).*

Proposition A.1.4.1. *Let E be the elliptic curve $E\colon y^2 = x^3 + x$ over $\mathbb{F}_{p^2}$ and suppose that $\mathbb{F}_{p^2}$ has an element $\alpha \in \mathbb{F}_{p^2}$ satisfying $\alpha^2 = -1$. Define a map ψ by $\psi(x, y) = (-x, \alpha y)$ and $\psi(\mathcal{O}) = \mathcal{O}$.*

i. *Let $P \in E(\mathbb{F}_{p^2})$. Then $\psi(P) \in E(\mathbb{F}_{p^2})$, so ψ is a map from $E(\mathbb{F}_{p^2})$ to itself;*
ii. *The map ψ respects the addition law on E , i.e, $\psi(P_1 + P_2) = \psi(P_1) + \psi(P_2)$ $\forall\ P_1,\ P_2 \in E(\mathbb{F}_{p^2})$.*

Proof. See [19] and Section 13.1.6 from [6]. □

Definition A.2. *Let $\ell \geqq 3$ be a prime, Q and Q' be multiples of $P \in E[\ell]$. Thus, the modified Weil pairing $\hat{e}_\ell$ on $E[\ell]$ is defined by $\hat{e}_\ell(Q, Q') = e_\ell(Q, \psi(Q'))$.*

Proposition A.1.4.2. *Let E be an elliptic curve, let $P \in E[\ell]$, let ψ be an ℓ-distortion map for P, and let $\hat{e}_\ell$ be the modified Weil pairing relative to ψ. Let Q and $Q^{'}$ be multiples of P. Then, $\hat{e}_\ell(Q, Q') = 1$ if and only if $Q = \mathcal{O}$ or $Q' = \mathcal{O}$.*

Proof. See [19]. □

A.1.5 Isogenies

In short, isogeny-based cryptography utilizes unique algebraic maps between elliptic curves that satisfy group homomorphism. This original idea introduced by [1][20] detailed a Diffie-Hellman cryptosystem based on the hardness of computing isogenies between ordinary elliptic curves. Nevertheless, [11] developed a quantum algorithm that could compute isogenies between ordinary curves in subexponential time. This algorithm uses the fact that the structure of the elliptical group is commutative. Thus, [15] adapted the isogeny-based key exchange protocol to be based on the difficulty of computing isogenies between supersingular elliptic curves, which does not have commutative endomorphism ring.

Definition A.1. *Let E_1 and E_2 be elliptic curves over $\mathbb{F}_p$. An isogeny over $\mathbb{F}_p$ is a morphism $\phi\colon E_1 \to E_2$ over $\mathbb{F}_p$ such that $\phi(\mathcal{O}_{E_1}) = \mathcal{O}_{E_2}$ is a group homomorphism. The zero isogeny is the constant map $\phi\colon E_1 \to E_2$ given by $\phi(P) = \mathcal{O}_{E_2}$ for all $P \in E(\overline{\mathbb{F}}_p)$. If there is an isogeny between two elliptic curves E_1 and E_2 then:*

i. *E_1 and E_2 are isogenous;*

[20] See [31] to better understand the history of isogeny-based cryptographic.

ii. $\#E_1(\mathbb{F}_p) = \#E_2(\mathbb{F}_p)$ *[32];*
iii. E_1 *and* E_2 *have the same* $\jmath-$*invariant if and only if* $E_1 \backsimeq E_2$ *over* $\bar{\mathbb{F}}_p$ *(i.e. exists an isomorfism from* E_1 *to* E_2*)*[21].

Definition A.2. *Let* E_1 *and* E_2 *be elliptic curves over* $\mathbb{F}_p$ *and* $\phi\colon E_1 \to E_2$ *over* $\mathbb{F}_p$*. The degree of a non-zero isogeny is the degree of the morphism. The degree of the zero isogeny is* 0*. If there is an isogeny of degree* ℓ *between elliptic curves* E_1 *and* E_2 *then they are* ℓ*-isogenous.*

Definition A.3. *Let* E_1 *and* E_2 *be elliptic curves over* $\mathbb{F}_p$ *and* $\phi\colon E_1 \to E_2$ *an isogeny. Then the kernel of an isogeny is* $ker(\phi) = \{P \in E_1(\bar{\mathbb{F}_p}) : \phi(P) = \mathcal{O}_{E_2}\}$.

Definition A.4. *A non-zero isogeny separable* $\phi\colon E_1 \to E_2$ *over* $\mathbb{F}_p$ *of* ℓ *- degree has* $\#ker(\phi) = \ell$*, i.e, the number of kernel elements equals to* ℓ*. [16].*

Definition A.5. *Let* $\phi\colon E_1 \to E_2$ *and* $\hat{\phi}\colon E_2 \to E_3$ *be two isogenies with* ℓ*-degree and* $\hat{\ell}$*-degree, respectively. Then, their composition is an isogeny* $\hat{\phi}(\phi)\colon E_1 \to E_3$ *with* $(\ell \cdot \hat{\ell})$*-degree.*

Proposition A.1.5.1. *Let* E_1 *be an elliptic curve over* $\mathbb{F}_p$ *and* $\mathbb{G}$ *a finite subgroup of* $E_1(\bar{\mathbb{F}}_p)$ *that is defined over* $\mathbb{F}_p$*. Then, there is an unique elliptic curve* $E_{\mathbb{G}}$ *and a separable isogeny* $\phi\colon E_1 \to E_{\mathbb{G}} = E_1/\mathbb{G}$ *such that* $Ker(\phi) = \mathbb{G}$.

Proof. See Theorem 25.1.6 and Corollary 25.1.7 from [16]. □

The Vélu's formula [34] can be used for computing a separable isogeny from an elliptic curve E_1 with given kernel $\mathbb{G}$. Velú showed how to explicitly find the rational function form of a normalized $\phi\colon E_1 \to E_{\mathbb{G}}$. The Vélu's formula is presented in next proposition.

Proposition A.1.5.2. *Let* $E(\mathbb{F}_p) : y^2 = x^3 + ax + b$ *be an elliptic curve defined over* $\mathbb{F}_p$*, and let* $\mathbb{G} \subset E(\bar{\mathbb{F}}_p)$ *be a finite subgroup. The separate isogeny* $\phi\colon E_1 \to E_{\mathbb{G}}$*, of kernel* $\mathbb{G}$*, can be written as*

$$\phi(P) = \left(x(P) + \sum_{Q \in \mathbb{G}/\{\mathcal{O}\}} x(P+Q) - x(Q), y(P) + \sum_{Q \in \mathbb{G}/\{\mathcal{O}\}} y(P+Q) - y(Q) \right)$$

and the curve $E_{\mathbb{G}}$ has equation $y^2 = x^3 + a^{'}x + b^{'}$, where

$$a^{'} = a - 5 \sum_{Q \in \mathbb{G}/\{\mathcal{O}\}} (3x(Q)^2 + a),$$

$$b^{'} = b - 7 \sum_{Q \in \mathbb{G}/\{\mathcal{O}\}} (5x(Q)^3 + 3ax(Q) + b)$$

Proof. See [14]. □

[21] Theorem 9.3.6 from [16].

Supersingular curves have the property that for every prime $\ell \neq p$, there exist $\ell + 1$ isogenies of degree ℓ originating from a given supersingular curve and a common field that includes all isogenous curves in $\mathbb{F}_{p^2}$ [27]. However, it is believed to be hard to determine the isogeny degree. Thus, the security of the cryptosystem from supersingular elliptic curve isogenies is based on this assumption. From now on, in this work, we will consider all the elliptic curves as supersingular over $\mathbb{F}_{p^2}$.

A.2 Oblivious Transfer protocol

Oblivious Transfer (OT) is a protocol in which a sender transfers one of many pieces of information to a receiver, but remains oblivious as to what piece has been transferred. The original notion of OT was first proposed by Rabin in 1981 [26] in which a sender sends an encrypted message to a receiver and this one could decrypt such message with probability 1/2. After this, [13] presented a general form of OT, named 1-out-of-2 OT, $\binom{2}{1}$ - OT for short, *i.e*, where a sender sends two encrypted messages to a receiver being able to decrypt only one of them. Moreover, many authors have generalized this to $\binom{n}{1}$ - OT where the receiver chooses one message out of n and $\binom{k}{n}$-OT in which the receiver chooses a subset of size k from among n messages. In this work, we will be focused only on $\binom{2}{1}$ - OT. Therefore, for a merely conceptual and basic view, the OT protocol [5] is presented on appendix C. As can be seen, a sender has two input messages[22] M_0, $M_1 \in \mathcal{M}$ and a receiver has a choice bit b. At the end of the protocol the receiver is supposed to learn the message M_b and nothing else, while the sender is supposed to learn nothing, after he sends the messages M_0 and M_1. The fig 3 describes the pseudocode of the exchange of information[23] between two parts of the OT protocol proposed in [5].

B Linearly independent points and $\binom{2}{1}$ - SIOT protocol implementation

In this appendix B, we present definitions for the understanding of the process that determines the choice of linearly independent points P_A, Q_A, P_B and Q_B in the proposed protocol.

Definition B.1 (Frobenius). *Let $E(\mathbb{F}_q)$ be an ellipic curve, and let $E(\mathbb{F}_{q^k})$ be its $\mathbb{F}_{q^k}$-rational extension. The* Frobenius *map is the function $\Phi : E(\mathbb{F}_{q^k}) \to E(\mathbb{F}_{q^k})$ defined by $\Phi(x, y) = (x^q, y^q)$ for any $(x, y) \in E(\mathbb{F}_{q^k})$. Φ^i denotes its i-th self-composition, i.e. for any $P \in E(\mathbb{F}_{q^k})$, $\Phi^i(P) := P$ for $i = 0$, and $\Phi^i(P) = \Phi(\Phi^{i-1}(P))$ for $i > 0$.*

[22] Let $\mathcal{M}$ be the set of all plaintexts with binary strings of fixed length.
[23] This exchange of information involves the sharing of one K key between the parts.

```
for b, i ∈ {0, 1} do
  if b = i then
    k_b = k_i
    return "Shared key"
  elseif b ≠ i then
    k_b ≠ k_i
    return "No shared key"
  endfor
```

Figure 3. Pseudocode of information exchange between two parts from $\binom{2}{1}$- OT protocol.

Definition B.2 (Trace). *Let $E(\mathbb{F}_q)$ be an ellipic curve, and let $E(\mathbb{F}_{q^k})$ be its $\mathbb{F}_{q^k}$-rational extension. The* trace *map is the function $tr : E(\mathbb{F}_{q^k}) \to E(\mathbb{F}_{q^k})$ defined by $tr(P) = (1/k)\sum_{i=0}^{k-1} \Phi^i(P)$ where $1/k$ denotes the inverse of k mod the order of $E(\mathbb{F}_{q^k})$. In particular, $k = 2$ for a supersingular curve in characteristic $p > 3$, and $tr(P) = (1/2)(P + \Phi(P))$.*

Hence, the trace map is important in that its eigenspaces, if nontrivial, form two linearly independent groups that can be used to sample points P_A, Q_A, P_B, Q_B efficiently. Moreover, the trace definition assumes that $\gcd(k, \#E(\mathbb{F}_{q^k})) = 1$, which may not be the case, especially in the important setting where $\ell_A = 2$. Thus, for this scenario we also define the $quasi - trace$ map:

Definition B.3 (Quasi-trace). *Let $E(\mathbb{F}_q)$ be an ellipic curve, and let $E(\mathbb{F}_{q^k})$ be its $\mathbb{F}_{q^k}$-rational extension. The* quasi-trace *map is the function $tr : E(\mathbb{F}_{q^k}) \to E(\mathbb{F}_{q^k})$ defined by $tr(P) = \sum_{i=0}^{k-1} \Phi^i(P)$. In particular, $k = 2$ for a supersingular curve in characteristic $p > 3$, and $tr(P) = P + \Phi(P)$.*

C Protocol random $\binom{2}{1}$- OT

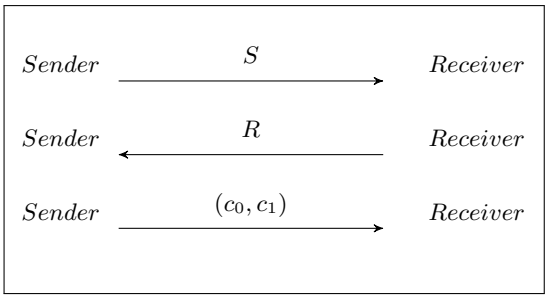
In this appendix, we see the simplified scheme of the random OT proposed in [5].

C.1 Premises

1. The scheme from [5] works in a primitive additive group $(\mathbb{G}, B, \mathbb{F}_p, +)$ of prime order p, generated by base point B;
2. The hash function $\mathcal{H} : (\mathbb{G} \times \mathbb{G}) \times \mathbb{G} \to \{0,1\}^s$ is used to generate a cryptographic key k_i for use in a symmetric cipher defined by the functions $\mathcal{E}$ (encryption) and $\mathcal{D}$ (decryption);

   Remark C.1.0.1. Let s be a safety parameter.

3. Abstract view of information exchange from protocol $\binom{2}{1}$- OT.



Remark C.1.0.2. Let $(c_0, c_1) = (\mathcal{E}(k_0, M_0), \mathcal{E}(k_1, M_1))$.

C.2 Setup: Sender and Receiver

C.2.1 Setup - Sender

1. Sender secretly chooses a value $y \in \mathbb{F}_p$;
2. Sender computes:

$$S = yB \qquad (1)$$
$$T = yS; \qquad (2)$$

3. Sender sends S to Receiver which refuses if $S \notin \mathbb{G}$.

C.2.2 Setup - Receiver

1. Receiver secretly chooses a value $x \in \mathbb{F}_p$;
2. Receiver computes:

$$R = b.S + x.B \qquad (3), \ \textit{where}\, b \in \{0,1\}\, \textit{is chosen by Receiver};$$

3. Receiver sends R to Sender which refuses if $R \notin \mathbb{G}$.

C.3 Generation of cryptographic keys k_j, $j \in \{0,1\}$.

1. Sender computes $k_j = \mathcal{H}_{(S,R)}(yR - jT)$; (4)
2. Receiver computes $k_b = \mathcal{H}_{(S,R)}(bS + xB)$. (5)

C.4 Encryption and Decryption

1. Sender encrypts and sends $c = (c_0, c_1)$ to Receiver. Recalling $c_0 = \mathcal{E}(k_0, M_0)$ and $c_1 = \mathcal{E}(k_1, M_1)$;
2. Receiver decrypts and gets $M_b = \mathcal{D}(k_b, c_j)$, $j \in \{0, 1\}$.

Remark C.4.0.1. It is verified that a key $k_j, j \in \{0, 1\}$, is computed by $\mathcal{H}_{(S,R)}[xyB + (b - j)T]$. Hence, at the end of the protocol if both parts are honest then we have that $k_b = k_j$. In other words, if $j = 0$ then $c = c_0 = 0$ and $k_0 = k_b = \mathcal{H}_{(S,R)}(xyB)$. Otherwise, if $j = 1$ then $c = c_1 = 1$ and $k_1 = k_b = \mathcal{H}_{(S,R)}(xyB)$.

$$
\begin{aligned}
k_j &= yR - jT; \\
&= y(bS + xB) - jT; && from\, equation\ (3) \\
&= byS + xyB - jT; \\
&= bT + xyB - jT; && from\, equations\ (1)\, and\, (2) \\
&= xyB + (b - j)T.
\end{aligned}
$$

Therefore, we can conclude that if the Receiver chooses $b \notin j$, it will not share the secret (cryptographic key) with the Sender.

D A possibility of symmetric pairings in the SIOT

Under certain circumstances, it is possible to define a *symmetric* pairing $\hat{e} : E_B[\ell_A^{e_A}] \to F_{p^2}$. We now analyze the condition under which this can happen. In what follows, recall that a distortion map is a linear transform that maps a curve point to a linearly independent point.

The embedding degree for $E_B[\ell_A^{e_A}]$ is only 1, not 2 as it is for E_0, because E_B is defined over F_q with $q := p^2$, and since $p = (\ell_A^{e_A} \ell_B^{e_B} f)^2 - 1$, it follows that $\#E_B[\ell_A^{e_A}] = (\ell_A^{e_A})^2 \mid q - 1$. Hence a distortion map ψ must map a point $P \in E[\ell_A^{e_A}](F_q)$ to a point $Q \in E[\ell_A^{e_A}](F_q)$ that is linearly independent from P, in which case ψ linearly maps a basis (G_B, H_B) to another basis (G'_B, H'_B). In particular, all coefficients of ψ in basis (G_B, H_B) must be integers mod $\ell_A^{e_A}$. For such a map to be a distortion map, it must have no eigenvectors (otherwise it would fail to map those points to linearly independent points), so we can simply require the characteristic polynomial to have no roots mod $\ell_A^{e_A}$.

In that case, the map $\psi(uG_B + vH_B) := vG_B - uH_B$ could be a suitable distortion map. Its characteristic polynomial is $\lambda^2 + 1$ which has no roots mod $\ell_A^{e_A}$ for a careful choice of ℓ_A (e.g. $\ell_A = 3$). Now define the modified pairing

$\hat{e}(P,Q) := e(P,\psi(Q))$ where $e(\cdot)$ is the Weil pairing. Then:

$$\begin{aligned}
\hat{e}(aG_B + bH_B, cG_B + dH_B) &= e(aG_B + bH_B, dG_B - cH_B) \\
&= e(aG_B, -cH_B) \cdot e(bH_B, dG_B) \\
&= e(G_B, H_B)^{-ac-bd}, \\
\hat{e}(cG_B + dH_B, aG_B + bH_B) &= e(cG_B + dH_B, bG_B - aH_B) \\
&= e(cG_B, -aH_B) \cdot e(dH_B, bG_B) \\
&= e(G_B, H_B)^{-ac-bd},
\end{aligned}$$

so this modified pairing is symmetric.

It remains to determine if it is isogeny-equivariant. If it is, a further constraint exists for the coefficients of U and V, namely:

$$\begin{aligned}
\hat{e}(G_B + \lambda U, H_B + \lambda V) &= \hat{e}(G_B, H_B)^{(1+\lambda\alpha)(1+\lambda\delta)} \\
&\cdot\ \hat{e}(H_B, G_B)^{\lambda\beta\lambda\gamma} \\
&= \hat{e}(G_B, H_B)^{(1+\lambda\alpha)(1+\lambda\delta)+\lambda^2\beta\gamma} \\
&= e(G_B, H_B),
\end{aligned}$$

so we also need $(1+\lambda\alpha)(1+\lambda\delta) + \lambda^2\beta\gamma = 1 \pmod{\ell_A^{e_A}}$. □

D.1 Taking symmetric pairings into account

Coupling the above constraints $\gamma = -\alpha^2/\beta \pmod{\ell_A^{e_A}}$ and $\delta = -\alpha \pmod{\ell_A^{e_A}}$ with the additional condition $(1+\lambda\alpha)(1+\lambda\delta) + \lambda^2\beta\gamma = 1 \pmod{\ell_A^{e_A}}$, we have $(1+\lambda\alpha)(1-\lambda\alpha) - \lambda^2\beta\alpha^2/\beta = 1 - 2\lambda^2\alpha^2 = 1 \pmod{\ell_A^{e_A}}$ for any λ, or simply $2\alpha^2 = 0 \pmod{\ell_A^{e_A}}$, which has the solution $\alpha = \alpha_0 \cdot 2^{\lfloor e_A/2 \rfloor}$ for $\ell_A = 2$ and any $0 \leqslant \alpha_0 < 2^{\lceil e_A/2 \rceil}$, or $\alpha = \alpha_0 \cdot \ell_A^{\lceil e_A/2 \rceil}$ for $\ell_A \neq 2$ and any $0 \leqslant \alpha_0 < \ell_A^{\lfloor e_A/2 \rfloor}$. □

E Validating the process of sharing points (U, V)

Now, we are going to verify the sharing of points U and V between Bob and Alice.. This is important from the point of view of the correct functionality of the SIOT protocol with regard to the oblivious characteristic, *i.e*, in practical terms, the U and V points provide the sender to generate two secret keys. Thus, Bob will hash a uniformly random bit string w to compute the coefficients α and β, as suggested in Section 3.4. He defines U and V points by means of the algebraic relation $U = \alpha G_B + \beta H_B$ and $V = -(\alpha/\beta)U$. After that, he sends to Alice one of the pairs (G_B, H_B) or $(G_B - U, H_B - V)$. Obviously, Alice doesn't distinguish[24] which pair of points she received. Therefore, Alice can map the shared w string applying the same Bob's process. Thus, upon receipt of $\hat{G}_B$ and

[24] Recalling Subsection 3.4, Lemma 1.

$\hat{H}_B$ points from Bob's public key, say $\hat{pk}_B = (E_B, \hat{G}_B, \hat{H}_B)$, and knowing the algebraic relation above, Alice defines $\hat{U} = \alpha\hat{G}_B + \beta\hat{H}_B$, $\hat{V} = -(\alpha/\beta)\hat{U}$ yielding $\hat{U} = U$ and $\hat{V} = V$. In other words, Alice and Bob have the assurance that U and V points are being correctly shared between the parties.

Proof.

1. In a first assumption, Alice receives $\hat{G}_B = (G_B - U)$ and $\hat{H}_B = (H_B - V)$ points from Bob. Evidently, she has not any knowledge about G_B and H_B points. Thus, she performs the algebraic development below.

$$
\begin{aligned}
&\hat{U} = \alpha \cdot \hat{G}_B + \beta \cdot \hat{H}_B;\\
&\hat{U} = \alpha \cdot (G_B - U) + \beta \cdot (H_B - V);\\
&\hat{U} = \alpha \cdot G_B - \alpha \cdot U + \beta \cdot H_B - \beta \cdot V;\\
&\hat{U} = \underbrace{\alpha \cdot G_B + \beta \cdot H_B}_{U} - (\alpha \cdot U + \beta \cdot V);\\
&\hat{U} = U - (\alpha \cdot U + \beta \cdot V);\\
&\hat{U} = U - \underbrace{[\alpha \cdot U + \beta \cdot (-\frac{\alpha}{\beta} \cdot U)]}_{0};\\
&\hat{U} = U.
\end{aligned}
$$

If $\hat{U} = U$ and $V = -(\alpha/\beta)U$, then $\hat{V} = V$. □

2. In this second assumption, Alice receives $\hat{G}_B = G_B$ e $\hat{H}_B = H_B$ points from Bob. Similarly,

$$
\begin{aligned}
&\hat{U} = \alpha \cdot \hat{G}_B + \beta \cdot \hat{H}_B;\\
&\hat{U} = \alpha \cdot G_B + \beta \cdot H_B;\\
&\hat{U} = \underbrace{\alpha \cdot G_B + \beta \cdot H_B}_{U};\\
&\hat{U} = U.
\end{aligned}
$$

Evidently, in this case, If $\hat{U} = U$ then, $\hat{V} = V$ □

.