Supersingular Isogeny Oblivious Transfer

Paulo Barreto[1], **Gláucio Oliveira**[2], **Waldyr Benits**[3], **Anderson Nascimento**[1]

[1]Escola de Engenharia & Tecnologia – Universidade de Washington, Tacoma, EUA.

[2]Instituto de Matemática e Estatística – Universidade de São Paulo (IME-USP)

[3]Centro de Análises de Sistemas Navais – Marinha do Brasil (CASNAV-MB)

{pbarreto,andcla}@uw.edu, glaucioaorj@gmail.com, wbenits@yahoo.com.br

Abstract. *In this paper we present an Oblivious Transfer (OT) protocol that combines an OT scheme together with the supersingular isogeny Diffie-Hellman (SIDH) primitive. Our proposal 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* $\binom{2}{1}$*-SIOT and analyze the protocol to verify its security.*

Resumo. *Neste artigo, apresentamos um protocolo Oblivious Transfer (OT) que combina um esquema OT juntamente com a primitiva do protocolo Supersingular Isogeny Diffie-Hellman (SIDH). Nossa proposta é um candidato para segurança pós-quântica OT e demonstra que o SIDH naturalmente suporta a funcionalidade OT. Consideramos o protocolo na configuração mais simples de* $\binom{2}{1}$*-SIOT e analisamos a sua segurança.*

1. Introdução

[Rabin 1981] propôs a primeira noção de *Oblivious Transfer* (OT). Desde então, muitos protocolos criptográficos foram implementados utilizando uma estrutura OT. [Chou and Orlandi 2015] afirmam que eficientes protocolos OT são vulneráveis em um cenário quântico, em que a hipótese de segurança se baseia na dificuldade de resolver o Problema do Logaritmo Discreto ou de fatorar números inteiros. Vários artigos tem introduzido o sistema OT no contexto da criptografia quântica tais como [Crepeau and Kilian 1988] e [Bennett et al. 1992] entre outros, onde os usuários legítimos manipulam estados quânticos. Por outro lado, a pesquisa em construções OT pós-quânticas tem crescido gradualmente. Desta forma, pode-se citar, por exemplo, os trabalhos de [Kazmi 2015], e [Vitse 2019]. Em geral, em um protocolo $\binom{2}{1}$-OT, o remetente envia duas mensagens distintas, m_a e m_b para um destinatário, que escolhe somente uma delas para ter acesso ao conteúdo. No final do protocolo, o remetente não tem o conhecimento de qual mensagem foi escolhida e, também, o destinatário não tem acesso ao conteúdo da outra mensagem.

Nossa contribuição. De acordo com [Hazay and Lindell. 2010], OT é uma das estruturas mais importantes em criptografia e vantajosa para a construção de protocolos seguros. Em termos de aplicação, esses tipos de protocolos podem ser utilizados em processos de leilões eletrônicos, ou ainda em assinaturas de contratos [Even et al. 1985] ou esquemas de transações com dinheiro eletrônico [Barak 2007]. Neste trabalho, o nosso objetivo principal foi a implementação do $\binom{2}{1}$-SIOT usando as primitivas do SIDH de [Feo et al. 2014]

com o propósito de proporcionar privacidade entre remetente e destinatário e, ao mesmo tempo, prover uma resistência contra o eminente advento da computação quântica.

Este artigo está organizado da seguinte forma: A seção 2 descreve o protocolo $\binom{2}{1}$-SIOT bem como apresenta uma análise de segurança. Na seção 3, apresentamos a conclusão da análise de segurança do protocolo proposto. A seção 4 mostra alguns resultados e parâmetros do $\binom{2}{1}$-SIOT. Por fim, na seção 5, tem-se a conclusão.

2. Protocolo $\binom{2}{1}$-SIOT

2.1. Notações

Nós utilizamos as primitivas criptográficas do protocolo de troca de chaves SIDH de [Feo et al. 2014]. Assim, as seguintes notações serão usadas:

i. $\mathcal{M}, \mathcal{K}, \mathcal{C} \rightarrow$ Conjunto de todos os textos claros, chaves e cifras em uma cadeia binária de bits com comprimento fixado, respectivamente;
ii. $p \rightarrow$ um número primo tal que $p = 3(\mathrm{mod}4)$;
iii. $\mathbb{F}_{p^2} \rightarrow$ Uma extensão quadrática de um corpo $\mathbb{F}_p$, onde $\mathbb{F}_{p^2} = \mathbb{F}_p[i]/\langle i^2 + 1\rangle$;
iv. $E_0[\mathbb{F}_{p^2}] \rightarrow$ Uma curva elíptica supersingular sobre $\mathbb{F}_{p^2}$;
v. $\mathbb{Z}/\ell\mathbb{Z} \rightarrow$ Um corpo de inteiros módulo ℓ, onde ℓ é um número primo tal que $\ell \nmid p$;
vi. $P_A, Q_A \rightarrow$ Pontos linearmente independentes sobre uma curva elíptica supersingular $E_0[\ell_A^{e_A}]$;
vii. $P_B, Q_B \rightarrow$ Pontos linearmente independentes sobre uma curva elíptica supersingular $E_0[\ell_B^{e_B}]$;
viii. $\phi_A, \phi_B \rightarrow$ Isogenias entre as curvas elípticas supersingulares E_0 e E_A, E_0 e E_B, respectivamente;
ix. $\phi_A^{'}, \phi_B^{'} \rightarrow$ Isogenias entre as curvas elípticas supersingulares E_B e E_{BA}, E_A e E_{AB}, respectivamente;
x. $G_A, H_A \rightarrow$ Imagens dos pontos P_B e Q_B sobre a isogenia ϕ_A de um remetente;
xi. $G_B, H_B \rightarrow$ Imagens dos pontos P_A e Q_A sobre a isogenia ϕ_B de um destinatário;
xii. $\jmath(E_{AB}), \jmath(E_{BA}) \rightarrow$ Invariante $\jmath$ das curvas elípticas supersingulares E_{AB} e E_{BA}, respectivamente;
xiii e_A, $e_B \rightarrow$ Números inteiros positivos;
xiv. $r_A, r_B \rightarrow$ Inteiros de $\mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ e $\mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$, respectivamente;
xv. $f \rightarrow$ Um cofator para garantir um número primo da forma $p = \ell_A^{e_A}\ell_B^{e_B}f \pm 1$;
xvi. $\mathcal{H} \rightarrow$ Função hash;
xviii. $\mathcal{E}, \mathcal{D} \rightarrow$ Algoritmos de Encriptação e Decriptação, respectivamente;
xviii. pk_A, $\mathsf{pk}_B \rightarrow$ Chaves públicas do remetente e destinatário, respectivamente;
xiv. sk_A, $\mathsf{sk}_B \rightarrow$ Chaves privadas do remetente e destinatário, respectivamente;
xx. $\mathcal{O} \rightarrow$ Ponto especial localizado no infinito. Atua como um elemento neutro em operações de adição de pontos de uma curva elíptica sobre corpos finitos;
xxi. $ker(\phi) \rightarrow$ Kernel de uma isogenia ϕ. Em particular, é um subgrupo finito de uma curva elíptica E sobre um corpo fechado $\bar{\mathbb{F}}_{p^2}$. Pode ser também denotado por $\langle P, Q\rangle$, onde os pontos $P, Q \in E[\bar{\mathbb{F}}_{p^2}]$.

2.2. Parâmetros Públicos

Seja E_0 uma curva elíptica supersingular definida sobre um corpo $\mathbb{F}_{p^2}$. Por conveniência, suponha a escolha de um número primo da forma $p = \ell_A^{e_A}\ell_B^{e_B}f - 1$ onde $\ell_A = 2$, $e_A \geqslant 4$ e $f = 1$ ou $p = 4\ell_A^{e_A}\ell_B^{e_B} - 1$ onde ℓ_A e ℓ_B são primos ímpares e, $f = 4$. Assim, qualquer uma dessas referidas escolhas resultam em um número primo da forma $p = 3(\mathrm{mod}4)$. Isso habilita a representação[1] $\mathbb{F}_{p^2} = \mathbb{F}_p[i]/(i^2 + 1)$ e assegura que a curva $E_0[\mathbb{F}_{p^2}] : y^2 = x^3 + x$ seja supersingular com uma ordem de grupo igual $(\ell_A^{e_A}\ell_B^{e_B})^2$. Adicionalmente, sejam P_A, $Q_A \in \mathbb{F}_{p^2}$ uma base de $E_0[\ell_A^{e_A}]$. De forma similar, outros pontos P_B e Q_B geram $E_0[\ell_B^{e_B}]$. Em outras palavras, $E_0[\ell_A^{e_A}]$ e $E_0[\ell_B^{e_B}]$ são gerados pelo kernel $\langle P_A, Q_A\rangle$ e $\langle P_B, Q_B\rangle$, respectivamente.

2.3. Premissas

Em um canal de comunicação estabelecido no protocolo $\binom{2}{1}$-SIOT, o remetente e o destinatário serão usualmente chamados de Alice e Bob, respectivamente.

1. Seja um esquema de criptografia simétrica $(\mathcal{E}, \mathcal{D})$, de acordo com as definições 1 e 2 de [Chou and Orlandi 2015]. A chave simétrica compartilhada é o valor do invariante $\jmath$ entre duas curvas elípticas supersingulares e isomorfas[2]. Como será visto na figura 1, esse invariante $\jmath$ será submetido a uma função hash $\mathcal{H} = \{H_k : k \in \mathcal{K}\}$ indexada por um conjunto finito $\mathcal{K}$, onde cada H_k é uma função de $\mathbb{F}_{p^2}$;
2. Alice deseja encriptar duas mensagens m_0, $m_1 \in \mathcal{M}$ e enviá-las para Bob. Por sua vez, Bob irá decriptar somente uma dessas duas mensagens e Alice não terá conhecimento de sua escolha;
3. Alice e Bob utilizam um protocolo *coin-flipping* de [Wagner 2016] para compartilharem uma única cadeia aleatória e uniforme de bits w. Isso assegura que nem Alice e nem Bob podem adivinhar antecipadamente ou controlar o valor de w. Em seguida, eles devem utilizar, por exemplo, uma função *hash* nessa cadeia de bits para obter os pontos U e V linearmente independentes. Caso contrário, eles devem gerar uma nova cadeia w.

2.4. Protocolo

A figura 1 mostra uma abstração do funcionamento do protocolo proposto na sua forma mais simples, isto é, $\binom{2}{1}$ - SIOT.

2.4.1. Geração de par de chaves

2.4.1.1 Setup - Remetente

1. Alice escolhe secretamente um valor $r_A \in Z/\ell_A^{e_A}Z$;
2. Ela calcula:
   (a) $ker(\phi_A) = \langle P_A + r_A Q_A\rangle$;
   (b) $\phi_A : E_0 \to E_A$;

[1] Veja [Hoffstein et al. 2014], corolário 2.56.
[2] Veja [Silvermann 2009], proposição III.1.4 (b).

Supersingular Isogeny Oblivious Transfer (SIOT)

Alice		**Bob**
Entrada: $m_0, m_1 \in \mathcal{M}$		Entrada: $\sigma \in \{0,1\}$
Saída: None		Saída: m_σ
$r_A \leftarrow \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$		$r_B \leftarrow \mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$
$\phi_A : E_0 \rightarrow E_A$		$\phi_B : E_0 \rightarrow E_B$
$E_A \leftarrow E_0/\langle P_A + r_A Q_A \rangle$		$E_B \leftarrow E_0/\langle P_B + r_B Q_B \rangle$
$G_A \leftarrow \phi_A(P_B); H_A \leftarrow \phi_A(Q_B)$		$G_B \leftarrow \phi_B(P_A); H_B \leftarrow \phi_B(Q_A)$
$\mathsf{pk}_A \leftarrow (E_A, G_A, H_A)$		$\mathsf{pk}_B \leftarrow (E_B, G_B, H_B)$
	$\xrightarrow{\mathsf{pk}_A}$	Se $G_A, H_A \notin E_A[\ell_B^{e_B}]$, então rejeite ($\perp$)
		$U, V \leftarrow_\$ E_B[\ell_A^{e_A}]$ então,
		$\hat{G}_B \leftarrow (G_B - \sigma U)$
		$\hat{H}_B \leftarrow (H_B - \sigma V)$
		$\hat{\mathsf{pk}}_B \leftarrow (E_B, \hat{G}_B, \hat{H}_B)$
Se $\hat{G}_B, \hat{H}_B \notin E_B[\ell_A^{e_A}]$, então, rejeite ($\perp$)	$\xleftarrow{\hat{\mathsf{pk}}_B}$	
$\forall i \in \{0,1\}$ e $U, V \leftarrow_\$ E_B[\ell_A^{e_A}]$ então,		$\phi_B^{'} : E_A \rightarrow E_{AB}$
$\phi_{A_i}^{'} : E_B \rightarrow E_{BA_i}$		
$E_{BA_i} \leftarrow E_B/\langle(\hat{G}_B + iU) + r_A(\hat{H}_B + iV)\rangle$		$E_{AB} \leftarrow E_A/\langle G_A + r_B H_A \rangle$
$k_0 = \mathcal{H}(j(E_{BA_0}))$		$k_\sigma = \mathcal{H}(j(E_{AB}))$
$k_1 = \mathcal{H}(j(E_{BA_1}))$		
	$\xrightarrow{\substack{c_0 \leftarrow \mathcal{E}_{k_0}(m_0) \\ c_1 \leftarrow \mathcal{E}_{k_1}(m_1)}}$	
		$m_\sigma \leftarrow \mathcal{D}_{k_\sigma}(c_\sigma);$

Figura 1. Protocolo $\binom{2}{1}$ - SIOT.

(c) $\phi_A(P_B) = G_A; \phi_A(Q_B) = H_A$.

3. Alice cria um par de chaves $sk_A = (\phi_A, r_A)$ e $pk_A = (E_A, G_A, H_A)$, isto é, suas chaves privada e pública, respectivamente; e
4. Ela envia para Bob a sua chave pública $pk_A = (E_A, G_A, H_A)$. Ele verifica se $G_A, H_A \in E_A[\ell_B^{e_B}]$, isto é, $\ell_B^{e_B} G_A = \ell_B^{e_B} H_A = \mathcal{O}_A \in E_A$, visto que $\{P_B, Q_B\} \subset E_0[\ell_B^{e_B}]$. Se essa verificação for válida então, Bob aceitará a chave pública de Alice. Caso contrário, a referida chave pública será rejeitada ($\perp$).

Denota-se $E_A = E_0/\langle P_A + r_A Q_A \rangle$ onde $|ker(\phi_A)| = |\langle P_A + r_A Q_A \rangle| = \ell_A^{e_A}$, isto é, uma isogenia separável[3] de grau $\ell_A^{e_A}$.

[3] Veja [Washington. 2008], Proposição 12.8.

2.4.1.2 Setup - Destinatário

1. Bob escolhe secretamente um valor $r_B \in Z/\ell_B^{e_B} Z$;
2. Ele calcula:
   (a) $ker(\phi_B) = \langle P_B + r_B Q_B \rangle$;
   (b) $\phi_B : E_0 \to E_B$;
   (c) $\phi_B(P_A) = G_B$; $\phi_B(Q_A) = H_B$;
   (d) Para um secreto e escolhido $\sigma \in \{0, 1\}$, tem-se que $\hat{G}_B = (G_B - \sigma U)$ e $\hat{H}_B = (H_B - \sigma V)$.
3. Bob cria um par de chaves $sk_B = (\phi_B, r_B)$ e $\hat{pk}_B = (E_B, \hat{G}_B, \hat{H}_B)$, isto é, suas chaves privada e pública, respectivamente; e
4. Ele envia para Alice a sua chave pública $\hat{pk}_B$. Ela executa duas verificações:
   - Se $\hat{G}_B, \hat{H}_B \in E_B[\ell_A^{e_A}]$, isto é, $\ell_A^{e_A} \hat{G}_B = \ell_A^{e_A} \hat{H}_B = \mathcal{O}_B \in E_B$, visto que $\{P_A, Q_A\} \subset E_0[\ell_A^{e_A}]$ e;
   - Se os pontos $U, V \in E_B[\ell_A^{e_A}]$. Isso assegura que o par de pontos $(\hat{G}_B + U, \hat{H}_B + V)$ e $(\hat{G}_B - U, \hat{H}_B - V)$ foi gerado por $E_B[\ell_A^{e_A}]$. Caso contrário, a referida chave pública será rejeitada ($\perp$) e todo o protocolo é reiniciado para executar uma outra cadeia w de bits.

2.4.2. Geração de chaves secretas

2.4.2.1 Setup - Alice

- Alice calcula:

1. Para todo $i \in \{0, 1\}$, $ker(\phi'_{A_i}) = \langle (\hat{G}_B + iU) + r_A(\hat{H}_B + iV) \rangle$;
2. $\phi'_{A_i} : E_B \to E_{BA_i}$;
3. $\jmath_i \leftarrow invariante\, \jmath(E_{BA_i})$;
4. $k_i = \mathcal{H}(\jmath_i)$.

2.4.2.2 Setup - Bob

- Bob calcula:

1. $ker(\phi'_B) = \langle G_A + r_B H_A \rangle$;
2. $\phi'_B : E_B \to E_{AB}$;
3. $\jmath_\sigma \leftarrow invariante\, \jmath(E_{AB})$;
4. $k_\sigma = \mathcal{H}(\jmath_\sigma)$.

2.4.3. Encriptação e Decriptação

1. Para todo $i \in \{0, 1\}$, Alice encripta m_i. Assim, $c_i \leftarrow \mathcal{E}(k_i, m_i)$. Após isso, ela envia (c_0, c_1) para Bob e;
2. Ele decripta e obtém um único $m_\sigma = \mathcal{D}(k_\sigma, c_\sigma)$.

2.5. Análise de segurança do $\binom{2}{1}$-SIOT

2.5.1. Conceitos preliminares para a segurança

Definição 1. *Uma função $\epsilon : \mathbb{N} \to \mathbb{R}$ é definida como desprezável se para cada polinômio positivo $p(\cdot)$, existe um N tal que para todo $n > N$. Então, $\epsilon(n) < \frac{1}{p(n)}$.*

Definição 2. *Um conjunto de probabilidades $\mathcal{X} = \{\mathcal{X}(n, a)\}$ é uma sequência infinita de variáveis aleatórias indexadas por $n \in \mathbb{N}$ e $a \in \{0,1\}^*$. O valor $\underline{n}$ representa um parâmetro de segurança e $\underline{a}$ representa as entradas das partes ou participantes.*

Definição 3. *Dois conjuntos de distribuição $\mathcal{X} = \{\mathcal{X}(n, a)\}$ e $\mathcal{Y} = \{\mathcal{Y}(n, a)\}$ são computacionalmente indistinguíveis, denotado por $\mathcal{X} \overset{c}{\equiv} \mathcal{Y}$, se para cada algoritmo de tempo não-polinomial $\mathcal{D}$, existe uma função desprezável $\epsilon(\cdot)$ tal que para cada $n \in \mathbb{N}$ e $a \in \{0,1\}^*$. Então, $|\Pr[\mathcal{D}(\mathcal{X}(n, a)) = 1] - \Pr[\mathcal{D}(\mathcal{Y}(n, a)) = 1]| \leqslant \epsilon(n)$.*

2.5.2. Problemas computacionais de isogenias entre curvas elípticas supersingulares

Nesta seção, serão apresentados os problemas de decisão e computacionais que garantem a segurança do protocolo SIDH. Para maiores detalhes, o leitor deve consultar [Feo et al. 2014]. Como será visto a seguir, a segurança do $\binom{2}{1}$-SIOT está fundamentada nesses problemas.

Problema 1 (Problema Decisional de Isogenia Supersingular - DSSI)**.** *Seja E_A uma outra curva supersingular definida sobre $\mathbb{F}_{p^2}$. Decida se E_A e E_0 são isogeneas de grau $\ell_A^{e_A}$.*

Problema 2 (Problema Computational de Isogenia Supersingular - CSSI)**.** *Seja $\phi_A : E_0 \to E_A$ uma isogenia cujo kernel[4] é $R_A = \langle [m_A]P_A + [r_A]Q_A \rangle$ onde m_A e r_A são escolhidos aleatoriamente de $\mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ e ambos não são divisíveis por $\ell_A^{e_A}$. Seja ainda a chave pública (E_A, G_A, H_A). Determine R_A.*

Problema 3 (Problema Computacional Diffie-Helmann Supersingular - SSCDH)**.** *Sejam $\phi_A : E_0 \to E_A$ uma isogenia cujo kernel é $R_A = \langle P_A + [r_A]Q_A \rangle$, para algum $r_A \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ e $\phi_B : E_0 \to E_B$ uma isogenia cujo kernel é $R_B = \langle P_B + [r_B]Q_B \rangle$, para algum $r_B \in \mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$. Dado as chaves públicas (E_A, G_A, H_A) e (E_B, G_B, H_B). Determine o invariante $\jmath$ de $E_0/\langle P_A + [r_A]Q_A, P_B + [r_B]Q_B \rangle$.*

Problema 4 (Problema Decisional Diffie-Hellman de Isogenia Supersingular - SSDDH)**.** *Dado uma amostra com probabilidade $1/2$ de uma das duas seguintes distribuições:*

1. *$(E_A, E_B, G_A, H_A, G_B, H_B, E_{AB})$ onde E_A, E_B, G_A, H_A, G_B e H_B como no problema 3 e,*
$$E_{AB} \simeq E_0/\langle P_A + [r_A]Q_A, P_B + [r_B]Q_B \rangle,$$
2. *$(E_A, E_B, G_A, H_A, G_B, H_B, E_C)$ onde E_A, E_B, G_A, H_A, G_B e H_B como no problema 3 e,*

[4] No protocolo $\binom{2}{1}$-SIOT, utilizou-se o kernel $\langle P_A + [r_A]Q_A \rangle$.

$$E_C \simeq E_0/\langle P_A + [\hat{r}_A]Q_A, P_B + [\hat{r}_B]Q_B\rangle,$$

onde $r_A, \hat{r}_A \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ *e* $r_B, \hat{r}_B \in \mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$ *e* $r_A \neq \hat{r}_A$ *e* $r_B \neq \hat{r}_B$. *Além disso, ambos* $(r_A, \hat{r}_A)$ *e* $(r_B, \hat{r}_B)$ *não são divisíveis por* $\ell_A^{e_A}$ *e* $\ell_B^{e_B}$, *respectivamente. Determine a qual distribuição a tupla pertence.*
Nota: *Cabe ressaltar que como cada amostra tem uma probabilidade de* $1/2$ *então, pela definição 3, tem-se que:*

$$\{E_A, E_B, G_A, H_A, G_B, H_B, E_{AB}\} \overset{\mathbf{c}}{\equiv} \{E_A, E_B, G_A, H_A, G_B, H_B, E_C\}.$$

2.5.3. Notações para a análise de segurança

Na análise de segurança do protocolo proposto, as seguintes notações serão utilizadas, a saber:

1. Aplicação da fórmula de Vélu[5] $\rightarrow Vélu's\ formula\{ker(\phi), E\}$;
2. De acordo com [Kalai 2005], em um protocolo OT, a substituição de uma das mensagens do par de entrada m_0 e m_1 por outra mensagem m deve passar despercebida pelo destinatário. Dado $\tau \in \mathcal{M}$ e $\sigma \in \{0, 1\}$, a compreensão ou visão de Alice na execução de um protocolo OT é denotada por $\{\Omega_{Alice}(Alice(1^n, \tau), Bob(1^n, \sigma))\}$, onde um parâmetro de segurança é definido por 1^n. Similarmente, denotamos a compreensão ou visão de Bob por $\{\Omega_{Bob}(Alice(1^n, \tau), Bob(1^n, \sigma))\}$ e;
3. Ambos Alice e Bob como usuários desonestos são denotados por $Alice^*$ e Bob^*, respectivamente.

2.5.4. Alguns requisitos para a Análise de Segurança

Em princípio, qualquer protocolo seguro deverá resistir a qualquer ataque malicioso. Assim, para comprovar que um protocolo OT seja seguro, [Hazay and Lindell. 2010] afirmam que os mais importantes requisitos em qualquer protocolo de segurança são a *corretude* e *privacidade*.
Corretude. No $\binom{2}{1}$-SIOT, suponhamos que tanto Alice quanto Bob são partes honestas na execução do protocolo. Sejam $\sigma, i \in \{0, 1\}$ tal que $\sigma = i$. Então, a *corretude* do protocolo segue pelas identidades abaixo.

$$\begin{aligned}
\jmath(E_{BA_i}) \simeq & \jmath(E_B/\langle (G_B - \sigma \cdot U + i \cdot U) + r_A \cdot (H_B - \sigma \cdot V + i \cdot V)\rangle) \\
\simeq & \jmath(\phi_B(E_0)/\langle (\phi_B(P_A) - \sigma \cdot U + i \cdot U) + r_A(\phi_B(Q_A) - \sigma \cdot V + i \cdot V)\rangle) \\
\simeq & \jmath(\phi_B(E_0)/\langle \phi_B(P_A) + r_A \cdot \phi_B(Q_A)\rangle) \\
\simeq & \jmath(\phi_B(E_0)/\langle \phi_B(P_A + r_A \cdot Q_A)\rangle) \\
\simeq & \jmath(\phi'_{A_i}(\phi_B(E_0))) \simeq \jmath(\phi_A(E_0)/\langle \phi_A(P_B + r_B \cdot Q_B)\rangle) \\
\simeq & \jmath(\phi'_{A_i}(\phi_B(E_0))) \simeq \jmath(E_B/\langle \phi_A(P_B) + r_B \cdot \phi_B(Q_B)\rangle) \\
\simeq & \jmath(\phi'_{A_i}(\phi_B(E_0))) \simeq \jmath(E_B/\langle G_A + r_B \cdot H_A\rangle) \simeq \jmath(\phi'_B(\phi_A(E_0))) \simeq \jmath(E_{AB}).
\end{aligned}$$

Privacidade. No protocolo $\binom{2}{1}$-SIOT, as escolhas de Bob não devem ser conhecidas por Alice. Além disso, no final da execução do referido protocolo, Bob não poderá obter

[5] Veja [Galbraith. 2012], corolário 25.1.7.

nenhum conhecimento sobre a mensagem que ele não decriptou. Cabe ressaltar que essa privacidade decorre da dificuldade de resolução dos problemas computacionais vistos na seção 2.5.2. Por fim, para complementar a prova de segurança do protocolo $\binom{2}{1}$-SIOT , elaborou-se o Teorema 1 a seguir:

Teorema 1. *Suponha que os problemas computacionais de isogenia supersingular CSSI, SSCDH e SSDDH são difíceis no grupo $E(\mathbb{F}_{p^2})$. Então, o protocolo $\binom{2}{1}$-SIOT assegura a privacidade entre duas partes.*

Demonstração. A prova consiste em adaptar a definição 2.6.1 de [Hazay and Lindell. 2010] ao protocolo $\binom{2}{1}$-SIOT pela compatibilização com os problemas computacionais citados acima. Sejam duas mensagens, m_0 e m_1, entre duas partes (Alice e Bob). Um protocolo OT é privado se os seguintes requisitos forem válidos:

i. Não-trivialidade

Se Alice e Bob seguem corretamente o protocolo, então depois de uma execução em que Alice possui como entrada os textos claros m_0, $m_1 \in \mathcal{M}$ e Bob possui como entrada um bit $\sigma \in \{0, 1\}$, a saída de Bob será um único texto decriptado m_σ. Em outras palavras, Bob recebe a chave pública pk_A e o par de textos encriptados (c_0, c_1) de Alice. Relembrando que $pk_A \leftarrow (E_A, G_A, H_A)$ e $c_\sigma \leftarrow \mathcal{E}(k_\sigma, m_\sigma)$ são bem definidos no protocolo $\binom{2}{1}$-SIOT . Assim, a não-trivialidade provém da seguinte relação:

$$V\acute{e}lu's\ formula\{\langle G_A + r_B H_A\rangle, E_A\} \Rightarrow E_{AB} \therefore$$
$$\mathcal{H}(\jmath(E_{AB})) \Rightarrow k_\sigma.$$

Portanto, Bob recupera k_σ o que implica em $m_\sigma \leftarrow \mathcal{D}(k_\sigma, c_\sigma)$ tal que σ é um único valor escolhido secretamente por ele. Além disso, após receber pk_A, Bob não será capaz de calcular a chave privada (ϕ_A, r_A) de Alice. Caso contrário, haveria uma violação da hipótese de dificuldade do *problema CSSI*.

ii. Privacidade no caso de Bob ser uma parte desonesta

Seja $\hat{pk}_B \leftarrow (E_B, \hat{G}_B, \hat{H}_B)$ a chave pública de Bob enviada para Alice. Relembre que $\hat{G}_B \leftarrow G_B$, $\hat{H}_B \leftarrow H_B$, se $\sigma = 0$ e $\hat{G}_B \leftarrow (G_B - U)$, $\hat{H}_B \leftarrow (H_B - V)$, se $\sigma = 1$. Além disso, há a chave pública $pk_A \leftarrow (E_A, G_A, H_A)$ de Alice enviada para Bob e um único valor do invariante $\jmath_\sigma \leftarrow j(V\acute{e}lu's\ formula\{\langle G_A + r_B H_A\rangle, E_A\})$ calculado por Bob ao receber a bem definida chave pública pk_A de Alice. Após isso, $\forall\, i \in \{0, 1\}$, Alice computará $\jmath_i \leftarrow \jmath(V\acute{e}lu's\ formula\{\langle(\hat{G}_B + iU) + r_A(\hat{H}_B + iV)\rangle, E_B\})$, isto é, $\jmath_0$ e $\jmath_1$.

Adicionalmente, Alice compartilhará uma única chave secreta com Bob. Assim, a privacidade de Alice é baseada no seguinte fato: Bob não pode computar ambos os valores dos invariantes $\jmath_0$ and $\jmath_1$ ($\jmath_0 \neq \jmath_1$) se a hipótese do problema SSCDH é difícil. Em outras palavras, Bob será capaz de computar um único valor de um invariante $\jmath_\sigma$.

Além disso, seja $\sigma \in \{0, 1\}$ uma entrada auxiliar e uma outra entrada com a tupla m_0, m_1, $m \in \mathcal{M}$. Assim, uma outra forma de visualizar a privacidade de Alice é que a primeira mensagem de Bob, denotada por Bob* $(1^n, \sigma)$, determina qual dos textos m_0 ou m_1 será recebido por ele. Por exemplo, se é determinado que o texto m_0 deve ser recebido, então a visualização de Bob quando a entrada de Alice

corresponde a (m_0, m_1) é indistinguível de sua visualização quando a entrada de Alice corresponde a (m_0, m). Evidentemente, isso implica que Bob não pode aprender nada sobre o texto m_1 quando o texto m_0 é recebido e vice-versa. Então,

$$\{\Omega_{Bob^*}(Alice(1^n,(m_0,m_1));Bob^*(1^n,\sigma))\}_{n\in\mathbb{N}} \stackrel{\mathbf{c}}{\equiv} \{\Omega_{Bob^*}(Alice(1^n,(m_0,m));Bob^*(1^n,\sigma))\}_{n\in\mathbb{N}}$$

ou

$$\{\Omega_{Bob^*}(Alice(1^n,(m_0,m_1));Bob^*(1^n,\sigma))\}_{n\in\mathbb{N}} \stackrel{\mathbf{c}}{\equiv} \{\Omega_{Bob^*}(Alice(1^n,(m,m_1));Bob^*(1^n,\sigma))\}_{n\in\mathbb{N}}$$

iii. Privacidade no caso de Alice ser uma parte desonesta

Observe que este requisito mostra que Alice não consegue distinguir as possíveis escolhas secretas de Bob, isto é, quando o bit σ assume o valor 0 ou 1, isso é indistinguível na visão de Alice. Em outras palavras, ela visualiza simplesmente uma chave pública $\hat{pk}_B$. Assim, a privacidade do destinatário será verificada pelo seguinte Lema:

Lema 1. *Ao receber a chave pública $\hat{pk}_B$ de Bob, Alice não pode adivinhar o valor do bit σ a uma probabilidade maior que $1/2 + \epsilon(n)$, para alguma função desprezável $\epsilon(n)$ e $\forall\, n \in \mathbb{N}$.*

Demonstração. Suponha que no recebimento da chave pública $\hat{pk}_B$ e não conhecendo o valor secreto do bit σ de Bob, Alice não pode distinguir entre os pares de tupla $\{(E_B, G_B, H_B)\}$ e $\{(E_B, (G_B - U), (H_B - V)\}$, isto é, para algum $\hat{G}_B, \hat{H}_B \in E_B[\ell_A^{e_A}]$ tal que $\Pr[(E_B, G_B, H_B) = (E_B, \hat{G}_B, \hat{H}_B)]$ = $\Pr[(E_B, G_B - U, H_B - V) = (E_B, \hat{G}_B, \hat{H}_B))]$ é independente do valor de σ. Assim, a dificuldade dessa indistinguibilidade entre essas tuplas é baseada na hipótese de dificuldade do *problema SSDDH*. Então, tem-se que:

$$\{(E_B, G_B, H_B)\} \stackrel{\mathbf{c}}{\equiv} \{E_B, (G_B - U), (H_B - V)\}$$

De acordo com [Rogaway 2004], um diferenciador é um algoritmo probabilístico que descreve a vantagem de um adversário. Assim, suponha que, por contradição, existe um diferenciador probabilístico Θ de tempo polinomial e uma função não-desprezável ϵ tal que para todo n, tem-se que

$$|P_r[\Theta(E_B, G_B, H_B) = 1] - P_r[\Theta(E_B, (G_B - U), (H_B - V)) = 1]| \geqslant \epsilon(n),$$

Desta forma, subtraindo e somando o seguinte termo,

$$P_r[\Theta(E_B, (G_B - U - R), (H_B - V - S)) = 1]^6$$

Tem-se que
$|P_r[\Theta(E_B, G_B, H_B) = 1] - P_r[\Theta(E_B, (G_B - U), (H_B - V)) = 1]| \leqslant$
$|P_r[\Theta(E_B, G_B, H_B) = 1] - P_r[\Theta(E_B, (G_B - U - R), (H_B - V - S)) = 1]| +$
$|P_r[\Theta(E_B, (G_B - U - R), (H_B - V - S)) = 1] - P_r[\Theta(E_B, (G_B - U), (H_B - V)) = 1]|$
Por contradição, suponhamos que

$$|P_r[\Theta(E_B, G_B, H_B) = 1] - P_r[\Theta(E_B, (G_B - U - R), (H_B - V - S)) = 1]| \geqslant \frac{\epsilon(n)}{2} \quad (1)$$

[6] $R, S \in E_B[\ell_A^{e_A}]$.

ou

$$|P_r[\Theta(E_B,(G_B-U-R),(H_B-V-S))=1]-P_r[\Theta(E_B,(G_B-U),(H_B-V))=1]|\geqslant\frac{\epsilon(n)}{2} \quad (2)$$

Suponha que a equação 1 seja válida. Então, pode-se construir um diferenciador $\tilde{\Theta}$ para o problema SSDDH que funciona da seguinte forma: Após receber a chave pública $\hat{p}k_B \leftarrow (E_B,(G_B-U),(H_B-V))$, o diferenciador $\tilde{\Theta}$ aleatoriamente escolhe os pares de pontos R, S e, assim, $\hat{p}k'_B \leftarrow (E_B,(G_B-U-R),(H_B-V-S))$. Por outro lado, se $\hat{p}k_B \leftarrow (E_B,G_B,H_B)$ então, $\hat{p}k'_B \leftarrow (E_B,(G_B-R),(H_B-S))$. Observe que os pares de pontos U e V não são utilizados nesta última tupla. Contudo, esses pontos, assim como os outros pontos R e S, são oriundos do mesmo grupo $E_B[\ell_A^{e_A}]$ e poderiam ser também aleatoriamente escolhidos pelo diferenciador $\tilde{\Theta}$, digo os pontos U e V. Portanto, tem-se que $\hat{p}k'_B \leftarrow (E_B,G_B,H_B)$ e,

$$|P_r[\tilde{\Theta}(E_B,G_B,H_B)=1]-P_r[\tilde{\Theta}(E_B,(G_B-U),(H_B-V))=1]|=$$
$$|P_r[\Theta(E_B,G_B,H_B)=1]-P_r[\Theta(E_B,(G_B-U-R),(H_B-V-S))=1]|\geqslant\frac{\epsilon(n)}{2},$$

o que contradiz com as hipóteses de dificuldade do problema SSDDH. De modo análogo, tem-se um resultado similar caso a equação 2 como válida.
A prova da privacidade de Bob é concluída por observar que $\{(E_B,G_B,H_B)\}$, $\{(E_B,(G_B-U),(H_B-V))\}$, independentemente do valor de σ, são indistinguíveis na visualização de Alice. Em outras palavras, seja $\tau \in \{0,1\}^*$ uma entrada auxiliar então,

$$\{\Omega_{Alice^*}(Alice^*(1^n,\tau),Bob(1^n,0))\} \stackrel{\mathbf{c}}{\equiv} \{\Omega_{Alice^*}(Alice^*(1^n,\tau),Bob(1^n,1))\}.$$

□

De acordo com o Lema 1, tem-se que a privacidade de Bob fundamenta-se na hipótese de dificuldade do problema SSDDH sobre um grupo $E_B[\ell_A^{e_A}]$.

□

2.5.5. Análise algébrica de segurança do protocolo $\binom{2}{1}$-SIOT

Considere o hipotético caso de uma Alice desonesta onde ela tentará utilizar um diferenciador baseado em emparelhamento de *Weil* para descobrir o valor secreto do bit σ de um Bob honesto. Em uma segunda situação hipotética, as funções são invertidas, isto é, Alice será considerada como uma remetente honesta e Bob como um destinatário desonesto. Desta forma, uma análise algébrica é desenvolvida de tal forma que certas condições devem ser obedecidas para que Bob não seja capaz de decriptar ambas as mensagens transmitidas por Alice.

2.5.5.1 Prevenindo contra uma Alice desonesta que utiliza um diferenciador baseado em emparelhamento de *Weil*

Considere a situação em que Alice, atuando como um remetente desonesto, recebe a chave pública $(E_B, \hat{G}_B, \hat{H}_B)$ de Bob. Assim, em princípio, ela não tem o conhecimento se recebeu a chave pública (E_B, G_B, H_B) ou $(E_B, G_B - U, H_B - V)$. Desta forma, Alice utilizará o emparelhamento de *Weil* para diferenciar esses dois valores. No que se segue, todos os emparelhamentos tem ordem $\ell_A^{e_A}$. Durante a execução do protocolo, os pontos $G_B = \phi_B(P_A)$ e $H_B = \phi_B(Q_A)$ são bem definidos para satisfazer a relação $e(G_B, H_B) = e(P_A, Q_A)^{\ell_A^{e_A}}$. Se essa relação não for válida para ambos $(\hat{G}_B, \hat{H}_B)$ ou $(\hat{G}_B + U, \hat{H}_B + V)$, isso revelará qual a chave pública que foi escolhida por Bob. Em geral, Alice pode somar qualquer múltiplo de (U, V) em $(\hat{G}_B, \hat{H}_B)$ e observar por uma incompatibilidade. Assim, é preciso ter $e(\hat{G}_B + \lambda U, \hat{H}_B + \lambda V) = e(G_B, H_B) = e(P_A, Q_A)^{\ell_A^{e_A}}$ para algum $\lambda \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$. Relembrando que $U, V, G_B, H_B \in E_B[\ell_A^{e_A}]$, então eles podem ser escritos como uma combinação linear, isto é, $U = \alpha G_B + \beta H_B$ e $V = \gamma G_B + \delta H_B$ tal que $\alpha, \beta, \gamma, \delta \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$. Assim, isto significa que:

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

Desta forma, é necessário que $(1+\lambda\alpha)(1+\lambda\delta) - \lambda^2\beta\gamma = 1 \pmod{\ell_A^{e_A}}$, ou equivalentemente $\lambda(\alpha+\delta) + \lambda^2(\alpha\delta - \beta\gamma) = 0 \pmod{\ell_A^{e_A}}$. Isto deve ser válido para qualquer λ, em particular para aqueles que são invertíveis mod $\ell_A^{e_A}$, e então, deve ser válido que $\lambda(\alpha\delta - \beta\gamma) = -(\alpha+\delta) \pmod{\ell_A^{e_A}}$. Mais uma vez, isto pode somente ser válido para algum λ se $\alpha\delta - \beta\gamma = 0 \pmod{\ell_A^{e_A}}$ e $\alpha + \delta = 0 \pmod{\ell_A^{e_A}}$, ou equivalentemente, $\delta = -\alpha \pmod{\ell_A^{e_A}}$ e $\alpha^2 + \beta\gamma = 0 \pmod{\ell_A^{e_A}}$. Portanto, tais condições devem ser obedecidas para evitar que Alice descubra o valor secreto do bit σ de Bob. □

2.5.5.2 Prevenindo contra uma possível decriptação indesejada de um Bob desonesto

Relembre que $U, V \in E_B[\ell_A^{e_A}]$ são pontos linearmente independentes e que tais pontos podem ser expressos por $U = \alpha G_B + \beta H_B$, $V = \gamma G_B + \delta H_B$. Suponha que

Alice receba a chave pública $(E_B, \hat{G}_B, \hat{H}_B)$ de Bob. Assim, Alice irá computar a isogenia $\phi'_{A_0} : E_B \rightarrow E_{BA_0}$, com grau $\ell_A^{e_A}$, cujo kernel é $ker(\phi'_{A_0}) = \langle G_B + r_A H_B \rangle$. Analogamente, ela irá calcular a isogenia $\phi'_{A_1} : E_B \rightarrow E_{AB_1}$, com grau $\ell_A^{e_A}$, cujo kernel é $ker(\phi_{A_1}) = \langle (G_B + U) + r_A(H_B + V) \rangle$. Ressalta-se que[7] se $ker(\phi'_{A_0}) \subsetneqq ker(\phi_{A_1})$ então, a isogenia E_{BA_0} é isomorfa com relação a isogenia E_{BA_1}, isto é, denota-se $E_{BA_0} \cong E_{BA_1}$. Adicionalmente, se ϕ_{A_1} é separável, então existe uma única isogenia $\hat{\phi_A} : E_{BA_0}. \rightarrow E_{BA_1}$. Nesse momento, considere $(G_B + U) + r_A(H_B + V) = (G_B + \alpha G_B + \beta H_B) + r_A(H_B + \gamma G_B + \delta H_B) = (1 + \alpha + \gamma r_A)G_B + (r_A + \beta + \delta r_A)H_B$. Assim, por inspeção, esse ponto somente pode pertencer ao kernel $\langle G_B + r_A H_B \rangle$ considerando as seguintes condições:

1. $(1 + \alpha + \gamma r_A)$ é invertível mod $\ell_A^{e_A}$ (i.e. if $\ell_A \nmid 1 + \alpha + \gamma r_A$);
2. $(r_A + \beta + \delta r_A)/(1 + \alpha + \gamma r_A) = r_A \pmod{\ell_A^{e_A}}$, que significa $\gamma r_A^2 + (\alpha + \delta) r_A - \beta = 0 \pmod{\ell_A^{e_A}}$ e, assim, $\gamma r_A^2 + (\alpha - \delta) r_A - \beta = 0 \pmod{\ell_A}$. Então, uma simples restrição nos coeficientes assegura que essa última equação não tenha solução. Desta forma, arbitrariamente define-se $\ell_A \mid \gamma$ and $\ell_A \mid (\alpha - \delta)$, no entanto $\ell_A \nmid \beta$.

Portanto, é importante que a equação $\gamma r_A^2 + (\alpha - \delta) r_A - \beta = 0 \pmod{\ell_A}$ não tenha solução porque, caso contrário, se Alice e Bob não podem controlar os coeficientes α, β, γ, δ, além de garantir as condições supracitadas, Bob poderia ser capaz de decriptar ambas as mensagens de Alice. □

2.5.5.3 Sintetizando as condições algébricas

Nesta seção, são consideradas as três condições, envolvendo os coeficientes α, β, γ e δ, baseadas nas equações obtidas nas seções 2.5.5.1 e 2.5.5.2 para garantir a segurança do protocolo $\binom{2}{1}$-SIOT quando Alice e Bob são partes desonestas. Assim, as premissas dos referidos coeficientes são alcançadas de forma a assegurar que Alice não seja capaz de obter a escolha secreta do valor do bit σ de Bob e que ele não seja capaz de decriptar ambas as mensagens cifradas do par c_0 e c_1 enviadas por Alice. Combinando essas relações, tem-se $\gamma = -\alpha^2/\beta \pmod{\ell_A^{e_A}}$ desde que β é invertível mod $\ell_A^{e_A}$. Em particular, isso significa que $V = -(\alpha/\beta)U$. □

3. Conclusão da segurança do protocolo $\binom{2}{1}$-SIOT

A prova de segurança do protocolo $\binom{2}{1}$-SIOT foi fundamentada em 03 (três) partes, a saber:

i. As características de segurança herdadas do protocolo de [Feo et al. 2014], isto é, os problemas computacionais mencionados na seção 2.5.2 ;

ii. A prova da privacidade entre um remetente e destinatário em um canal de comunicação, por meio do Teorema 1 e;

iii. Uma análise algébrica que utilizou o emparelhamento de *Weil* que definiu algumas condições necessárias para que um remetente desonesto não fraude a segurança do $\binom{2}{1}$-SIOT em relação a um destinatário desonesto e vice-versa.

[7] Veja [Galbraith. 2012], Teorema 9.6.18 e [Washington. 2008], Proposição 12.12.

4. Implementação do protocolo proposto

O protocolo $\binom{2}{1}$-SIOT foi implementado na linguagem *python*, utilizando um MacBook `Air` com um processador de 1.6 GHz Intel Core i.5, memória de 4GB, 1.600MHz e DDR3. A tabela 1 mostra os valores dos números primos p que foram usados na implementação do protocolo proposto.

Tabela 1. Valores utilizados de p no protocolo $\binom{2}{1}$-SIOT.

$p = \ell_A^{e_A} \ell_B^{e_B} f \pm 1$	Valor	Tamanho (bits)
$(3^4 \cdot 5^3 \cdot 4) - 1$	40.499	16
$(3^6 \cdot 5^3 \cdot 4) - 1$	364.499	19
$(3^7 \cdot 5^4 \cdot 4) - 1$	5.467.499	23
$(3^{11} \cdot 5^6 \cdot 4) - 1$	11.071.687.499	34

Neste trabalho, utilizou-se um valor máximo de p correspondente ao tamanho de 34 bits. Evidentemente, tais valores de p são insuficientes para garantir a segurança do protocolo proposto. [Azarderakhsh et al. 2016], [Feo et al. 2014] consideram que o tamanho do valor de p para a segurança de um protocolo criptográfico pós-quântico baseado em isogenias de curvas elípticas seja no mínimo igual a 512 bits. No entanto, isso não invalida a prova de conceito do $\binom{2}{1}$-SIOT. Além disso, considerando que o tipo de operação mais onerosa no nosso protocolo é o cálculo de isogenias e, assim, ao compará-lo com os dois protocolos OT pós-quânticos de [Vitse 2019], verificou-se que o $\binom{2}{1}$-SIOT requer menos operações desse tipo.

5. Conclusão

Neste artigo, foi apresentada uma proposta de um protocolo pós-quântico chamado $\binom{2}{1}$-SIOT cuja segurança é fundamentada na dificuldade de um adversário calcular isogenias entre curvas elípticas supersingulares e na inspiração da relativa simplicidade do protocolo OT de[Chou and Orlandi 2015] para assegurar a privacidade entre o remetente e destinatário. No que se refere a essa privacidade, foi importante elaborar um teorema compatibilizando uma definição de privacidade OT de [Hazay and Lindell. 2010] com problemas computacionais de isogenias entre curvas elípticas supersingulares de [Feo et al. 2014], considerando um cenário hipotético entre um remetente desonesto e um destinatário honesto, e vice-versa. Por fim, uma análise algébrica com o emparelhamento de *Weil* delineou certas condições necessárias para que não houvesse violações de segurança e privacidade no protocolo proposto.

Referências

Azarderakhsh, R., Koziel, B., Jalali, A., Kermani, M. M., and Jao, D. (2016). Neonsidh: Efficient implementation of supersingular isogeny diffe - hellman key exchange protocol on arm. Cryptology ePrint Archive, Report 2016/669.

Barak, B. (2007). Oblivious transfer and private information retrieval. https://www.cs.princeton.edu/courses/archive/fall07/cos433/lec19.pdf.

Bennett, C. H., Brassard, G., Crépeau, C., and Skubiszewska, M.-H. (1992). Pratical quantum oblivious transfer. In Feigenbaum, J., editor, *Advances in Cryptology — CRYPTO '91: Proceedings*, pages 351–366, Berlin, Heidelberg. Springer Berlin Heidelberg.

Chou, T. and Orlandi, C. (2015). The simplest protocol for oblivious transfer. In Lauter, K. and Rodríguez-Henríquez, F., editors, *Progress in Cryptology – LATINCRYPT 2015: 4th International Conference on Cryptology and Information Security in Latin America, Guadalajara, Mexico, August 23-26, 2015, Proceedings*. Springer International Publishing.

Crepeau, C. and Kilian, J. (1988). Achieving oblivious transfer using weakened security assumptions. In *29th Annual Symposium on Foundations of Computer Science*, pages 42–52.

Even, S., Goldreich, O., and Lempel, A. (1985). A randomized protocol for signing contracts. *ACM*, 28(6):637–647.

Feo, L. D., Jao, D., and Plût, J. (2014). Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247.

Galbraith., S. D. (2012). *Mathematics of public key cryptography*. Cambridge University Press, Cambridge.

Hazay, C. and Lindell., Y. (2010). *Efficient Secure Two - Party Protocols - Techniques and Constructions*. Springer Berlin Heidelberg.

Hoffstein, J., Pipher, J., and Silverman, J. H. (2014). *An introduction to mathematical cryptography*. Undergraduate Texts in Mathematics. Springer, New York, second edition edition.

Kalai, Y. T. (2005). Smooth projective hashing and two-message oblivious transfer. *Advances in Cryptology – EUROCRYPT 2005.*, 3494.

Kazmi, R. A. (2015). *Cryptography from Post-quantum Assumptions*. McGill theses. McGill University Libraries.

Rabin, M. O. (1981). How to exchange secrets with oblivious transfer. Cryptology ePrint Archive, Report 2005/187.

Rogaway, P. (2004). On the role of definitions in and beyond cryptography. *Springer and Berlin and Heidelberg*, 3321.

Silvermann, J. H. (2009). *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer, Dordrecht, second. edition.

Vitse, V. (2019). Simple oblivious transfer protocols compatible with kummer and supersingular isogenies. *hal-01981552*.

Wagner, D. (2016). Technical perspective: Fairness and the coin flip. *Communications of the ACM.*, 59(4):75.

Washington., L. C. (2008). *Elliptic curves - Number Theory and Cryptography*. Taylor & Francis Group. LLC, second edition. edition.