Algebraic Eraser: Key Exchange Protocol, Hickory Hash Functions and Attacks

Lindsey Cioffi, Jonathan Katz, Jiahui Liu, Elijah Soria

December 1, 2017

Abstract

We give an overview of the Algebraic Eraser and the Colored Burau Key Agreement Protocol (CBKAP), following Anshel et al. [1]. We provide a worked example of the protocol with small parameters in order to elucidate the scheme. Then we discuss a recent attack on this protocol due to Ben-Zvi, Blackburn, and Tsaban [2]. In 2016, the Hickory Hash functions are designed based on a modified version of Algebraic Eraser function and we describe the malleability, an insecure property we find about the Hickory hash functions.

1 Introduction

Alice and Bob wish to communicate over an insecure channel and there exist efficient and secure methods if they share a secret (key):Symmetric encryption (AES,. . .). The important thing is to decide a shared secret key over an insecure channel. We know that the Diffie-Hellman Key Exchange Protocol(1976) solves this problem as the most important breakthrough in cryptography.

2 Preliminaries

We begin by introducing the notion of braid groups. The *Artin braid group* B_m [4] is the group generated by elements $\sigma_1, \ldots, \sigma_{m-1}$ satisfying the following "braid relations":

- For all $i, j \in \{1, 2, \ldots, m-1\}$ with $|i - j| \geq 2$, it holds that $\sigma_i \sigma_j = \sigma_j \sigma_i$.
- For all $k \in \{1, 2, \ldots, m-2\}$, it holds that $\sigma_k \sigma_{k+1} \sigma_k = \sigma_{k+1} \sigma_k \sigma_{k+1}$.

Let $n \geq 7$ be an integer and $q > n$ be prime. Let $t = (t_1, \ldots, t_n, t_1^{-1}, \ldots, t_n^{-1})$ be commutative indeterminates and their inverses. Define

$$x_1 = \begin{bmatrix} -t_1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n \times n} \text{ and}$$

$$x_i = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & 0 & 0 & & \\ & & t_i & -t_i & 1 & & \\ & & 0 & 0 & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n\times n} \text{ for } 2 \le i \le n-1.$$

That is, x_i (for $i > 1$) is the identity matrix but with the $(i, i-1)$th entry set to t_i, the (i,i)th entry set to $-t_i$, and the $(i,i+1)$th entry set to 1. Each x_i is invertible since

$$x_1^{-1} = \begin{bmatrix} -t_1^{-1} & t_1^{-1} & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n\times n} \text{ and}$$

$$x_i^{-1} = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & 0 & 0 & & \\ & & 1 & -t_i^{-1} & t_i^{-1} & & \\ & & 0 & 0 & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n\times n} \text{ for } 2 \le i \le n-1.$$

Let $M \le \mathrm{GL}_n(\mathbb{F}_q[t])$ denote the subgroup generated by $\{x_1, \ldots, x_{n-1}\}$ under matrix multiplication. (This is called the *reduced Burau representation* of B_n.) Fixing nonzero elements $\kappa_1, \ldots, \kappa_n \in \mathbb{F}_q$, we define the evaluation homomorphism $\varphi : M \to \mathrm{GL}_n(\mathbb{F}_q)$ in which the value κ_i is substituted for the indeterminate t_i.

Let S_n be the symmetric group on n elements, with the identity denoted by e. For $i \in \{1, 2, \ldots, n-1\}$, let $s_i = (i \;\; i+1) \in S_n$ be the transposition of elements i and $i+1$. (Elements of S_n can be represented either as a product of transpositions or in cyclic notation, as these are polynomially equivalent.) We let $\cdot$ denote the group operation in the direct-product group $\mathrm{GL}_n(\mathbb{F}_q) \times S_n$. Also, let $M \rtimes S_n$ denote the *colored Burau group* with group operation $\circ$ defined by $(m_1, \sigma_1) \circ (m_2, \sigma_2) = ({}^{\sigma_1}m_1 m_2, \sigma_1\sigma_2)$, where ${}^{\sigma_1}m_2$ is the action of σ_1 on m_2 in which the indeterminates $t_1, \ldots, t_n$ are permuted according to σ_1.

Define the function $* : (\mathrm{GL}_n(\mathbb{F}_q) \times S_n) \times (M \rtimes S_n) \to \mathrm{GL}_n(\mathbb{F}_q) \times S_n$ by

$$(n, \sigma_1) * (m, \sigma_2) = (n\varphi({}^{\sigma_1}m), \sigma_1\sigma_2).$$

The following lemma is used crucially in the key-agreement protocol.

Lemma 2.1. *For all $(n, \sigma) \in GL_n(\mathbb{F}_q) \times S_n$ and $(m_1, \sigma_1), (m_2, \sigma_2) \in M \rtimes S_n$,*

$$((n, \sigma) * (m_1, \sigma_1)) * (m_2, \sigma_2) = (n, \sigma) * ((m_1, \sigma_1) \circ (m_2, \sigma_2)).$$

Proof. We have:

$$\begin{aligned}((n,\sigma)*(m_1,\sigma_1))*(m_2,\sigma_2) &= (n\varphi({}^{\sigma}m_1),\sigma\sigma_1)*(m_2,\sigma_2)\\ &= (n\varphi({}^{\sigma}m_1)\varphi({}^{\sigma\sigma_1}m_2),\sigma\sigma_1\sigma_2)\\ &= (n\varphi({}^{\sigma}m_1^{\sigma\sigma_1}m_2),\sigma\sigma_1\sigma_2)\\ &= (n\varphi({}^{\sigma}(m_1^{\sigma_1}m_2)),\sigma\sigma_1\sigma_2)\\ &= (n,\sigma)*(m_1^{\sigma_1}m_2,\sigma_1\sigma_2)\\ &= (n,\sigma)*((m_1,\sigma_1)\circ(m_2,\sigma_2)),\end{aligned}$$

as desired. □

Elements $(a,\sigma),(b,\sigma')\in M\rtimes S_n$ are said to be $*$-*commuting* if

$$(\varphi(a),\sigma)*(b,\sigma')=(\varphi(b),\sigma')*(a,\sigma).$$

Two sets S_1,S_2 are $*$-commuting if s_1,s_2 are $*$-commuting for all $s_1\in S_1$ and $s_2\in S_2$.

One can show that the set $\{(x_1,s_1),\ldots,(x_{n-1},s_{n-1})\}$ generates $M\rtimes S_n$. Let $A=\{(x_{l_1},s_{l_1}),\ldots,(x_{l_\alpha},s_{l_\alpha})\}$ and $B=\{(x_{r_1},s_{r_1}),\ldots,(x_{r_\beta},s_{r_\beta})\}$ be subsets of these generating elements such that $|\ell_i-r_j|\geq 2$ for all i,j. Define

$$\begin{aligned}A^1 &= \{(x_{l_1},s_{l_1})^{-1},\ldots,(x_{l_\alpha},s_{l_\alpha})^{-1}\}\\ B^{-1} &= \{(x_{r_1},s_{r_1})^{-1},\ldots,(x_{r_\beta},s_{r_\beta})^{-1}\};\end{aligned}$$

That is, A^{-1} and B^{-1} are the sets containing the (right) inverses of the elements of A and B, respectively, in the group $M\rtimes S_n$. We have

$$\begin{aligned}A^{-1} &= \{({}^{s_{l_1}^{-1}}x_{l_1}^{-1},s_{l_1}^{-1}),\ldots,({}^{s_{l_\alpha}^{-1}}x_{l_\alpha}^{-1},s_{l_\alpha}^{-1})\}=\{({}^{s_{l_1}}x_{l_1}^{-1},s_{l_1}),\ldots,({}^{s_{l_\alpha}}x_{l_\alpha}^{-1},s_{l_\alpha})\}\\ B^{-1} &= \{({}^{s_{r_1}^{-1}}x_{r_1}^{-1},s_{r_1}^{-1}),\ldots,({}^{s_{r_\beta}^{-1}}x_{r_\beta}^{-1},s_{r_\beta}^{-1})\}=\{({}^{s_{r_1}}x_{r_1}^{-1},s_{r_1}),\ldots,({}^{s_{r_\beta}}x_{r_\beta}^{-1},s_{r_\beta})\},\end{aligned}$$

using the fact that $s_i=s_i^{-1}$ (since s_i is simply a transposition).

Claim 2.2. *$A\cup A^{-1}$ and $B\cup B^{-1}$ are $*$-commuting.*

Proof. Let $(\tilde{x}_k(t),s_k)\in A\cup A^{-1}$ and $(\tilde{x}_l(t),s_l)\in B\cup B^{-1}$ be arbitrary. Without loss of generality, assume $k<l$, so $l=k+j$ for some $2\leq j<n-1$. It is clear that $s_ks_l=s_ks_{k+j}=s_{k+j}s_k=s_ls_k$ since $j\geq 2$. It is also easily verified that $\tilde{x}_k(t)\tilde{x}_l(t)=\tilde{x}_k(t)\tilde{x}_{k+j}(t)=\tilde{x}_{k+j}(t)\tilde{x}_k(t)=\tilde{x}_l(t)\tilde{x}_k(t)$ due to the fact that $j\geq 2$ and the structure of $x_i(t)$ and $x_i(t)^{-1}$ each contain only non-zero elements along the main diagonal and to the left and right of the main diagonal in the i^{th} row of each matrix. Similarly, since $2\leq j<n-1$, ${}^{s_l}\tilde{x}_k(t)=\tilde{x}_k(t)$ and ${}^{s_k}\tilde{x}_l(t)$ due to the fact that neither s_l nor s_k permute the elements of t found in $\tilde{x}_k(t)$ and $\tilde{x}_l(t)$ respectively. Thus, the following equalities hold.

$$\begin{aligned}(\varphi(\tilde{x}_k(t)),s_t)*(\tilde{x}_l(t),s_l) &= (\varphi(\tilde{x}_k(t))\varphi({}^{s_t}\tilde{x}_l(t)),s_ts_l)\\ &= (\varphi(\tilde{x}_k(t))\varphi(\tilde{x}_l(t)),s_ts_l)\\ &= (\varphi(\tilde{x}_k(t)\tilde{x}_l(t)),s_ts_l)\end{aligned}$$

$$
\begin{aligned}
&= (\varphi(\tilde{x}_l(t)\tilde{x}_k(t)), s_l s_k) \\
&= (\varphi(\tilde{x}_l(t))\varphi(\tilde{x}_k(t)), s_l s_k) \\
&= (\varphi(\tilde{x}_l(t))\varphi({}^{s_l}\tilde{x}_k(t)), s_l s_k) \\
&= (\varphi(\tilde{x}_l(t)), s_l) * (\tilde{x}_k(t), s_k)
\end{aligned}
$$

The result follows. □

Claim 2.2 trivially implies that A' and B' $*$-commute as well.

Now, fix a matrix $m_0 \in \mathrm{GL}_n(\mathbb{F}_q)$ of order $q^n - 1$. Let $C \leq \mathrm{GL}_n(\mathbb{F}_q)$ be the subgroup

$$C = \left\{ \sum \ell_i m_0^{k_i} \mid \ell_i \in \mathbb{F}_q,\ k_i \in \mathbb{Z}^+ \right\}.$$

Note that C is abelian.

3 The Colored Burau Key Agreement Protocol

The Colored Burau Key Agreement Protocol (CBKAP) requires a trusted authority to generate public data. This is done by choosing a uniform $z \in M \rtimes S_n$ and then publishing (descriptions of) the $*$-commuting groups $A = z \circ \langle A' \rangle \circ z^{-1}$ and $B = z \circ \langle B' \rangle \circ z^{-1}$ where $\langle A' \rangle$ and $\langle B' \rangle$ denote the groups generated by of A' and B' respectively.

Claim 3.1. *A and B are $*$-commuting subgroups of $M \rtimes S_n$.*

Proof. Let $z = (m_z, s_z) \in M \rtimes S_n$ be uniformly chosen but otherwise arbitrary, and let $A = z \circ \langle A' \rangle \circ z^{-1}$ and $B = z \circ \langle B' \rangle \circ z^{-1}$ where A' and B' are the subsets of generators of $M \rtimes S_n$ defined in Section 2. Let $(m_a, s_a) \in A$ and $(m_b, s_b) \in B$ be arbitrary. Thus, we can express (m_a, s_a) and (m_b, s_b) as

$$
\begin{aligned}
(m_a, s_a) &= z \circ (y_{a'_1}, \sigma_{a'_1}) \circ (y_{a'_2}, \sigma_{a'_2}) \circ \cdots \circ (y_{a'_u}, \sigma_{a'_u}) \circ z^{-1} \\
(m_b, s_b) &= z \circ (y_{b'_1}, \sigma_{b'_1}) \circ (y_{b'_2}, \sigma_{b'_2}) \circ \cdots \circ (y_{b'_v}, \sigma_{b'_v}) \circ z^{-1},
\end{aligned}
$$

where each $(y_{a'_i}, \sigma_{a'_i})$ and $(y_{b'_j}, \sigma_{b'_j})$ are elements from $A' \cup A^*$ and $B' \cup B^*$, respectively. From the proof of Claim 2.2, we know that each $(y_{a'_i}, \sigma_{a'_i})$ and each $(y_{b'_j}, \sigma_{b'_j})$ commute with each other with respect to the operation induced from $M \rtimes S_n$. Thus, by the use of Lemma 2.1, it follows that

$$
\begin{aligned}
(\varphi(m_a), s_a) * (m_b, s_b) &= ((I_n, e) * (m_a, s_a)) * (m_b, s_b) \\
&= (I_n, e) * ((m_a, s_a) \circ (m_b, s_b)) \\
&= (I_n, e) * ((z \circ (y_{a'_1}, \sigma_{a'_1}) \circ (y_{a'_2}, \sigma_{a'_2}) \circ \cdots \circ (y_{a'_u}, \sigma_{a'_u}) \circ z^{-1}) \\
&\qquad \circ (z \circ (y_{b'_1}, \sigma_{b'_1}) \circ (y_{b'_2}, \sigma_{b'_2}) \circ \cdots \circ (y_{b'_v}, \sigma_{b'_v}) \circ z^{-1})) \\
&= (I_n, e) * (z \circ (y_{a'_1}, \sigma_{a'_1}) \circ (y_{a'_2}, \sigma_{a'_2}) \circ \cdots \circ (y_{a'_u}, \sigma_{a'_u}) \\
&\qquad \circ (y_{b'_1}, \sigma_{b'_1}) \circ (y_{b'_2}, \sigma_{b'_2}) \circ \cdots \circ (y_{b'_v}, \sigma_{b'_v}) \circ z^{-1}) \\
&= (I_n, e) * (z \circ (y_{b'_1}, \sigma_{b'_1}) \circ (y_{b'_2}, \sigma_{b'_2}) \circ \cdots \circ (y_{b'_v}, \sigma_{b'_v}) \\
&\qquad \circ (y_{a'_1}, \sigma_{a'_1}) \circ (y_{a'_2}, \sigma_{a'_2}) \circ \cdots \circ (y_{a'_u}, \sigma_{a'_u}) \circ z^{-1})
\end{aligned}
$$

$$
\begin{aligned}
&= (I_n, e) * ((z \circ (y_{b'_1}, \sigma_{b'_1}) \circ (y_{b'_2}, \sigma_{b'_2}) \circ \cdots \circ (y_{b'_v}, \sigma_{b'_v}) \circ z^{-1}) \\
&\qquad\qquad \circ (z \circ (y_{a'_1}, \sigma_{a'_1}) \circ (y_{a'_2}, \sigma_{a'_2}) \circ \cdots \circ (y_{a'_u}, \sigma_{a'_u}) \circ z^{-1})) \\
&= (I_n, e) * ((m_b, s_b) \circ (m_a, s_a)) \\
&= ((I_n, e) * (m_b, s_b)) * (m_a, s_a)) \\
&= (\varphi(m_b), s_b) * (m_a, s_a)
\end{aligned}
$$

The result follows. □

The security of Algebraic Eraser is based on the hardness of the *generalized simultaneous conjugacy search problem*. This problem is stated as such: G is a group, given $y_1, y_2, ..., y_n \in G$ and $x_1, x_2, ..., x_n \in G$, find ζ such that $y_i = \zeta x_i \zeta^{-1}$ for all i.

According to the Trusted Third Party (TTP) algorithm given in [1], A and B are published through the following steps:

1. The TTP chooses the sets A' and B' described in Section 2.
2. The TTP chooses a secret element $z \in M \rtimes S_n$.
3. Choose words $\{w_1, \ldots, w_\gamma\}$ of bounded length from the set A' and it's inverses. Note that $\{w_1, \ldots, w_\gamma\}$ are all elements in the groups A.
4. Choose words $\{v_1, \ldots, v_\gamma\}$ of bounded length from the set B' and it's inverses. Similarly $\{v_1, \ldots, v_\gamma\}$ are all elements in the group B.
5. For $1 \leq i \leq \gamma$, do the following.
   (a) Calculate the left normal form $z \circ w_i \circ z^{-1}$ and reduce the result modulo the square of the fundamental braid Δ, which is defined as
   $$
   \begin{aligned}
   \Delta = &((x_{n-1}(t), s_{n-1}) \circ (x_{n-2}(t), s_{n-2}) \circ \cdots \circ (x_1(t), s_1)) \\
   &\circ ((x_{n-1}(t), s_{n-1}) \circ (x_{n-2}(t), s_{n-2}) \circ \cdots (x_2(t), s_2)) \circ \cdots \circ (x_{n-1}(t), s_{n-1}).
   \end{aligned}
   $$
   Let w'_i be the result of this modulus.
   (b) Calculate the left normal form $z \circ v_i \circ z^{-1}$ and reduce the result modulo the square of the fundamental braid Δ. Let v'_i be result of this modulus.
6. Publish the sets $\{w'_1, \ldots, w'_\gamma\}$ and $\{v'_1, \ldots, v'_\gamma\}$.

Because any even power of the fundamental braid is a central element of the braid group, reducing the element $z \circ z^{-1}$ by the square of Δ and any odd power of the fundamental braid can be replaced by the fundamental braid itself; set w'_i equal to the resulting braid.

If the private braid element z is known by an attacker, Anshel et al. give an attack [1, Section 6] that recovers the shared secret between Alice and Bob in a negligible amount of time. This is why having a truly secure TTP is of paramount importance.

A party Alice generates her private key by choosing a and a The element (a, g) is chosen arbitrarily from A, but note that it would be beneficial for Alice to choose an element that is composed of a significant amount of elements from the generating set as the second element

g of (a, g) will be made public. Having a long product of elements from the generating set of A helps to defend against brute force attacks. Alice then sets her public key equal to $(c, e) * (a, g) = (c\varphi(a), g) \in \mathrm{GL}_n(\mathbb{F}_q) \times S_n$.

A second party Bob acts symmetrically, though using B rather than A. I.e., Bob chooses a uniform matrix $d = \sum \ell_i' m_0^{\beta_i} \in C$ and a uniform $(b, h) \in B$, and then sets his public key equal to $(d, e) * (b, h) = (d\varphi(b), h) \in \mathrm{GL}_n(\mathbb{F}_q) \times S_n$.

Bob and Alice can now compute a shared key k. Alice, given her private key and Bob's public key, will compute the shared key as

$$k = (c, e) \cdot \big(\big((d, e) * (b, h)\big) * (a, g) \big) .$$

Analogously, Bob can compute k using his private key and Alice's public key as

$$k = (d, e) \cdot \big(\big((c, e) * (a, g)\big) * (b, h) \big) .$$

Claim 3.2. *For all $(a, g) \in A, (b, h) \in B$, and $c, d \in C$,*

$$(c, e) \cdot \big(\big((d, e) * (b, h)\big) * (a, g) \big) = (d, e) \cdot \big(\big((c, e) * (a, g)\big) * (b, h) \big)$$

Proof. Let $(a, g) \in A, (b, h) \in B$, and $c, d \in C$. Consider:

$$\begin{aligned}
(c, e) \cdot \big(\big((d, e) * (b, h)\big) * (a, g) \big) &= (c, e) \cdot ((d\varphi(b), h) * (a, g)) \\
&= (c, e) \cdot (d\varphi(b\,{}^h a), hg) \\
&= (cd\varphi(b\,{}^h a), hg) \\
&= (cd, e) \cdot \big(\varphi(b\,{}^h a), hg\big) \\
&= (cd, e) \cdot ((\varphi(b), h) * (a, g)) \\
&= (dc, e) \cdot ((\varphi(a), g) * (b, h)) \\
&= (dc, e) \cdot (\varphi(a\,{}^g b), gh) \\
&= (dc\varphi(a\,{}^g b), gh) \\
&= (d, e) \cdot (c\varphi(a\,{}^g b), gh) \\
&= (d, e) \cdot ((c\varphi(a), g) * (b, h)) \\
&= (d, e) \cdot \big(\big((c, e) * (a, g)\big) * (b, h) \big)
\end{aligned}$$

By using the $*$-commuting property of the sets A and B as well as the fact that elements of C commute, we see that the shared key computed by the two parties is indeed the same. □

Since the second component of the shared secret is just the product of the symmetric group elements from Alice and Bob's public keys, the symmetric group component of the shared secret is indeed not a secret at all.

4 A Worked Example

Let $q = 11$ and $n = 7$. Set $\tau_i = i$ for $i = 1, \ldots, 7$. Define the two subsets A' and B' as

$$A' = \{(x_1(t), s_1), (x_2(t), s_2)\}$$

$$B' = \{(x_4(t), s_4), (x_5(t), s_5), (x_6(t), s_6)\},$$

and let

$$m_0 = \begin{bmatrix} 0 & 10 & 6 & 2 & 0 & 10 & 7 \\ 5 & 4 & 8 & 3 & 9 & 5 & 10 \\ 4 & 0 & 3 & 4 & 9 & 6 & 1 \\ 10 & 8 & 7 & 4 & 2 & 9 & 10 \\ 4 & 7 & 5 & 4 & 8 & 4 & 3 \\ 3 & 10 & 5 & 5 & 2 & 4 & 7 \\ 8 & 10 & 1 & 9 & 2 & 9 & 0 \end{bmatrix}.$$

(Since the characteristic polynomial of m_0 is nonzero, we assume its order is $11^7 - 1$. We have not yet verified this.)

Say the secret element $z \in M \rtimes S_n$ is chosen as

$$\begin{aligned} z &= (x_1(t), s_1) \circ (x_6(t), s_6) \circ (x_3(t), s_3) \\ &= \left(\begin{bmatrix} -t_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & t_3 & -t_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_6 & -t_6 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (1\ \ 2)(6\ \ 7)(3\ \ 4) \right). \end{aligned}$$

With this choice of z, z^{-1} then becomes

$$\begin{aligned} z^{-1} &= ({}^{s_1 s_6 s_3} x_1(t)^{-1} x_6(t)^{-1} x_3(t)^{-1}, s_1 s_6 s_3) \\ &= \left({}^{s_1 s_6 s_3} \begin{bmatrix} -t_1^{-1} & t_1^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -t_3^{-1} & t_3^{-1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -t_6^{-1} & t_6^{-1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (1\ \ 2)(6\ \ 7)(3\ \ 4) \right) \\ &= \left(\begin{bmatrix} -t_2^{-1} & t_2^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -t_4^{-1} & t_4^{-1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -t_7^{-1} & t_7^{-1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (1\ \ 2)(6\ \ 7)(3\ \ 4) \right) \end{aligned}$$

Then

$$\begin{aligned} A &= z \circ \langle (x_1(t), s_1), (x_2(t), s_2) \rangle \circ z^{-1} \\ B &= z \circ \langle (x_4(t), s_4), (x_5(t), s_5), (x_6(t), s_6) \rangle \circ z^{-1}. \end{aligned}$$

The sets A and B are published by calculating the left normal form of each $z \circ a_i \circ z^{-1}$ and $z \circ b_j \circ z^{-1}$ and then reducing each $z \circ a_i \circ z^{-1}$ and $z \circ b_j \circ z^{-1}$ modulo the square of the fundamental braid.

Alice's public/private keys. Say Alice chooses $c \in C$ as

$$c = m_0 + 2m_0^2 + 3m_0^3 = \begin{bmatrix} 2 & 0 & 8 & 5 & 3 & 6 & 6 \\ 3 & 6 & 1 & 3 & 2 & 6 & 6 \\ 6 & 5 & 4 & 8 & 9 & 8 & 9 \\ 5 & 5 & 4 & 4 & 8 & 4 & 0 \\ 1 & 5 & 4 & 10 & 8 & 1 & 0 \\ 7 & 2 & 10 & 9 & 5 & 4 & 5 \\ 2 & 10 & 5 & 6 & 8 & 2 & 2 \end{bmatrix},$$

and $(a, g) \in A$ as

$$\begin{aligned}
(a, g) &= (z \circ (x_1(t), s_1)) \circ z^{-1} \\
&= \left(\begin{bmatrix} -t_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & t_3 & -t_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_6 & -t_6 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -t_2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (6 \quad 7)(3 \quad 4) \right) \circ z^{-1} \\
&= \left(\begin{bmatrix} t_1 t_2 & -t_1 + 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & t_3 & -t_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_6 & -t_6 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -t_2^{-1} & t_2^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -t_3^{-1} & t_3^{-1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -t_6^{-1} & t_6^{-1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{pmatrix} 1 & 2 \end{pmatrix} \right) \\
&= \left(\begin{bmatrix} -t_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (1 \quad 2) \right).
\end{aligned}$$

Her public key is then

$$(c, e) * (a, g) = (c\varphi(a), g)$$

$$= \left(\begin{bmatrix} 2 & 0 & 8 & 5 & 3 & 6 & 6 \\ 3 & 6 & 1 & 3 & 2 & 6 & 6 \\ 6 & 5 & 4 & 8 & 9 & 8 & 9 \\ 5 & 5 & 4 & 4 & 8 & 4 & 0 \\ 1 & 5 & 4 & 10 & 8 & 1 & 0 \\ 7 & 2 & 10 & 9 & 5 & 4 & 5 \\ 2 & 10 & 5 & 6 & 8 & 2 & 2 \end{bmatrix} \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (1 \quad 2) \right)$$

$$= \left(\begin{bmatrix} 9 & 2 & 8 & 5 & 3 & 6 & 6 \\ 8 & 9 & 1 & 3 & 2 & 6 & 6 \\ 5 & 0 & 4 & 8 & 9 & 8 & 9 \\ 6 & 10 & 4 & 4 & 8 & 4 & 0 \\ 10 & 6 & 4 & 10 & 8 & 1 & 0 \\ 4 & 9 & 10 & 9 & 5 & 4 & 5 \\ 9 & 1 & 5 & 6 & 8 & 2 & 2 \end{bmatrix}, (1 \quad 2) \right).$$

Bob's public/private keys. Say Bob chooses $d \in C$ as

$$d = 4m_0^2 + m_0^5 + 2m_0^6 = \begin{bmatrix} 1 & 9 & 2 & 3 & 0 & 2 & 8 \\ 8 & 7 & 8 & 6 & 7 & 8 & 9 \\ 7 & 4 & 5 & 10 & 1 & 8 & 6 \\ 2 & 9 & 7 & 6 & 4 & 1 & 8 \\ 7 & 3 & 5 & 2 & 3 & 1 & 9 \\ 1 & 8 & 4 & 3 & 9 & 8 & 9 \\ 4 & 0 & 2 & 5 & 4 & 9 & 3 \end{bmatrix},$$

and $(b, h) \in B$ as

$$(b, h) = (z \circ (x_6(t), s_6)) \circ z^{-1}$$

$$= \left(\begin{bmatrix} -t_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & t_3 & -t_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_6 & -t_6 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_7 & -t_7 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (1 \quad 2)(3 \quad 4) \right) \circ z^{-1}$$

$$= \left(\begin{bmatrix} -t_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & t_3 & -t_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_6 - t_6 t_7 & t_6 t_7 & -t_6 + 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -t_1^{-1} & t_1^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -t_3^{-1} & t_3^{-1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -t_7^{-1} & t_7^{-1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (6 \quad 7) \right)$$

$$= \left(\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_6 & -t_6 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (6 \quad 7) \right).$$

His public key is then

$$\begin{aligned}(d,e) * (b,h) &= (d\varphi(b), h) \\ &= \left(\begin{bmatrix} 1 & 9 & 2 & 3 & 0 & 2 & 8 \\ 8 & 7 & 8 & 6 & 7 & 8 & 9 \\ 7 & 4 & 5 & 10 & 1 & 8 & 6 \\ 2 & 9 & 7 & 6 & 4 & 1 & 8 \\ 7 & 3 & 5 & 2 & 3 & 1 & 9 \\ 1 & 8 & 4 & 3 & 9 & 8 & 9 \\ 4 & 0 & 2 & 5 & 4 & 9 & 3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6 & -6 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (6 \quad 7) \right) \\ &= \left(\begin{bmatrix} 1 & 9 & 2 & 3 & 1 & 10 & 8 \\ 8 & 7 & 8 & 6 & 0 & 7 & 9 \\ 7 & 4 & 5 & 10 & 5 & 7 & 6 \\ 2 & 9 & 7 & 6 & 10 & 5 & 8 \\ 7 & 3 & 5 & 2 & 9 & 5 & 9 \\ 1 & 8 & 4 & 3 & 2 & 7 & 9 \\ 4 & 0 & 2 & 5 & 3 & 1 & 3 \end{bmatrix}, (6 \quad 7) \right).\end{aligned}$$

Shared key.

$$\begin{aligned}(c,e) \cdot ((d,e) * (b,h)) * (a,g) &= \left(\begin{bmatrix} 2 & 7 & 9 & 5 & 6 & 10 & 5 \\ 2 & 9 & 5 & 4 & 9 & 9 & 0 \\ 1 & 3 & 3 & 3 & 10 & 9 & 9 \\ 2 & 10 & 0 & 5 & 2 & 3 & 7 \\ 8 & 9 & 0 & 9 & 8 & 5 & 7 \\ 6 & 8 & 7 & 3 & 6 & 4 & 5 \\ 3 & 1 & 5 & 8 & 4 & 2 & 5 \end{bmatrix}, (1 \quad 2)(6 \quad 7) \right) \\ &= (d,e) \cdot ((c,e) * (a,g)) * (b,h).\end{aligned}$$

Alice's Calculation

$$\begin{aligned}&(c,e) \cdot ((d,e) * (b,h)) * (a,g) \\ &\qquad = (c,e) \cdot ((d\varphi(b), h) * (a,g))\end{aligned}$$

$$= (c,e)\cdot\left(\left(\left(\begin{bmatrix}1&9&2&3&1&10&8\\8&7&8&6&0&7&9\\7&4&5&10&5&7&6\\2&9&7&6&10&5&8\\7&3&5&2&9&5&9\\1&8&4&3&2&7&9\\4&0&2&5&3&1&3\end{bmatrix},(6\ \ 7)\right)*(a,g)\right)\right)$$

$$= (c,e)\cdot\left(\left(\begin{bmatrix}0&5&10&0&8&3&5\\1&7&8&10&8&5&5\\7&0&10&7&6&6&7\\5&8&5&7&1&10&5\\7&5&1&7&6&2&10\\10&5&1&1&9&2&5\\8&5&7&2&1&10&0\end{bmatrix}\begin{bmatrix}-1&1&0&0&0&0&0\\0&1&0&0&0&0&0\\0&0&1&0&0&0&0\\0&0&0&1&0&0&0\\0&0&0&0&1&0&0\\0&0&0&0&0&1&0\\0&0&0&0&0&0&1\end{bmatrix},(6\ \ 7)(1\ \ 2)\right)\right)$$

$$= (c,e)\cdot\left(\left(\begin{bmatrix}10&10&2&3&1&10&8\\3&4&8&6&0&7&9\\4&0&5&10&5&7&6\\9&0&7&6&10&5&8\\4&10&5&2&9&5&9\\10&9&4&3&2&7&9\\7&4&2&5&3&1&3\end{bmatrix},(6\ \ 7)(1\ \ 2)\right)\right)$$

$$= \left(\left(\begin{bmatrix}2&7&9&5&6&10&5\\2&9&5&4&9&9&0\\1&3&3&3&10&9&9\\2&10&0&5&2&3&7\\8&9&0&9&8&5&7\\6&8&7&3&6&4&5\\3&1&5&8&4&2&5\end{bmatrix},(6\ \ 7)(1\ \ 2)\right)\right)$$

Bob's Calculation

$$(d,e)\cdot((c,e)*(a,g))*(b,h) = (d,e)\cdot((c\varphi(a),g)*(b,h))$$

$$= (d,e)\cdot\left(\left(\left(\begin{bmatrix}9&2&8&5&3&6&6\\8&9&1&3&2&6&6\\5&0&4&8&9&8&9\\6&10&4&4&8&4&0\\10&6&4&10&8&1&0\\4&9&10&9&5&4&5\\9&1&5&6&8&2&2\end{bmatrix},(1\ \ 2)\right)*(b,h)\right)\right)$$

$$= (d,e)\cdot\left(\begin{bmatrix} 0 & 9 & 0 & 1 & 0 & 9 & 1 \\ 10 & 9 & 0 & 3 & 4 & 1 & 9 \\ 3 & 8 & 3 & 8 & 4 & 0 & 6 \\ 2 & 9 & 1 & 8 & 1 & 4 & 9 \\ 3 & 9 & 1 & 8 & 0 & 8 & 3 \\ 8 & 1 & 1 & 1 & 1 & 8 & 1 \\ 0 & 9 & 6 & 4 & 1 & 4 & 0 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6 & -6 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, (1\quad 2)(6\quad 7)\right)$$

$$= (d,e)\cdot\left(\begin{bmatrix} 9 & 2 & 8 & 5 & 6 & 8 & 6 \\ 8 & 9 & 1 & 3 & 5 & 8 & 6 \\ 5 & 0 & 4 & 8 & 2 & 7 & 9 \\ 6 & 10 & 4 & 4 & 10 & 9 & 0 \\ 10 & 6 & 4 & 10 & 3 & 5 & 0 \\ 4 & 9 & 10 & 9 & 7 & 9 & 5 \\ 9 & 1 & 5 & 6 & 9 & 10 & 2 \end{bmatrix}, (1\quad 2)(6\quad 7)\right)$$

$$= \left(\begin{bmatrix} 1 & 9 & 2 & 3 & 0 & 2 & 8 \\ 8 & 7 & 8 & 6 & 7 & 8 & 9 \\ 7 & 4 & 5 & 10 & 1 & 8 & 6 \\ 2 & 9 & 7 & 6 & 4 & 1 & 8 \\ 7 & 3 & 5 & 2 & 3 & 1 & 9 \\ 1 & 8 & 4 & 3 & 9 & 8 & 9 \\ 4 & 0 & 2 & 5 & 4 & 9 & 3 \end{bmatrix}\begin{bmatrix} 0 & 9 & 0 & 1 & 10 & 1 & 1 \\ 10 & 9 & 0 & 3 & 10 & 5 & 9 \\ 3 & 8 & 3 & 8 & 4 & 0 & 6 \\ 2 & 9 & 1 & 8 & 3 & 9 & 9 \\ 3 & 9 & 1 & 8 & 4 & 7 & 3 \\ 8 & 1 & 1 & 1 & 5 & 7 & 1 \\ 0 & 9 & 6 & 4 & 3 & 9 & 0 \end{bmatrix}, (1\quad 2)(6\quad 7)\right)$$

$$= \left(\begin{bmatrix} 2 & 7 & 9 & 5 & 6 & 10 & 5 \\ 2 & 9 & 5 & 4 & 9 & 9 & 0 \\ 1 & 3 & 3 & 3 & 10 & 9 & 9 \\ 2 & 10 & 0 & 5 & 2 & 3 & 7 \\ 8 & 9 & 0 & 9 & 8 & 5 & 7 \\ 6 & 8 & 7 & 3 & 6 & 4 & 5 \\ 3 & 1 & 5 & 8 & 4 & 2 & 5 \end{bmatrix}, (1\quad 2)(6\quad 7)\right)$$

5 The Ben-Zvi, Blackburn, and Tsaban (BBT) Attack

Assume that Eve, the adversary, sees all public information. With the public information, Eve hopes to produce the Shared Secret between Alice and Bob, which is retrieved directly instead of through discovering each of their respective private keys. Before going through the attack specifically, a few definitions will be needed.

Let H be an arbitrary group of $n \times n$ matrices over $\mathbb{F}_q$. Let $\mathrm{Alg}(H)$ denote the $\mathbb{F}_q$-algebra generated by H; that is, $\mathrm{Alg}(H)$ is the collection of a $\mathbb{F}_q$-linear combinations of elements in H. Furthermore, let $\mathrm{Alg}^*(H)$ denote the set of all invertible matrices in $\mathrm{Alg}(H)$. Note that in the CBKAP, $C = \mathrm{Alg}^*(C)$.

Let $\gamma_1, \gamma_2, \ldots, \gamma_\rho \in C$ be a basis for C, which Eve will need to compute. Let $P \trianglelefteq A$ be the *pure subgroup* of A defined as

$$P = \{(\alpha, e) \in A\}.$$

Thus, $\varphi(P)$ is a subgroup of $\mathrm{GL}_n(\mathbb{F}_q)$.

Eve's main goal is to find $\tilde{c} \in C$, $(\tilde{a}, g) \in M \rtimes S_n$, and

$$\sum_{i=1}^{k} \ell_i \varphi(\alpha_i),$$

where $(\alpha_i, e) \in P$ and $\ell_i \in \mathbb{F}_q$ for all $i \in \{1, \ldots, k\}$, that satisfy

$$(c, e) * (a, g) = \tilde{c} \cdot \left(\sum_{i=1}^{k} \ell_i \varphi(\alpha_i), e \right) * (\tilde{a}, g),$$

where $(c, e) * (a, g)$ is Alice's public key. The reason is that, with these elements in hand, Eve can compute the shared secret as follows:

- First compute the matrix
$$\beta' = \sum_{i=1}^{k} \ell_i \varphi({}^{h}\alpha_i),$$
where h is the element from the symmetric group from Bob's public key.

- Eve can then compute
$$(\tilde{c} d \varphi(b) \beta', h) * (\tilde{a}, g),$$
which is exactly the shared secret between Alice and Bob. (See [2] for the proof of this claim)

The rest of the attack is devoted to finding the desired elements that allow Eve to construct the shared secret.

1. **Find the α_i's:** Eve needs to find the elements (α_i, e)'s from A such that the collection $\{\varphi(\alpha_1), \ldots, \varphi(\alpha_j)\}$ form a basis for $\mathrm{Alg}^*(\varphi(P))$. The authors note that this step can be carried out before the transmission of messages between Alice and Bob take place as this does not rely on their public keys.

   Following the method given in [3], Eve generates an element $(a', g') \in A$ such that the order of $g' \in S_n$ is smaller than n, and then computes $\alpha_1 = (a', g')^r$. Eve repeats this process to find $\alpha_2, \alpha_3, \ldots$, until the dimension of the $\mathbb{F}_q$-linear span of the matrices $\varphi(\alpha_i)$ stops growing, usually when the dimension stops growing after four α_i's are added. Eve then fixes a linearly independent subset of these matrices, and thus we have that the matrices $\varphi(\alpha_1), \ldots, \varphi(\alpha_j)$ are a basis for a subspace V of $\mathrm{Alg}(\varphi(P))$. With high probability, we expect that $V = \mathrm{Alg}(\varphi(P))$, so this is assumed from now on.

2. **Find $\tilde{a}$:** Again using the method found in [3], we find a product of generators of A whose second component is equal to g, and this product will be $(\tilde{a}, g)$. Also, define $\delta \in \mathrm{GL}_n(\mathbb{F}_q)$ by
$$(\delta, e) = (c\varphi(a), g) * (\tilde{a}, g)^{-1}.$$

3. **Find $\tilde{c}$:** Assume that Eve has already found the elements $\gamma_1, \gamma_2, \ldots, \gamma_\rho \in C$ that form a basis for C (recall that $C = \mathrm{Alg}(C)$ by assumption). Eve then finds element $y_1, \ldots, y_\rho \in \mathbb{F}_q$ such that

$$\delta^{-1}(y_1\gamma_1 + y_2\gamma_2 + \cdots + y_\rho\gamma_\rho) \in \mathrm{Alg}(\varphi(P)), \text{ and} \tag{5.1}$$
$$y_1\gamma_1 + y_2\gamma_2 + \cdots + y_\rho\gamma_\rho \in \mathrm{GL}_n(\mathbb{F}_q) \tag{5.2}$$

Let $\tilde{c} = y_1\gamma_1 + y_2\gamma_2 + \cdots + y_\rho\gamma_\rho \in C$. To find the elements $y_1, \ldots, y_\rho \in \mathbb{F}_q$, Eve randomly generates solutions y_i to the equation given in 5.1. Due to the linearity of 5.1, this turns out to be easy. If the solution that satisfies 5.1 also satisfies 5.2, then Eve stops; otherwise, Eve starts the process again. Ben-Zvi et al. show in [2] that the proportion of solutions to 5.1 that satisfy 5.2 is bounded by $1 - n/q$. The element $\tilde{c}$ is used in the calculation of the shared secret.

4. **Everything Else:** Since $\delta^{-1}\tilde{c} \in \mathrm{Alg}(\varphi(P))$, it follows that $\tilde{c}^{-1}\delta \in \mathrm{Alg}(\varphi(P))$ as well. Thus, Eve can calculate coefficients ℓ_i that satisfy

$$\sum_{i=1}^{k} = \ell_i\varphi(\alpha_i) = \tilde{c}^{-1}\delta.$$

Therefore, Eve can calculate Alice's public key as follows.

$$(\delta, e) * (\tilde{a}, g) = ((p, g) * (\tilde{a}, g)^{-1}) * (\tilde{a}, g) = (c\varphi(a), g).$$

The BBT attack has been implemented and recovers the shared secret for a transmission where $n = 16$, $q = 256$, and generating words of A being length 650 in less than 8 hours, using only 64MB of memory and running on a 2GHz core. The attack has yet to be optimized, but the mathematical beauty of the scheme remains!

6 Algebraic Eraser Hash/Hickory Hash Overview

Let $n \geq 7$ be an integer and $q > n$ be a prime (power). Let $t = (t_1, t_2, \ldots, t_n)$ be commutative indeterminates with inverses $t^{-1} = (t_1^{-1}, t_2^{-1}, \ldots, t_n^{-1})$. Define

$$x_1 = \begin{bmatrix} -t_1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n\times n} \text{ and} \tag{6.1}$$

$$x_i = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & 0 & 0 & & \\ & & t_i & -t_i & 1 & & \\ & & 0 & 0 & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n\times n} \text{ for } 2 \leq i \leq n-1. \tag{6.2}$$

That is, x_i (for $i > 1$) is the identity matrix but with the $(i, i-1)$th entry set to t_i, the (i,i)th entry set to $-t_i$, and the $(i, i+1)$th entry set to 1. Each x_i is invertible since

$$x_1^{-1} = \begin{bmatrix} -t_1^{-1} & t_1^{-1} & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n \times n} \text{ and} \tag{6.3}$$

$$x_i^{-1} = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & 0 & 0 & & \\ & & 1 & -t_i^{-1} & t_i^{-1} & & \\ & & 0 & 0 & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{bmatrix} \in (\mathbb{F}_q[t])^{n \times n} \text{ for } 2 \leq i \leq n-1. \tag{6.4}$$

Let $M \leq \mathrm{GL}_n(\mathbb{F}_q[t])$ denote the subgroup generated by $\{x_1, \ldots, x_{n-1}\}$ under matrix multiplication.

Let S_n be the symmetric group on n elements, with the identity denoted by e. Given a polynomial $f(t) \in \mathbb{F}_q[t]$, we define the notion of a permutation $\sigma \in S_n$ acting on $f(t)$ as

$$^{\sigma}f(t) = {}^{\sigma}f(t_1, t_2, \ldots, t_n) = f(t_{\sigma(1)}, t_{\sigma(2)}, \ldots, t_{\sigma(n)}).$$

Similarly, we can define the notion of a permutation $\sigma \in S_n$ acting on a matrix $M(t) \in \mathrm{GL}_n(\mathbb{F}_q[t])$ by simply having σ permute each entry in the matrix $M(t)$. We denote $\sigma \in S_n$ acting on a matrix $M(t) \in \mathrm{GL}_n(\mathbb{F}_q[t])$ as ${}^{\sigma}M(t)$.

Let $\{\tau_1, \tau_2, \ldots, \tau_n\} \subset \mathbb{F}_q$ be such that $\tau_i \neq 0$ for all $1 \leq i \leq n$, herein referred to as the set of t-values. Using the set of t-values, we can evaluate a matrix $M(t) \in \mathrm{GL}(\mathbb{F}_q[t])$ by substituting the value τ_i for the indeterminate t_i; we denote this operation by $M(t) \downarrow_{t-\text{values}}$. To evaluate a permuted matrix ${}^{\sigma}M(t)$, we first permute the indeterminants in $M(t)$ and then evaluate the matrix according to the t-values, i.e.

$${}^{\sigma}M(t) \downarrow_{t-\text{values}} = ({}^{\sigma}M(t)) \downarrow_{t-\text{values}} .$$

The AE Hash function is a modification of the original Algebraic Eraser function, which can be found in CITEHERE. The main function used is the function $\star' : \mathrm{GL}_n(\mathbb{F}_q[t]) \times S_n \to \mathrm{GL}_n(\mathbb{F}_q) \times S_n$, which will require some persistence to define entirely.

The main difference between $\star'$ and the original Algebraic Eraser function is that the t-values are permuted while iterating throughout computation. Let $\sigma \in S_n$ and assume we have a set of t-values $T = \{\tau_1, \tau_2, \ldots, \tau_n\} \subset \mathbb{F}_q$, where for each $\tau_i \in T$, the i indicates that the indeterminant t_i gets replaced by τ_i when we evaluate the t-values. We define σ acting on T as ${}^{\sigma}T = \{\tau'_1, \tau'_2, \ldots, \tau'_n\}$, where $\tau'_i = \tau_{\sigma(i)}$ for each $1 \leq i \leq n$. For a simple example, let $n = 4$, $q = 11$, $\sigma = (1, 3, 4, 2)$, and $T = \{10, 4, 6, 2\}$. The following mappings take place between the indeterminant $t = (t_1, t_2, t_3, t_4)$ and T before and after the permutation, respectively.

$$\begin{array}{lcccccccccc} t: & t_1 & t_2 & t_3 & t_4 & & & t: & t_1 & t_2 & t_3 & t_4 \\ & \downarrow & \downarrow & \downarrow & \downarrow & \Longrightarrow & & & \downarrow & \downarrow & \downarrow & \downarrow . \\ T: & 10 & 4 & 6 & 2 & & & {}^{\sigma}T: & 6 & 4 & 10 & 2 \end{array}$$

Now, for $i \in \{1, 2, \ldots, n-1\}$, let $s_i = (i \;\; i+1) \in S_n$ be the transposition of elements i and $i+1$. (Elements of S_n can be represented either as a product of transpositions or in cyclic notation, as these are polynomially equivalent.) Let $(n_0, s_{i_0}) \in \mathrm{GL}_n(\mathbb{F}_q) \times S_n$, and let $(x_{i_1}^{\epsilon_1}, s_{i_1}), (x_{i_2}^{\epsilon_2}, s_{i_2}), \ldots, (x_{i_k}^{\epsilon_k}, s_{i_k}) \in \mathrm{GL}_n(\mathbb{F}_q[t]), S_n$ where each $x_{i_j}^{\epsilon_j}$ are matrices of the form found in equations 6.1, 6.2, 6.3, or 6.4. The operation $\star'$ is defined as

$$\begin{aligned}(n_0, s_{i_0}) \star' (x_{i_1}^{\epsilon_1}, s_{i_1}) \star' (x_{i_2}^{\epsilon_2}, s_{i_2}) \star' \cdots \star' (x_{i_k}^{\epsilon_k}, s_{i_k}) \\ = \left(n_0 \cdot^{s_{i_0}} x_{i_1}^{\epsilon_1} \downarrow_{T_1} \cdot^{s_{i_0} s_{i_1}} x_{i_2}^{\epsilon_2} \downarrow_{T_2} \cdots^{s_{i_0} s_{i_1} s_{i_2} \cdots s_{i_{k-1}}} x_{i_k}^{\epsilon_k} \downarrow_{T_k}, s_{i_0} s_{i_1} s_{i_2} \cdots s_{i_{k-1}} s_{i_k}\right),\end{aligned}$$

Where $T_1 = \{\tau_1, \tau_2, \ldots, \tau_n\} \subset \mathbb{F}_q$ is our original t-values, and

$$\begin{aligned} T_2 &= {}^{s_{i_0} s_{i_1}} T_1 \\ T_3 &= {}^{s_{i_0} s_{i_1} s_{i_2}} T_1 \\ &\vdots \\ T_k &= {}^{s_{i_0} s_{i_1} s_{i_2} \cdots s_{i_{k-1}}} T_1. \end{aligned}$$

7 AE Hash Protocol

Let S be a string of bits and let λ be a fixed non-zero integer. Through padding S with a sufficient amount of zeroes, assume that λ divides $\mathrm{len}(S)$. This allows us to break S into $D_S = \mathrm{len}(S)/\lambda$ blocks;

$$S = \bigcup_{i=1}^{D_S} \mathrm{Block}(i).$$

Let $v(i)$ denote the integer that $\mathrm{Block}(i)$ represents, so $1 \leq v(i) \leq 2^\lambda - 1$ for all i. The AE Hash function is specified by

$$\{n \in \mathbb{Z}^+, q, \lambda, \{\tau_1, \ldots, \tau_n\} \subset \mathbb{F}_q, \{c_0, c_1, \ldots, c_{2^\lambda - 1}\}, (n_0, \sigma_0) \in \mathrm{GL}_n(\mathbb{F}_q) \times S_n\},$$

where q is a power of 2, $\tau_i \neq 0$ for all $1 \leq i \leq n$, and $\{c_0, c_1, \ldots, c_{2^\lambda - 1}\}$ are braid group elements of B_n that are assumed to generate a free submonoid of B_n. The elements of $\{c_0, c_1, \ldots, c_{2^\lambda - 1}\}$ are published in terms of the Artin generators, where each element c_i is of length approximately $2n$. The Artin generators can thus be expressed as matrix-permutation pairs where $b_i \sim (x_i, s_i)$. The explicit sequence of matrix-permutation pairs that represent each c_i are what we use in the calculation of the final $\star'$ operation.

The input string S is associated to a sequence of the elements $c_{v(1)}, c_{v(2)}, \ldots, c_{v(D_S)}$ after being broken into blocks of length D_S. Let $(CB(c_i), \sigma_i)$ be the word in the pairs (x_i, s_i) that represents the braid element c_i. The hash of the string S, denoted $H_{AE}(S)$, is defined as the matrix portion of the following operation.

$$(n_0, \sigma_0) \star' (CB(c_{v(1)}), \sigma_{v(1)}) \star' (CB(c_{v(2)}), \sigma_{v(2)}) \star' \cdots \star' (CB(c_{v(D_S)}), \sigma_{v(D_S)}).$$

8 Properties of AE Hash

Lemma 8.1. *For all $\sigma \in S_n$ and $M(t) \in GL(\mathbb{F}_q[t])$*

$$^{\sigma}M(t) = M(t)\downarrow_{\sigma T}$$

Proof. Let $t = (t_1, t_2, \ldots, t_n)$ be commutative indeterminates, $\{\tau_1, \tau_2, \ldots, \tau_n\} \subset \mathbb{F}_q$ be such that $\tau_i \neq 0$ for all $1 \leq i \leq n$. Let $\sigma \in S_n$ and $M(t) \in \mathrm{GL}(\mathbb{F}_q[t])$ be arbitrary.
I: Permuting the indeterminates $^{\sigma}M(t)$
By the way this is defined, the indeterminates would be mapped in the following way:

$$\begin{aligned} t_1 &= t_{\sigma(1)} = \tau_{\sigma(1)} \\ t_2 &= t_{\sigma(2)} = \tau_{\sigma(2)} \\ &\vdots \\ t_n &= t_{\sigma(n)} = \tau_{\sigma(n)} \end{aligned}$$

II: Permuting the t-values $M(t)\downarrow_{\sigma T}$
By the way this is defined, the indeterminates would be mapped in the following way:

$$\begin{aligned} t_1 &= \tau_{\sigma(1)} \\ t_2 &= \tau_{\sigma(2)} \\ &\vdots \\ t_n &= \tau_{\sigma(n)} \end{aligned}$$

Therefore, regardless of whether the indeterminates within the matrix itself or the list of t-values is permuted, the two operations result in the indeterminates mapping to the same t-values. This results in the matrices being evaluated with the same values. Thus, they are equivalent operations and hence, $^{\sigma}M(t) = M(t)\downarrow_{\sigma T}$ □

Corollary 8.2. *For all $\sigma \in S_n$ and $M(t) \in GL(\mathbb{F}_q[t])$*

$$^{\sigma^2}M(t) = {}^{\sigma}M(t)\downarrow_{\sigma T}$$

To progress any further, we must define two new operations. The first we will denote as $\circ'$. Given elements $m_1, m_2 \in GL_n(\mathbb{F}_q[t])$ and $s_1, s_2 \in S_n$ then

$$(m_1, s_1) \circ' (m_2, s_2) = (m_1 \, {}^{(s_1)^2}m_2, s_1 s_2)$$

The second, we will denote as ✦. Given elements $n_0 \in GL_n(\mathbb{F}_q), m_1 \in GL_n(\mathbb{F}_q[t])$ and $s_0, s_1 \in S_n$ then

$$(n_0, s_0)\text{✦}(m_1, s_1) = (n_0\varphi(m_1), s_1)$$

Theorem 8.3. *For all* $(m_1, s_1), (m_2, s_2), \cdots, (m_k, s_k) \in (GL_n(\mathbb{F}_q[t]), S_n)$ *and* $(n_0, s_0) \in (GL_n(\mathbb{F}_q), S_n)$

$$(n_0, s_0)\blacklozenge[({}^{s_0}m_1, s_0s_1) \circ' (m_2, s_2) \circ' \cdots \circ' (m_k, s_k)] = (n_0, s_0) *' (m_1, s_1) *' (m_2, s_2) *' \cdots *' (m_k, s_k)$$

Proof.

$$\begin{aligned}
&(n_0, s_0)\blacklozenge[({}^{s_0}m_1, s_0s_1) \circ' (m_2, s_2) \circ' \cdots \circ' (m_k, s_k)] \\
&\quad = (n_0, s_0)\blacklozenge[({}^{s_0}m_1 \ {}^{(s_0s_1)^2}m_2, s_0s_1s_2) \circ' \cdots \circ' (m_k, s_k)] \\
&\quad = (n_0, s_0)\blacklozenge[({}^{s_0}m_1 \ {}^{(s_0s_1)^2}m_2 \ {}^{(s_0s_1s_2)^2}m_3, s_0s_1s_2s_3) \circ' \cdots \circ' (m_k, s_k)] \\
&\quad = (n_0, s_0)\blacklozenge({}^{s_0}m_1 \ {}^{(s_0s_1)^2}m_2 \ {}^{(s_0s_1s_2)^2}m_3 \ldots {}^{(s_0s_1\cdots s_{k-1})^2} m_k, s_0s_1 \cdots s_k) \\
&\quad = (n_0\varphi({}^{s_0}m_1 \ {}^{(s_0s_1)^2}m_2 \ {}^{(s_0s_1s_2)^2}m_3 \ldots {}^{(s_0s_1\cdots s_{k-1})^2} m_k), s_0s_1 \cdots s_k) \\
&\quad = (n_0\varphi({}^{s_0}m_1)\varphi({}^{(s_0s_1)^2}m_2)\varphi({}^{(s_0s_1s_2)^2}m_3) \cdots \varphi({}^{(s_0s_1\cdots s_{k-1})^2}m_k)), s_0s_1 \cdots s_k) \\
&\quad = (n_0 \ {}^{s_0}m_1 \downarrow_T \ {}^{(s_0s_1)^2}m_2 \downarrow_T \ {}^{(s_0s_1s_2)^2}m_3 \downarrow_T \ \ldots {}^{(s_0s_1\cdots s_{k-1})^2} m_k \downarrow_T), s_0s_1 \cdots s_k) \\
&\quad = (n_0 \ {}^{s_0}m_1 \downarrow_T \ {}^{(s_0s_1)}m_2 \downarrow_{(s_0s_1)T} \ {}^{(s_0s_1s_2)}m_3 \downarrow_{(s_0s_1s_2)T} \ \ldots {}^{(s_0s_1\cdots s_{k-1})} m_k \downarrow_{(s_0s_1\cdots s_{k-1})T}), s_0s_1 \cdots s_k) \\
&\quad = (n_0 \ {}^{s_0}m_1 \downarrow_T \ {}^{(s_0s_1)}m_2 \downarrow_{(s_0s_1)T} \ {}^{(s_0s_1s_2)}m_3 \downarrow_{(s_0s_1s_2)T}, s_0s_1s_2s_3) *' \cdots *' (m_k, s_k) \\
&\quad = (n_0 \ {}^{s_0}m_1 \downarrow_T \ {}^{(s_0s_1)}m_2 \downarrow_{(s_0s_1)T}, s_0s_1s_2) *' \cdots *' (m_k, s_k) \\
&\quad = (n_0 \ {}^{s_0}m_1 \downarrow_T, s_0s_1) *' (m_2, s_2) *' \cdots *' (m_k, s_k) \\
&\quad = (n_0, s_0) *' (m_1, s_1) *' (m_2, s_2) *' \cdots *' (m_k, s_k)
\end{aligned}$$

□

9 Malleability

Using the results given in Section 8, we can prove that the AE Hash is malleable for certain inputs. Suppose that we can find a product of the symmetric group elements that are associated to each c_i that is equal to the identity permutation. That is there exists a product

$$(CB(c_{i_1}), \sigma_{i_1}) \circ' (CB(c_{i_2}), \sigma_{i_2}) \circ' \cdots \circ' (CB(c_{i_k}), \sigma_{i_k}) = (CB(Y), \sigma_{i_1}\sigma_{i_2} \cdots \sigma_{i_k})$$

such that $\sigma_{i_1}\sigma_{i_2} \cdots \sigma_{i_k} = e$.

Suppose we have two binary input string S_X and S_Y such that the sequence of elements c_i that are associated to S_X and S_Y are equal to $c_{i_1}, c_{i_2}, \ldots, c_{i_k}$ and $c_{j_1}, c_{j_2}, \ldots, c_{j_{k'}}$, respectively. We do not place any restrictions as to what the product $(CB(c_{j_1}), \sigma_{j_1}) \circ' (CB(c_{j_2}), \sigma_{j_2}) \circ' \cdots \circ' (CB(c_{j_{k'}}), \sigma_{j_{k'}})$ is equal to.

Let $S_X || S_Y$ denote the concatenation of the strings S_X and S_Y. Let $(x_X, s_X), (x_Y, s_Y) \in \{x_1, \ldots, x_{n=1}\} \times \{s_1, \ldots, s_{n-1}\}$ be the matrix-permutation pairs that are equal to the very first matrix-permutation pairs that are used in c_{i_1} and c_{j_1}, respectively. By Theorem 8.3, the following equalities hold.

$$\begin{aligned}
(n_0, \sigma_0)\blacklozenge[({}^{\sigma_0}m_X, \sigma_0s_X) \circ' \cdots \circ' (CB(c_{i_2}), \sigma_{i_2}) \circ' \cdots \circ' (CB(c_{i_k}), \sigma_{i_k}) \\
\circ' (CB(c_{j_1}), \sigma_{j_1}) \circ' (CB(c_{j_2}), \sigma_{j_2}) \circ' \cdots \circ' (CB(c_{j_{k'}}), \sigma_{j_{k'}})]
\end{aligned}$$

$$= (n_0, \sigma_0)\bigstar[({}^{\sigma_0}m_X, \sigma_0 s_X) \cdots ({}^{(\sigma_0\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_{k-1}})^2}CB(c_{i_k}), \sigma_0\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_k})$$
$$\circ' (CB(c_{j_1}), \sigma_{j_1}) \circ' (CB(c_{j_2}), \sigma_{j_2}) \circ' \cdots \circ' (CB(c_{j_{k'}}), \sigma_{j_{k'}})]$$
$$= (n_0, \sigma_0)\bigstar[({}^{\sigma_0}m_X, \sigma_0 s_X) \cdots ({}^{(\sigma_0\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_{k-1}})^2}CB(c_{i_k}), \sigma_0)$$
$$\circ' (CB(c_{j_1}), \sigma_{j_1}) \circ' (CB(c_{j_2}), \sigma_{j_2}) \circ' \cdots \circ' (CB(c_{j_{k'}}), \sigma_{j_{k'}})]$$
$$= (n_0, \sigma_0)\bigstar[({}^{\sigma_0}m_X, \sigma_0 s_X) \cdots ({}^{(\sigma_0\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_{k-1}})^2}CB(c_{i_k}), \sigma_0)$$
$$({}^{\sigma_0^2}m_Y, \sigma_0 s_Y) \cdots ({}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}), \sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'}})]$$
$$= (n_0({}^{\sigma_0}m_X \cdots {}^{(\sigma_0\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_{k-1}})^2}CB(c_{i_k}){}^{\sigma_0^2}m_Y \cdots {}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}))\downarrow_T, \sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'}}).$$

Thus, since $H_{AE}(S_X||S_Y)$ is the matrix portion of this equation, it follows by the homomorphism property that

$$\begin{aligned}
H_{AE}(S_X||S_Y) &= n_0({}^{\sigma_0}m_X \cdots {}^{(\sigma_0\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_{k-1}})^2}CB(c_{i_k}){}^{\sigma_0^2}m_Y \cdots {}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}))\downarrow_T \\
&= n_0({}^{\sigma_0}m_X \cdots {}^{(\sigma_0\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_{k-1}})^2}CB(c_{i_k}))\downarrow_T ({}^{\sigma_0^2}m_Y \cdots {}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}))\downarrow_T \\
&= H_{AE}(S_X)({}^{\sigma_0^2}m_Y \cdots {}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}))\downarrow_T \\
&= H_{AE}(S_X)({}^{\sigma_0^2}m_Y \cdots {}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}))\downarrow_T \\
&= H_{AE}(S_X)({}^{\sigma_0^2}m_Y)\downarrow_T \cdots ({}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}))\downarrow_T \\
&= H_{AE}(S_X)({}^{\sigma_0^2}m_Y)\downarrow_T ({}^{s_0}m_Y^{-1})\downarrow_T n_0^{-1}(n_0({}^{s_0}m_Y \cdots ({}^{(\sigma_0\sigma_{j_1}\sigma_{j_2}\cdots\sigma_{j_{k'-1}})^2}CB(c_{j_{k'}}))\downarrow_T) \\
&= H_{AE}(S_X)({}^{\sigma_0^2}m_Y)\downarrow_T ({}^{s_0}m_Y^{-1})\downarrow_T n_0^{-1}H_{AE}(S_Y).
\end{aligned}$$

This shows that the AE Hash function is partially malleable on certain inputs.

References

[1] I. Anshel, M. Anshel, D. Goldfeld, and S. Lemieux, *Key agreement, the algebraic erasertm, and lightweight cryptography*, Contemporary Mathematics **418** (2007), 1–34.

[2] A. Ben-Zvi, S. R. Blackburn, and B. Tsaban, *A practical cryptanalysis of the algebraic eraser*, 2015. `http://eprint.iacr.org/`.

[3] A. Kalka, M. Teicher, and B. Tsaban, *Short expressions of permutations as products and cryptanalysis of the algebraic eraser*, Advances in Applied Mathematics **49** (2012), no. 1, 57–76.

[4] C. Kassel, O. Dodane, and V. Turaev, *Braid groups*, Graduate Texts in Mathematics, Springer New York, 2008.