Proc. of The Sixth Intl. Conf. On Advances In Computing, Control And Networking - ACCN 2017
Copyright © Institute of Research Engineers and Doctors, USA .All rights reserved.
ISBN: 978-1-63248-117-7 doi: 10.15224/ 978-1-63248-117-7-50

ELGAMAL KEY EXCHANGE USING Triple Decomposition Problem

Vaibhav Gupta[1], Saibal K. Pal[2] and Bhaskar Biswas[1]

***Abstract*—A new key agreement scheme based on elgamal and triple decomposition problem over non commutative platforms is presented. An understanding of the new scheme over braid groups is provided and the strengths of it over an earlier scheme that rely on a similar system based on other decomposition problems are discussed. The new scheme provides better security over earlier schemes on non-commutative platforms by countering the linear algebra and length based attacks.**

***Keywords*—Key exchange, Elgamal, Triple decomposition, Braid, Non-commutative, Linear algebra, cryptography**

I. Introduction

In a key exchange scheme, secrecy of the shared key is the major concern. The security of the Diffie-Hellman key exchange scheme relies on the difficulty of the Diffie-Hellman problem over finite fields. Emergence of index calculus attacks against the discrete logarithm problem, developments in quantum computing, and continuous research in the field have led to search for new cryptosystems that rely on different kind of hard problems.

In 1999 Anshel et al. introduced a key exchange scheme using non-commutative groups which he called the Commutator key exchange protocol which was based on the difficulty of conjugacy search problem [8]. Following this, Ko et al. in 2000 and Cha et al. in 2001 presented some new schemes which were also based over non-commutative platforms [11]. Another key agreement protocol that works in ElGamal setting and relies on solving conjugacy problems on non-commutative groups has been proposed by D. Kahrobaei and B. Khan in 2006[1].

In this paper we propose a system that works over non-commutative platform and is designed to counter the problems and weaknesses of the earlier systems. The conjugacy problem is to decompose a given element *u* in the group, into $a^{-1}ga$ where *g* is known and *a* is unknown. Even though this problem is hard in general, the special requirements for the system to be practical and the linear nature of the relations between the private and public keys allowed some attacks against the scheme [9].

In the new scheme the adversary has to deal with quadratic equations in addition to the linear ones in order to search for a key. One category **of** attacks against decomposition based key exchange schemes use linear representation of braids [4,15,16]. Another category of attacks uses a length based probabilistic method to solve equations [10]. We provide the effectiveness of these attacks on the new scheme in section 6. These polynomial-time algorithms do not seem to apply to the new scheme proposed in this manuscript as is stated in [11] (page 18).

II. Previous Work

The ElGamal key exchange protocol [1] similar to ours based on conjugacy search problem operates as follows. A finite non-abelian group *G* is selected which is believed to have solvable word problem. $S, T < G$ are two subgroups of *G* whose elements satisfy the commutative property i.e. let $s \in S$ and $t \in T$ then $st = ts$. Also, given $a, b \in G$, we say that the conjugate of *a* by *b* is $b^{-1}ab$ and write it as a^b.

The protocol operates between two parties Alice and Bob where Alice wishes to share a session key $x \in G$ with Bob. Bob takes $s \in S$, $b \in G$ and publishes *b* and $c = b^s$ as his public key. Alice selects $t \in T$ and sends

$$E = x\,(\,c^{\,t}\,)$$

to Bob, along with

$$h = b^{\,t}$$

With the help of *h*, Bob calculates $(\,b^{\,t}\,)^s = c^{\,t}$ along with

$$E' = (\,c^{\,t}\,)^{-1}$$

because of which he is able to decrypt and obtain the session key,

$$(\boxed{?}^{(\boxed{?}^{\boxed{?}})})^{\boxed{?}'} = (\boxed{?}^{(\boxed{?}^{\boxed{?}})})^{(\boxed{?}^{\boxed{?}})^{-1}}$$

The security of the scheme rests on the fact, to deduce Bob's private key would require solving the equation $c = b^s$ for *s*, which is known to be computationally hard problem commonly called the conjugacy search problem. However, some recent works in the field [4] show that even if the conjugacy search problem is hard, it is still possible to generate a copy of the session key (equivalent/ pseudo key) w.r.t. the linear representations of braid groups.

Lawrence - Krammer or Burau representations being some of those representations (section 3). In the setting of braid groups even if the problem of computing the session key is hard for the adversary, the equations can be transformed into a different form where the computation of the equivalent session key is possible.

III. Braid Groups and their Representations

In this section, we will provide a brief description on braid groups, their different properties and representations.

1. Department of Computer Science and Engineering, Indian Institute of Technology (BHU), Varanasi, India.

2. SAG Labs, DRDO, New Delhi, India.

Consider *n* parallel strands with i^{th} strand crossing over the $i+1^{th}$ strand, this is a simple example of a Braid.

The set $<B_n,*>$ is referred to as a group for B_n to be a collection of *n* strands with different possible orientations and * to be a binary operation on B_n (in this case catenation).

$$B_n = <\sigma_1 , \sigma_2 , \sigma_3 , ..., \sigma_{n-1}>$$

Just like all the other groups, braid groups satisfy all the properties of a group,

Closed : Catenation of two Braids is a Braid.

Inverse : Inverse of a Braid is the mirror image of that Braid.

Identity : n parallel strands represent an Identity braid.

Artin in [3] proposed a presentation of braids in the form of Artin generators σ where σ_i refers to the $i+1^{th}$ strand crossing over the i^{th} strand with all the other strands unchanged. Any braid can be decomposed into a sequence of artin generators.

Braid group generated by artin generator satisfy two relations:

$$\square_\square\square_\square = \square_\square\square_\square \;, \; / i - j / \geq 2 \qquad (1)$$

$$\square_\square\square_\square\square_\square = \square_\square\square_\square\square_\square \;, \; / i - j / = 1 \qquad (2)$$

Relation (1) is known as the swap relation and (2) is known as the shift relation. Both these relations are helpful in reducing the braids to simpler forms.

There are other forms as well in which braid groups are presented. We define a symmetric group S_n over an *n* element set $I_n = \{ 1, 2, ..., n \}$ along with Ref, the set of reflections in $S_n = \{(i, j)| 1 \leq i < j \leq n\}$. Let $S = \{(i, i+1)| 1\leq i < n\}$ be the subset of Ref. Any braid *S* can be presented in the form of sequence of elementary s_i*'s* , shortest possible presentation is used to define the length function,

$$\square(\square) := \square \; \square \; \{\square\square | \square = \square_1\square_2 ... \square_\square\square \; \square \; \square_\square\square \in \square\}$$

Another presentation on braids is defined with the help of generators $\{ rs / s \in S_n \}$ with relations

$$\square(\square \; \square) = (\square \; \square)(\square \; \square), \square \; \square \; (\square\square \; \square) = \square(\square) + \square(\square)$$

The largest permutation *w* with $w(i) = n + 1 - i$ yields a braid Δ, known as the fundamental braid. In this presentation, a braid *x* is uniquely written as, $x = \Delta^k x'$ where x' lies in $B_n^+ - \Delta B_n^+$ (B_n^+ is positive braid) also referred to as normal form of *x*. Let $x \in B_n^+$ be a braid, the greatest *y* such that $\{ y \in rS_n \mid y \leq n \}$ is called leftmost factor of *x*(LF(*x*)). A sequence of braids ($x_1 , x_2 , ..., x_k$) in $rS_n - \{1\}$ is called the greedy form of *x* if $LF(x_i \, x_{i+1}) = x_i$ for all *i* and $x_1 x_2 ...x_k = x$. The *k* here is known as the charney length.

A. Representation

A representation of a group *G* is a map $P : G \rightarrow GL(n,F)$, where *GL* is a group of $n \times n$ invertible matrices over field *F*. The basic idea behind a representation is that in a representation, we transfer each element of the group *G* to a corresponding matrix where the where the Binary relation w.r.t. the group is changed to matrix multiplication. An important aspect in this theory is whether a proposed representation is faithful. A representation *P* is faithful for *P* to be injective, because of which we sent the group elements to a unique matrix. Sometimes we use the phrase linear representation to denote faithful ones. For the Braid Groups, a popular faithful representation is the Lawrence – Krammer representation.

B. Lawrence - Krammer Representation

We will now discuss the Lawrence – Krammer representation for the braid groups. The representation is defined by $K : B_n \rightarrow GL\ (m, Z\ [t^{\pm 1}][q^{\pm 1}])$ where *q* and *t* are the invertible elements of the commutative ring *R* with basis $\{ x_{i,j} \mid s(i, j) \in Ref \}$. $K(\sigma_k)$ is called the krammer matrix w.r.t. the basis x_{ij} for a braid σ_k . Jung Hee Cheon and Byunheup Jun in [4] talked about the bounds of the entries in krammer matrix.

According to Theorem 1 presented in [4] for a braid *x* having the canonical form $\Delta^k x_1 x_2 ... x_l$ with δ to be the minimal number of Artin generators in *x*, following are the bounds for the entries in *K*(*x*) .

- $k \leq$ degree of $t \leq k+l$.
- $2(n-1) \min(0,k) + (n-2) \leq$ degree of $q \leq 2(n-1) \max(k, k+1) + (n-2)$.
- Each entry in *K*(*x*) is considered to be a polynomial in *t*, *q* and $1 - q$.

Krammer representation has been proven to be faithful for $0<q<1$ [3]. Hence we will consider a krammer matrix

$$K'(x) = K(x)_{q=1/2} \text{ so that } q = 1 - q$$

Now the new bounds are.

- $k \leq$ degree $\in t \leq \max(k, k+l)$.
- Each entry of *K'*(*x*) when represented as a ratio of two integers, the numerator is bounded by $2^{d-2(n-1)k}$ whereas the denominator is bounded by $2^{2(n-1)\max(k,k+l)}$.

C. Inverting the Lawrence - Krammer Representation

To obtain an equivalent key i.e. to break the system, we need to define an algorithm to invert the Lawerence - Krammer representation. Several work has been done to invert the Krammer representation. Recently Arkadius G. Kalka in [5] proposed an algorithm to invert the Krammer representation to obtain a preimage braid $x \in B_n$ in LNF (left normal form). Eonkyung Lee in [6] proposed a way to invert the Krammer representation in $O(|\Delta^{-inf(x)}x|n^6)$ where *inf*(*x*) refers to infimum of *x*. Here we will enlighten on the algorithm proposed by Jung Hee Cheon and Byunheup Jun in [4] to recover a braid from its corresponding image matrix in Lawrence Krammer representation.

Algorithm 1 : Invert Lawrence - Krammer representation.

Input : A matrix $K(x) \in GL_m(t^{\pm 1}, q^{\pm 1})$ where

```
$m=n(n-1)/2$
    Output : A braid $x \in B_n$.
    (a) Compute $K(x') = K(\Delta)^{-dt}K(x)$.
    (b) Perform the basis change from $(v_{ij})_{ij} \rightarrow (x_{ij})_{ij}$.
    (c) For $k=1$ to $l$ do
        i. Take a nonzero element $y \in D_\varphi$ [7] and compute $A=\{c_{ij}|K(x')y$ has a non-zero coefficient at the $ij$ coordinate$\}$.
        ii. Compute the maximal element $\tau_k \in S_n$ such that $L(z) \subset A$ as follows.
            • Find all reflections $\sigma_i = (i, i+1)$ with $1 \leq i < n$ such that $L(\sigma_i) \subset A$
            • Given $\tau\sigma_i$ and $\tau\sigma_j$ , a greater element is selected from $\tau\sigma_i\sigma_j$ or $\tau\sigma_i\sigma_j\sigma_i$
            • Find the maximal element $\tau_k \in S_n$ by the recursive use of the above method
        iii. Compute the positive braid $x_k$ corresponding to $\tau_k$
        iv. Replace $K(x')$ by $K(x_k)^{-1}K(x')$
    (d) Output $x = \Delta^{dt} x_1 x_2 ...x_k$.
```

The complexity of this algorithm is about d_t power of a matrix and n^2 multiplication of permutations, which is dominated by d_t power of a matrix.

IV. Attack on the scheme

The key exchange scheme presented in section 2 has the parameters as $b \in G$, $c=b^s$, $h=b^t$ and□ = □$^{(□^□)}$. To attack the scheme, we should be able to recover c^t from the given public parameters b, c and h to get information about the session key x. However, the above problem is referred to as conjugacy search problem which is computationally a hard problem.

Now, we will propose an attack on the scheme, which will be an extension to the attack on the DHCP presented by Jung Hee Cheon and Byunheup Jun in [4]. We will analyse the scheme for G to be a braid group of n strands(B_n) with S, $T < G$ be two subgroups LB_n and RB_n respectively. Without solving the conjugacy problem in B_n, we solve it in $GL_m(Z[t^{\pm 1}][q^{\pm 1}])$, where $q=1/2$ and $m=n(n-1)/2$ using lawrence - krammer representation for braid groups. Let the images of s, t, b, c and h under this representation(K') be S, T, B, C and H respectively. We consider a matrix S in $GL_m(Z[t])$ satisfying,

$$BS = SC$$

$$SK'(\sigma_i) = K'(\sigma_i)S, \text{ for } n/2 < i < n$$

Let S' be the invertible matrix solution to the above equations, using S', equivalent session key can be computed as follows,

$$key = SHS^{-1}$$
$$key = STBT^{-1}S^{-1}$$
$$key = TSBS^{-1}T^{-1}$$
$$key = TCT^{-1}$$
$$key = K(tsbs^{-1}t^{-1})$$

Here, S is playing the role of a pseudo key because of which it is used in place of s. Now we will propose the algorithm to find the shared key using an equivalent key for ElGamal encryption in matrix representation using Gaussian Elimination and finally inverting the matrix to obtain the actual braid.

```
Algorithm 2 : Find the shared session key.
Input : $b \in B_n(G)$, $s \in Lb_n(S)$, $t \in Rb_n(T)$, $m=(n(n-1))/2$, $E$, a prime $p$ and an irreducible polynomial of degree $d$.
Output : Shared key $x$.
(a) Let $k$ represent the residue field $Z[t] / (p,f(t))$ .
(b) Calculate the images of $b$, $c=sbs^{-1}$ in $Gl_m(k)$ using $K'$.
(c) Formulate a system of $1/8n^4$ linear equations in a total of $1/7n^4$ variables from the equations $K'(b)S=SK'(c)$ and $SK'(\sigma_i) = K'(\sigma_i)S$ for $n/2 < i < n$ over $k$.
(d) Apply Gaussian Elimination to solve for $S$.
(e) If $S$ is non-singular, compute $S^{-1}$ , else go back and calculate a different solution for $S$.
(f) Calculate $K'(h) = tbt^{-1}$ and output $SK'(h)S^{-1} = K'(tsbs^{-1}t^{-1})$ .
(g) Use Algorithm 1 to compute $c^t = tsbs^{-1}t^{-1}$.
(h) Calculate $x$ using□ = □$^{(□^□)^{-1}}$.
```

Gaussian Elimination step takes $1/3m^T$, where $T \sim 2.376$ (theoretically), a finite field of degree d takes d^2 multiplications. When p is sufficiently large, one multiplication takes O($log(p)loglog(p)logloglog(p)$) time using Schonhage and Strassen method. Using these statistics, Jung Hee Cheon and Byunheup Jun in [4] proposed an overall time complexity of $2^{-5}l^2n^{4T}f(d)$ bit operations, where $f(x) = xlog(x)\ loglog(x)$ and d is the word length of $stbt^{-1}s^{-1}$ less than $2ln^2$.

V. The new scheme

In 2014, Y. Peker [2] proposed a new key exchange scheme based on triple decomposition problem which was proven to resist the attacks mentioned above. Hence we propose a non - commutative key exchange in polycyclic groups using a modified ElGamal which uses triple decomposition problem.

Definition(Triple Decomposition Problem) : Let T, S, A be the subsets of a non-commutative group G where $t \in T$, $s \in S$, $a \in A$ & $u \in G$ and

$$u = tas$$

The the triple decomposition is the problem of finding a, t, s given u. Or in other words decomposing an element into a set of elements in which exactly three of them are unknown. Hence, the security of the proposed scheme lies on the triple decomposition problem.

A. *The Protocol*

The scheme requires a monoid G and two series of subsets S & T of G each, $S = \{S_1, S_2, S_3, X_1, X_2\}$ and $T = \{T_1, T_2, T_3, Y_1, Y_2\}$ where the elements of X_1, X_2, Y_1, Y_2 are invertible and $[S_2, Y_1]$, $[S_3, Y_2]$, $[T_1, X_1]$, $[T_2, X_2]$ are pair wise commutative. Please note that in our protocol we define conjugate of a by b to be $b^{-1}ab$ and write it as a^b .

The protocol resides between two parties Alice & Bob. Suppose Alice wishes to share a session key $x \in G$ with Bob. The protocol proceeds as follows,

1. Alice selects $s_1 \in S_1$, $s_2 \in S_2$, $s_3 \in S_3$, $x_1 \in X_1$, $x_2 \in X_2$ and calculates

$$u = s_1x_1$$

$$v = x_1^{-1}s_2x_2$$

$$w = x_2^{-1}s_3$$

2. Bob selects $t_1 \in T_1$, $t_2 \in T_2$, $t_3 \in T_3$, $y_1 \in Y_1$, $y_2 \in Y_2$ and calculates

$$p = t_1y_1$$

$$q = y_1^{-1}t_2y_2$$

$$r = y_2^{-1}t_3$$

3. Alice computes $k_1 = s_1ps_2qs_3r$ and sends$\square = \square^{\square_1}$to Bob.

4. Bob calculates $k_2 = ut_1vt_2wt_3$ and then$\square = (\square)^{\square_2^{-1}}$, thus retrieving the session key.

Proof of correctness:

$$k_1 = s_1ps_2qs_3r$$

$$k_1 = s_1(t_1y_1)s_2(y_1^{-1}t_2y_2)s_3(y_2^{-1}t_3)$$

$$k_1 = s_1t_1s_2t_2s_3t_3$$

$$k_2 = ut_1vt_2wt_3$$

$$k_2 = (s_1x_1)t_1(x_1^{-1}s_2x_2)t_2(x_2^{-1}s_3)t_3$$

$$k_2 = s_1t_1s_2t_2s_3t_3$$

Hence, $k_1 = k_2$, thus

$$(\square)^{\square_2^{-1}} = (\square^{\square_1})^{\square_2^{-1}}$$

$(\square)^{\square_2^{-1}} = \square$

VI. Security Analysis

Apart from some of the trivial choices of the subsets which make the system vulnerable to attacks, the only way to attack the system is to extract the private keys from the public information. Now for this the attacker would have to solve the following equations.

$$a_1x_1 = u \tag{3}$$

$$x_1a_2x_2 = v \tag{4}$$

$$x_2^{-1}a_3 = w \tag{5}$$

Solving (3) and (5) requires to decompose u and w into two elements whereas solving (4) requires to decompose v into three elements. (4) can be considered as a quadratic equation as it can be written as $a_2x_2 = vx_1$ and the equations (3) and (5) are linear. It can be easily observed that solving (5) is trivial when compared to (4). The main difference in this new scheme w.r.t. other non – commutative based schemes is to solve quadratic relations which in the aspect of decomposition is referred to as Triple Decomposition Problem.

However, attacking the system does not specifically mean attacking the triple decomposition problem. The security of the system comes into question even if we are able to retrieve an equivalent key from the public parameters. The security of the scheme would depend on a number of factors, the platform which we use and the choices of subsets. First we will see the choices of subsets which are advised to be avoided and finally we will consider the setting of braid groups as a platform to implement our key exchange.

A. *Cases to be avoided*

We will now discuss about some trivial cases in which the choices of subsets leaves the system vulnerable to attacks.

Case 1 : If $[X_1, Y_1] = 1$, $[X_2, Y_1] = 1$, $[X_2, Y_2] = 1$, along with other invertible and commutative conditions mentioned earlier in the definition of our system. Then $upvqwr$ gives the shared key.

Proof:

$$\begin{aligned} upvqwr = {} & a_1x_1b_1y_1x_1^{-1}a_2x_2y_1^{-1}b_2y_2x_2^{-1}a_3y_2^{-1}b_3 \\ & = a_1b_1x_1y_1x_1^{-1}a_2x_2y_1^{-1}b_2y_2x_2^{-1}a_3y_2^{-1}b_3 \\ & = a_1b_1y_1a_2x_2y_1^{-1}b_2y_2x_2^{-1}a_3y_2^{-1}b_3 \\ & = a_1b_1y_1a_2y_1x_2y_1^{-1}b_2y_2x_2^{-1}a_3y_2^{-1}b_3 \\ & = a_1b_1a_2x_2b_2y_2x_2^{-1}a_3y_2^{-1}b_3 \\ & = a_1b_1a_2b_2x_2y_2x_2^{-1}a_3y_2^{-1}b_3 \\ & = a_1b_1a_2b_2a_3b_3 \end{aligned}$$

Case 2 : If $[A_2, B_1] = 1$, $[A_3, B_2] = 1$, $[A_3, B_1] = 1$, along with other invertible and commutative conditions mentioned earlier in the definition of our system. Then

Proc. of The Sixth Intl. Conf. On Advances In Computing, Control And Networking - ACCN 2017
Copyright © Institute of Research Engineers and Doctors, USA .All rights reserved.
ISBN: 978-1-63248-117-7 doi: 10.15224/ 978-1-63248-117-7-50

uvwpqr gives the shared key.

Proof : $SharedKey = a_1b_1a_2b_2a_3b_3$

$$= a_1a_2a_3b_1b_2b_3$$

$$uvwpqr = a_1x_1x_1^{-1}a_2x_2x_2^{-1}a_3b_1y_1y_1^{-1}b_2y_2y_2^{-1}b_3$$

$$= a_1a_2a_3b_1b_2b_3$$

Hence, SharedKey = *uvwpqr*

Case 3 : To ensure quadratic setting of (2), there should be many solutions to (2) as otherwise the solution to the simultaneous equations (2) and (3) will be unique which will weaken the system against brute force attacks.

Case 4 : If $[A_2, B_1] = 1$ and $[X_2, B_1] = 1$, or $[A_3, B_2] = 1$ and $[A_3, Y_1] = 1$, along with other invertible and commutative conditions mentioned earlier in the definition of our system. Then the security comes down to the complexity of decomposing an element into two.

Proof : When $[A_2, B_1] = 1$, the shared key is $a_1a_2b_1b_2a_3b_3$. Multiplying u, v and p, q we get $uv = a_1a_2x_2$ and $pq = b_1b_2y_2$. Let $a_1a_2 = a$, where $a \in G$. Now considering the commutative conditions $[A_2, B_1] = 1$, $[X_2, B_1] = 1$ & $[A_3, Y_2] = 1$ the shared key can be decomposed into $apqa_3r$. The problem now comes down to decomposing uv into a and x_2 and then finding a_3 from the equation $w = x_2^{-1}a_3$. Hence, in these setting of commutative conditions we were able to determine the shared key by decomposing an element into two which is no longer the Triple Decomposition problem but a simple decomposition problem. Similar proof can be given for the second remark in this case.

Controlled division of generators of a braid group into sets could result in a system which satisfies the above requirements of commutativity conditions. Consider G_n, a braid group of size n where $n - 1 = 3d$ for a positive integer $d \geq 2$.

$A_1 = G_n$

,$X_1 = <\sigma_1, ..., \sigma_{d-1} >$ $B_1 = <\sigma_{d+1}, ..., \sigma_{n-1} >$

$A_2 = <\sigma_1, ..., \sigma_{d-1} >$ $Y_1 = <\sigma_{d+1}, ..., \sigma_{n-1} >$

$X_2 = <\sigma_1, ..., \sigma_{2d-1}>$ $B_2 = <\sigma_{2d+1} , ..., \sigma_{n-1}>$

$A_3 = <\sigma_1, ..., \sigma_{2d-1}>$ $Y_2 = <\sigma_{2d+1} , ..., \sigma_{n-1}>$

$B_3 = n$

It can be easily observed that X_1 is generated by 1st $d-1$ generators and so on. The condition (Case 3) that an equation of the type (the last one) $x_2^{-1}a_3 = w$ should have a large solution space is satisfied by $X_2 = A_3$ as $x_2 = x$, $a = xw$ is a solution for any $x \in X_2$.

The following observations prove that the Triple Decomposition problem is secure against linear algebra attacks.

1. In the setting which uses Triple Decomposition problem, we have a combination of linear and quadratic equations. There is a unique solution which satisfies both the equations. Hence, not all the solutions of the linear equations lead to a valid shared key.

2. We cannot reduce the system to positive braids. Or in other words when we try to reduce the system to positive braids, the corresponding subsets are not preserved in $x_1^{-1}a_2x_2 = v$. Hence, the matrix forms are destroyed.

Conclusion

We proposed a new way to achieve key exchange in a public key domain. The security of the new scheme relies on the triple decomposition problem in a non commutative group. We focused on braid groups as they have the required practical properties needed by the system. We analysed the scheme over a classical ElGamal protocol and presented a setting in which the scheme is resistant to linear algebra and length-based attacks.

Further research is needed to establish a stronger assurity in the scheme and to determine certain parameters for practical uses.

References

[1] *Kahrobaei, D., Khan, B.:A Non-Commutative Generalization of ElGamal Key Exchange using Polycyclic Groups: In Global Telecommunications Conference - GLOBECOM'06 (2006)*

[2] *Peker, Y.:A New Key Agreement Scheme Based on the Triple Decomposition Problem: In International Journal of Network Security. Vol.16, No.6. PP.426-436 (2014)*

[3] *White, J.: On the Linearity of Braid Groups. (2006)*

[4] *Cheon, J., Jun, B.:A polynomial time algorithm for the braid-diffie-hellman conjugacy problem: In Proc. Of Crypto 2003, LNCS 2729. pp. 212–225 (2003)*

[5] *Kalka, Arkadius G.:Improved linear time inversion heuristic for the Burau representation.*

[6] *Lee, E.:Inverting the Burau and Lawrence-Krammer Representations. Contemporary Mathematics, 418, p.153 (2006)*

[7] *Krammer, D.:Braid groups are linear. Annals of Mathematics, pp.131-156 (2002)*

[8] *Anshel, I., Anshel, M., Fisher, B., Goldfeld, D.: New key agreement protocols in braid group cryptography. In Topics in Cryptology—CT-RSA 2001 (pp. 13-27). Springer Berlin Heidelberg (2001)*

[9] *Shpilrain, V., Ushakov, A.:A new key exchange protocol based on the decomposition problem. arXiv preprint math/0512140 (2006)*

[10] *Garber, D., Kaplan, S., Teicher, M., Tsaban, B., Vishne, U.:Probabilistic solutions of equations in the braid group. Advances in Applied Mathematics, 35(3), pp.323-334 (2005)*

[11] *Tsaban, B.:Polynomial time solutions of computational problems in non-commutative algebraic cryptography. In IACR Cryptology ePrint Archive 2012, vol. 615 (2012)*