© de Gruyter 2009
J. Math. Crypt. **3** (2009), 1–12 DOI 10.1515 / JMC.2009.xxx

Linear Diophantine Equation Discrete Log Problem, Matrix Decomposition Problem and the AA $_\beta$-Cryptosystem

M.R.K. Ariffin and N.A. Abu

Communicated by xxx

Abstract. The Linear Diophantine Equation Discrete Log Problem (LDEDLP) is a discrete log problem on the linear Diophantine equation $U = Vx + Wy$. A proper implementation of LDEDLP would render an attacker to search for two private parameters amongst the exponentially many solutions. Embedded within the matrix decomposition problem (MDP) is the LDEDLP. The ability to re-produce the corresponding two square matrices from its product where both of the matrices a private and one of them singular is related to solving the LDEDLP (albeit in a stronger setting when compared to the situation where certain parameters are known). The AA $_\beta$-cryptosystem transmits a two-parameter ciphertext and utilizes only the basic arithmetic operation of multiplication for encryption and decryption. Since the LDEDLP follows a simple mathematical structure, its low computational requirement would enable communication devices with low computing power to deploy secure communication procedures efficiently.

Keywords. linear diophantine equation, linear diophantine equation discrete log problem (LDEDLP), discrete log problem (DLP), matrix decomposition problem (MDP), asymmetric cryptography, secure key exchange, passive adversary attack..

AMS classification. 11T71, 94A60, 11D45.

1 Introduction

The discrete log problem (DLP) and the elliptic curve discrete log problem (ECDLP) has been the source of security for cryptographic schemes such as the Diffie Hellman key exchange procedure, El-Gamal cryptosystem and elliptic curve cryptosystem (ECC) respectively [6], [10]. As for the world renowned RSA cryptosystem, the inability to find the e-th root of the ciphertext C modulo N from the congruence relation $C \equiv M^e(modN)$ coupled with the inability to factor $N = pq$ for large primes p and q is its fundamental source of security [11]. Recently, suggestions have been made that the ECC is able to produce the same level of security as the RSA with shorter key length. Thus, ECC should be the preferred asymmetric cryptosystem when compared to RSA [15]. Hence, the notion “cryptographic efficiency" is conjured. That is, to produce an asymmetric cryptographic scheme that could produce security equivalent to a certain key length of the traditional RSA but utilizing shorter keys. However, in certain situations where a large block needs to be encrypted, RSA is the better option than ECC because ECC would need more computational effort to undergo such

Fundamental Research Grant Scheme #5523934, Ministry of Higher Education, MALAYSIA.

a task [13]. Thus, adding another characteristic toward the notion of “cryptographic efficiency" which is it must be less “computational intensive". As such, in order to design a state-of-the-art public key mechanism, the above two characteristics must be adhered to apart from other well known security issues. In 1998 the cryptographic scheme known as NTRU was proposed with better "cryptographic efficiency" relative to RSA and ECC [9]. Much effort has been done to push NTRU to the forefront [8].

However, this paper would focus only on making comparisons against discrete log based cryptosystems and elliptic curve based cryptosystems. The cryptographic scheme in this paper is based on what is defined as the Linear Diophantine Equation Discrete Log Problem (LDEDLP). The LDEDLP is a new form of discrete log problem based on the linear Diophantine equation of the form $U = Vx + Wy$. The authors propose that the LDEDLP as outlined in this paper is also another discrete log problem that has secure cryptographic qualities coupled with the above described “cryptographic efficiency" qualities. Specifically, the LDEDLP arises within our scheme when one attempts to solve the matrix decomposition problem. On another note, it is not the intention of the authors to go into "provable" security concepts in this initial stage. The immediate objective is to be able to set up a mathematical concept that exhibits one-way characteristic functionality and differs from conventional "one-way" mathematical concepts (i.e. discrete log problem, integer factorization, elliptic curves etc.). The reason? For better "cryptographic efficiency".

The layout of this paper is as follows. In Section 2, the Linear Diophantine Equation Discrete Log Problem (LDEDLP) will be described, a proof that finding the preferred solution by brute force is intractable is given and results relating it to the Diffie Hellman key exchange and the RSA cryptosystem will be produced. The matrix decomposition problem (MDP) is then stated together with its relation with the LDEDLP. In Section 3 the authors describe the AA_{β}-function and the AA_{β}-matrices - an alternative way of evaluating the AA_{β}-function as described by Blackburn [3]. It has to be noted that the idea of designing the AA_{β}-function was motivated by the piecewise function designed by Bose [1],[4]. The mechanism of the AA_{β}-cryptosystem will be detailed in Section 4. Three underlying principles regarding the intractability of the scheme are also listed. In Section 5, cryptanalysis is conducted on the AA_{β}-cryptosystem with the objective to retrieve the private parameters other than trying to solve the LDEDLP. Finally, we conclude the paper by comparing “cryptographic efficiency" characteristics against RSA and ECC schemes in Section 6.

2 Linear Diophantine Equation Discrete Log Problem (LDEDLP)

Let $U, V, W, x, y \in \mathbb{Z}$. The integer solutions for the linear Diophantine equation of the form $U = Vx + Wy$ where U, V and W are known integers and x and y are unknown integers are obtained using the extended Euclidean algorithm. There are infinitely many solutions.

Definition 2.1. Let $U = Vx^* + Wy^*$. We define the pair (x^*, y^*) as the preferred integers used to obtain U. The pair (x^*, y^*) is an element from the set of solutions of $U = Vx + Wy$ which contains infinitely many elements.

Definition 2.2. The linear Diophantine equation given by $U = Vx + Wy$ is defined to be *prf*-solved when (x^*, y^*) are found in order to obtain U. The LDEDLP is solved when $U = Vx + Wy$ is *prf*-solved.

Remark 2.3. If the solution set of an equation is restricted to a finite set or can be limited to a finite number of possibilities the solution set can be found by brute force, that is, by testing each of the possible values.

Lemma 2.4. *The linear Diophantine equation $U = Vx + Wy$ is computationally infeasible to be prf-solved by brute force.*

Proof. We will prove by contradiction. Assume that the linear Diophantine equation is computationally feasible to be *prf*-solved by brute force. Hence, the linear Diophantine equation has a finite set of solutions. This is a contradiction. Thus, the linear Diophantine equation is computationally infeasible to be *prf*-solved by brute force. □

Definition 2.5. From the Diffie Hellman key exchange procedure we define $g^a \equiv A(modp)$ as the DH congruence relation. From the DH congruence relation we define $A = g^a - pt$ where $t \in \mathbb{Z}$ as the DH Diophantine equation. If the Diffie Hellman key exchange procedure utilizes optimum parameters then define the Diffie Hellman key exchange procedure as DH-optimum. Here optimum parameters would mean the utilization of prime p, private exponent key a and primitive root g which are considered best practice values.

Proposition 2.6. *Let the Diffie Hellman key exchange procedure be DH-optimum. The corresponding discrete log problem (DLP) is solved if and only if the DH Diophantine equation is prf-solved.*

Proof. If the DLP is solved then parameter "a" is obtained from the DH congruence relation. Then one can proceed to compute t from the DH Diophantine equation. Thus, the DH Diophantine equation is *prf*-solved. On the other hand it is obvious that if the DH Diophantine equation is *prf*-solved then the DLP is solved. □

Definition 2.7. Let $C \equiv M^e(mod\ N)$ be defined as the RSA congruence relation. From the RSA congruence relation we define $C = M^e - Nt$ where $t \in \mathbb{Z}$ as the RSA Diophantine equation. If the RSA cryptosystem utilizes optimum parameters then define the RSA cryptosystem as RSA-optimum. Here optimum parameters would mean the utilization of large strong primes p and q, encryption exponent e and decryption exponent d which are considered best practice values.

Corollary 2.8. *Let the RSA cryptosystem be RSA-optimum. The $e-$th root of C modulo N is solved if and only if the RSA Diophantine equation is prf-solved.*

Proof. It is similar to Proposition 2.6. □

Proposition 2.9. *If the linear Diophantine equation $U = Vx + Wy$ is computationally feasible to be prf-solved then both the DH Diophantine equation and RSA Diophantine equation are computationally feasible to be prf-solved.*

Proof. Let $U = Vx + Wy$ be an arbitrary linear Diophantine equation. If it were the case for the DH Diophantine equation let $U = A, V = 1, W = -p, x = g^a$ and $y = t$. Since the preferred integers $x^* = g^a$ and $y^* = t$ are computationally feasible to be found then the DH Diophantine equation is computationally feasible to be *prf*-solved. Likewise, in the case of the RSA Diophantine equation let $U = C, V = 1, W = -N, x = M^e$ and $y = t$. Since the preferred integers $x^* = M^e$ and $y^* = t$ are computationally feasible to be found then the RSA Diophantine equation is computationally feasible to be *prf*-solved. □

However, the converse of Proposition 2.9 is not necessary true.

2.1 Matrix Decomposition Problem (MDP)

A matrix is said to be non-invertible matrix if and only if its determinant is zero. Non-invertible matrices are also known as singular matrices. In this work we will be utilizing the 2 X 2 matrix. Given two 2 X 2 matrices **A** and **B** and its product **C** (**AB** $=$ **C**), where the matrices **B** and **C** are known while the matrix **A** is kept secret one can have the following situations:

Situation 1

If the matrix **B** has an inverse $\mathbf{B}^{-1}$, the matrix **A** can be recovered by $\mathbf{A} = \mathbf{C}\mathbf{B}^{-1}$.

Situation 2

If the matrix **B** does not have an inverse, the matrix **A** cannot be recovered by the method implemented in situation 1. This is the matrix decomposition problem (MDP). Now let,

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \mathbf{B} = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}, \mathbf{C} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

One can easily observe that

$$\begin{aligned} c_{11} &= a_{11}b_{11} + a_{12}b_{21} \\ c_{12} &= a_{11}b_{12} + a_{12}b_{22} \\ c_{21} &= a_{21}b_{11} + a_{22}b_{21} \\ c_{22} &= a_{21}b_{12} + a_{22}b_{22} \end{aligned}$$

The above equations are effectively linear Diophantine equations with two unknowns. Suppose one wants to *prf*-solve for c_{11}. By lemma 1, it would be computationally infeasible to *prf*-solve by brute force for c_{11}. The same holds for the other equations. Thus the MDP is reduced to the LDEDLP.

3 The AA_β-Function

We will begin this section with the following definitions.

Definition 3.1. The set of binary strings with a length of k bits is defined by $S^k = \left\{ s = \{b_i\}_{i=0}^{k-1} : b_i \in \{0,1\} \right\}$ where $k \in \mathbb{Z}^+$.

Definition 3.2. Let $\alpha, \beta \in \mathbb{Z}^+$ where $\alpha < \beta$ and both are of length k-bits. The AA_β-function is defined as

$$AA_\beta^s(x_i) = \begin{cases} (\alpha x_{i-1} + x_i)\,, \text{ if } b_i = 0 \\ (x_{i-1} + \beta x_i)\,, \text{ if } b_i = 1 \end{cases}$$

where $i = 0, 1, 2, \ldots, k-1,\ x_{-1} = 0,\ x_0 \in \mathbb{Z}^+$ and $s \in S^k$.

Definition 3.3. Let $s, d_A, d_B \in S^k$. The symbolic representation value is defined as the integer $m = AA_\beta^s(1)$ which is computed via Definition 3.2. We will denote $m_A = AA_\beta^{d_A}(1)$ and $m_B = AA_\beta^{d_B}(1)$ to be the symbolic representation value of Along and Busu respectively. The value of m is bounded below by the minimum value of $m_{min} = AA_\beta^s(1)$ where $s = \{1, 0, 0, \ldots, 0\}$ and bounded above by the maximum value of $m_{max} = AA_\beta^s(1)$ where $s = \{1, 1, 1, \ldots, 1\}$.

The following table depicts the "symbolic representation" of a 4-bit key and its value for selected α and β.

Binary string	Symbolic representation	Symbolic representation value for $\alpha = 5$ and $\beta = 7$
1000	$\alpha + \beta + 2\alpha\beta + \alpha^2$	107
1001	$\alpha + \beta + \alpha\beta + \alpha\beta^2 + \beta^2$	341
1010	$\alpha^2 + 2\alpha\beta + \beta^2 + \beta$	151
1011	$\alpha + \beta + \alpha\beta^2 + \beta^2 + \beta^3$	649
1100	$\alpha + \alpha\beta + \alpha\beta^2 + \beta^2 + 1$	335
1101	$\beta + \alpha\beta^2 + \beta^2 + \beta^3 + 1$	645
1110	$\beta^3 + \alpha\beta^2 + 2\beta + \alpha$	607
1111	$\beta^4 + 3\beta^2 + 1$	2549

Table 1 Symbolic representation value for 4-bit binary string

The following is a lemma from [1] regarding the symbolic representation value.

Lemma 3.4. *Let $s \in S^k$ and AA_β be a function as defined in Definition 3.2. Let $G = x_0 \in \mathbb{Z}^+$ then $AA_\beta^s(G) = AA_\beta^s(1) \cdot G = mG$.*

3.1 The AA_{β}-matrices

In 2010, Blackburn identified another mechanism to construct either the integer m_A or m_B when conducting an attack upon the AA $_{\beta}$-cryptosystem then. Let the integer matrices (to be known as the AA $_{\beta}$-matrices) be identified as follows:

$$\mathbf{A}_0 = \begin{pmatrix} 1 & \alpha \\ 1 & 0 \end{pmatrix}, \mathbf{A}_1 = \begin{pmatrix} \beta & 1 \\ 1 & 0 \end{pmatrix}$$

Correspondent A (Along) and B (Busu) will generate their private strings $d_A, d_B \in S^k$ respectively. Along will compute the integer matrix given by

$$\mathbf{A} = \mathbf{A}_{b_{k-1}}\mathbf{A}_{b_{k-2}} \cdots \mathbf{A}_0$$

where the choice of the matrix $\mathbf{A}_{b_j}$ to be utilized depends on the binary element. If the binary is 0 choose $\mathbf{A}_0$ otherwise choose $\mathbf{A}_1$. The integer m_A is the top left entry of the resulting matrix. Busu will also compute his m_B in the same manner (Busu's resultant integer matrix will be denoted as **B**).

We will now define parameters needed for the renewed AA_{β}-cryptosystem.

Definition 3.5. The private ephemeral base matrix $\mathbf{G}_{A1}$ is a 2 X 2 singular matrix. We will denote the integers within the matrix as follows:

$$\mathbf{G}_{A1} = \begin{pmatrix} gA1_{11} & gA1_{12} \\ gA1_{21} & gA1_{22} \end{pmatrix}$$

Each element will be chosen of "minimum length" (i.e. as long as $\mathbf{G}_{A1}$ is a singular matrix). There will be three base matrices namely $\mathbf{G}_{A1}$,$\mathbf{G}_{A2}$ and $\mathbf{A}_2$.

Definition 3.6. The ephemeral private seed is the k-bit random binary string $d_A \in S^k$.

Definition 3.7. The ephemeral private key is the 2 X 2 matrix **A** and its inverse $\mathbf{A}^{-1}$. For efficient implementation integers within the matrix **A**, are reduced in size by taking only the first $2k$-bits of the resulting integer. We will denote the integers within the matrix as follows:

$$\mathbf{A}_1 = \begin{pmatrix} a1_{11} & a1_{12} \\ a1_{21} & a1_{22} \end{pmatrix}$$

Observe that, differing from the work by Ariffin and Abu in 2009, instead of just utilizing $a1_{11}$ as the private key, we know utilize the whole matrix.

4 The AA_β-Cryptosystem

We begin with the following definitions.

Definition 4.1. Along's public key, are the matrices $\mathbf{E}_{1A}, \mathbf{E}_{2A}, \mathbf{E}_{3A}$ and $\mathbf{E}_{4A}$ defined by:

$$\mathbf{E}_{1A} = \mathbf{G}_{A1}\mathbf{A}_1 \tag{4.1}$$

$$\mathbf{E}_{2A} = \mathbf{G}_{A1}\mathbf{A}_2 \tag{4.2}$$

$$\mathbf{E}_{3A} = \mathbf{G}_{A2}\mathbf{A}_1 \tag{4.3}$$

$$\mathbf{E}_{4A} = \mathbf{A}_2\mathbf{A}_1 \tag{4.4}$$

Observe that the matrices (1)-(4) are all singular. The maximum length of an element in $\mathbf{E}_{1A}$ is $(2k + l_1)$-bits, where l_1 is the length of largest value in $\mathbf{G}_{A1}$, the maximum length in $\mathbf{E}_{2A}$ is (l_1+l_2)-bits, where l_2 is the length of largest value in $\mathbf{A}_2$, the maximum length in $\mathbf{E}_{3A}$ is $(2k + l_3)$-bits, where l_3 is the length of largest value in $\mathbf{G}_{A2}$ and the maximum length in $\mathbf{E}_{4A}$ is $(2k + l_2)$-bits.

Definition 4.2. Along's corresponding private keys are given by:

$$\mathbf{D}_{1A} = \mathbf{A}_1^{-1}\mathbf{A}_2 \tag{4.5}$$

$$\mathbf{D}_{2A} = (\mathbf{E}_{3A} - \mathbf{A}_2\mathbf{A}_2)^{-1} \tag{4.6}$$

The parameters within $\mathbf{D}_{2A}$ must be chosen such that the inverse exists.

Definition 4.3. Suppose Along is sending a plaintext to Busu utilizing Busu's public keys $\mathbf{E}_{1B}, \mathbf{E}_{2B}, \mathbf{E}_{3B}$ and $\mathbf{E}_{4B}$. Let $\mathbf{M}$ be Along's plaintext and $\mathbf{K}_A$ be Along's ephemeral session key. Both $\mathbf{M}$ and $\mathbf{K}_A$ are arbitrary 2 X 2 matrices. Let $\mathbf{C}_1 = \mathbf{K}_A\mathbf{E}_{1B} + \mathbf{M}\mathbf{E}_{4B}$ and $\mathbf{C}_2 = \mathbf{K}_A\mathbf{E}_{2B} + \mathbf{M}\mathbf{E}_{3B}$ be the pair of ciphertexts that Along will relay to Busu.

Proposition 4.4. $[\boldsymbol{C}_2 - (\boldsymbol{C}_1\boldsymbol{D}_{1B})]\boldsymbol{D}_{2B} = \boldsymbol{M}$

Proof. $[\mathbf{C}_2 - (\mathbf{C}_1\mathbf{D}_{1B})]\mathbf{D}_{2B} = \mathbf{M}[\mathbf{E}_{3B} - \mathbf{B}_2\mathbf{B}_2]\mathbf{D}_{2B} = \mathbf{M}$ □

4.1 The AA $_\beta$ - public key cryptography scheme

We will now discuss the AA $_\beta$-cryptosystem. It is as follows: the scenario is that Along will send an encrypted message to Busu. Busu will provide Along with his public key pair $\mathbf{E}_{1B}, \mathbf{E}_{2B}, \mathbf{E}_{3B}$ and $\mathbf{E}_{4B}$. Along will then generate an ephemeral session key $\mathbf{K}_A$ according to Definition 12. Along will proceed to generate $\mathbf{M}$ and then compute $\mathbf{C}_1$ and $\mathbf{C}_2$. Then Along transmits the two-parameter ciphertext $(\mathbf{C}_1, \mathbf{C}_2)$ to Busu. Upon receiving the ciphertext from Along, Busu will proceed to compute $[\mathbf{C}_2-(\mathbf{C}_1\mathbf{D}_{1B})]\mathbf{D}_{2B}$ with his private keys $\mathbf{D}_{1B}$ and $\mathbf{D}_{2B}$ which would result in the plaintext $\mathbf{M}$. The usage of private ephemeral session key by Along is to ensure that the known plaintext attack will not be successful.

Example 4.5. The scenario is that Along will encrypt information to Busu. We will use $k = 8$. The three private base matrices is given by

$$\mathbf{G}_{B1} = \begin{pmatrix} 6 & 8 \\ 3 & 4 \end{pmatrix}$$

$$\mathbf{G}_{B2} = \begin{pmatrix} 22 & 2 \\ 11 & 1 \end{pmatrix}$$

$$\mathbf{B}_2 = \begin{pmatrix} 25 & 2 \\ 50 & 4 \end{pmatrix}$$

.

We will use $\alpha = 131$ and $\beta = 137$. The first ephemeral private seed for Busu is given by the binary string $d_{1B} = [10000001]$ It will generate Busu's ephemeral private key given by

$$\mathbf{B}_1 = \begin{pmatrix} 42993 & 40160 \\ 39335 & 36745 \end{pmatrix}$$

.

The public keys for Busu are

$$\mathbf{E}_{1B} = \begin{pmatrix} 572638 & 534920 \\ 286319 & 267460 \end{pmatrix}$$

,

$$\mathbf{E}_{2B} = \begin{pmatrix} 550 & 44 \\ 275 & 22 \end{pmatrix}$$

,

$$\mathbf{E}_{3B} = \begin{pmatrix} 1024516 & 957010 \\ 512258 & 478505 \end{pmatrix}$$

and

$$\mathbf{E}_{4B} = \begin{pmatrix} 1153495 & 1077490 \\ 2306990 & 2154980 \end{pmatrix}$$

Along will then utilize the ephemeral binary string $d_{KA} = [10001100]$ to generate

$$\mathbf{K}_A = \begin{pmatrix} 63156 & 58861 \\ 63132 & 58838 \end{pmatrix}.$$

Along's plaintext message to be transmitted is the matrix given by

$$\mathbf{M} = \begin{pmatrix} 33013 & 61536 \\ 33542 & 62522 \end{pmatrix}.$$

The matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ are given by

$$\mathbf{C}_1 = \begin{pmatrix} 233061815262 & 217706397230 \\ 235926377608 & 220382210060 \end{pmatrix},$$

$$\mathbf{C}_2 = \begin{pmatrix} 6539557757 & 61043128616 \\ 66442613398 & 62021191274 \end{pmatrix}$$

To simulate decryption by utilizing Busu's private keys is trivial.■

The following sub-sections discuss the LDEDLP within the setting of the AA_β-cryptosystem.

4.2 The AA_β-LDEDLP

Determine the preferred integers from the matrix $\mathbf{A}_1$ given by the tuple $(a1_{11}, a1_{12}, a1_{21}, a1_{22})$ and the preferred integers from the matrix $\mathbf{G}_{A1}$ given by the tuple $(gA1_{11}, gA1_{12}, gA1_{21}, gA1_{22})$ such that $\mathbf{E}_{1A} = \mathbf{G}_{A1}\mathbf{A}_1$.

4.3 The Computational AA_β-LDEDLP, C-AA_β

Given the tuple $(\alpha, \beta, \mathbf{E}_{1A}, \mathbf{E}_{2A}, \mathbf{E}_{3A}, \mathbf{E}_{4A}, \mathbf{C}_1, \mathbf{C}_2)$ where $\mathbf{E}_{1A}$ and $\mathbf{E}_{2A}$ were computed by utilizing random and independent $d_{1A}, d_{2A} \in S^k$; compute $\mathbf{A}_1$ and $\mathbf{A}_2$.

4.4 The Decisional AA_β-LDEDLP, D-AA_β

The probability distributions of the two following tuples are computationally indistinguishable:

- $(\alpha, \beta, \mathbf{E}_{1A}, \mathbf{E}_{2A}, \mathbf{E}_{3A}, \mathbf{E}_{4A}, \mathbf{C}_1, \mathbf{C}_2, \mathbf{A}_1, \mathbf{A}_2)$ where $d_{1A}, d_{2A} \in S^k$ are randomly and independently chosen.
- $(\alpha, \beta, \mathbf{E}_{1A}, \mathbf{E}_{2A}, \mathbf{E}_{3A}, \mathbf{E}_{4A}, \mathbf{C}_1, \mathbf{C}_2, \mathbf{A}_{C1}, \mathbf{A}_{C2})$ where $d_{1A}, d_{2A}, d_{C1}, d_{C2} \in S^k$ are randomly and independently chosen.

The importance of utilizing random and independent elements from S^k is that these elements will produce random and independent preferred integers.

It could also be deduced that if the AA_β -LDEDLP is *prf*-solvable it means that the preferred integers could be obtained. This would imply that the C-AA_β and D-AA_β problems can be solved. Observe that if the AA_β-LDEDLP can be *prf*-solved efficiently by an attacker, he would be able to retrieve the preferred integers given by the tuple $(a1_{11}, a1_{12}, a1_{21}, a1_{22})$ from $\mathbf{E}_{1A}$. He would then be able to compute $\mathbf{A}_1$ and eventually he will also be able to compute $\mathbf{A}_2$. As a consequence, the attacker would be able to distinguish from the tuple $(\mathbf{E}_{1A}, \mathbf{E}_{2A}, \mathbf{E}_{3A}, \mathbf{E}_{4A}, \mathbf{C}_1, \mathbf{C}_2, \mathbf{z}_1, \mathbf{z}_2)$ whether $\mathbf{A}_1 =$

$\mathbf{z}_1$, $\mathbf{A}_2 = \mathbf{z}_2$ or not. However, obtaining the preferred integers does not necessarily imply that the random seeds from S^k could also be obtained.

In parallel with the connection between the discrete log problem, the computational Diffie-Hellman problem (CDH) and the decisional Diffie-Hellman problem (DDH), there also exists such connection between the AA_β-LDEDLP, the C-AA_β and the D-AA_β. The C-AA_β problem is considered a stronger assumption than AA_β-LDEDLP, in the following sense: the ability to compute either $\mathbf{A}_1$ and $\mathbf{A}_2$ without solving the AA_β-LDEDLP. Thus, requiring that the C-AA_β assumption to hold is a more restricting requirement.

The D-AA_β assumption is related to the C-AA_β. If it were possible to efficiently compute $\mathbf{A}_1$ and $\mathbf{A}_2$ from $(\alpha, \beta, \mathbf{E}_{1A}, \mathbf{E}_{2A}, \mathbf{E}_{3A}, \mathbf{E}_{4A}, \mathbf{C}_1, \mathbf{C}_2)$, then one could easily distinguish the two probability distributions above stated in 4.4. Similar to above, D-AA_β is considered a stronger assumption than C-AA_β, in the following sense: the ability to distinguish without actually computing $\mathbf{A}_1$ and $\mathbf{A}_2$. It is believed that C-AA_β is a weaker assumption than D-AA_β: there are instances where detecting D-AA_β tuples is easy, but solving C-AA_β problems is believed to be much harder.

Corollary 4.6. *If the AA_β-LDEDLP is computationally feasible to be prf-solved then both the DH Diophantine equation and RSA Diophantine equation are computationally feasible to be prf-solved.*

Proof. The AA_β-LDEDLP is essentially a system of linear Diophantine equations. The result is a direct implication from Proposition 2.9. □

5 Cryptanalysis - Passive Adversary Attack

As what was illustrated by Blackburn in 2010, the parallel objective of a passive adversary type attack on this renewed AA_β-cryptosystem is as follows: to obtain a "candidate" of $\mathbf{A}_1$ (denoted by $\mathbf{A}_1^{'}$ and $\mathbf{A}_1^{'} \neq \mathbf{A}_1$) such that the candidate is able to decrypt the ciphertext. However, since $\mathbf{A}_1 \neq \mathbf{A}_1^{'}$ we have

$$\mathbf{A}_1^{-1} \neq (\mathbf{A}_1^{-1})^{'} \tag{5.1}$$

Thus, it is clear that unlike its predecessor, only the exact key is useful for the attacker when conducting an attack upon the renewed AA_β-cryptosystem.

6 Conclusion

The AA_β-cryptosystem has the capacity to become a novel public key cryptosystem whose hard mathematical problem is based upon the difficulty of the LDEDLP. The minimum key length for optimum security is yet to be determined. Theoretically, the preferred solution for the LDEDLP is an element from the set of solutions consisting of infinitely many elements. Practically, the preferred solution is actually an element constructed from the tuple $(a_{11}, a_{12}, a_{21}, a_{22})$ of length $2k$-bit each. Hence, the possibility of actually implementing private keys of minimum length 128-bits cannot be

ruled out [12],[14]. On another note, it is known that the implementation of RSA and ECC is $O(n^3)$ operations where n is the length of the message block [5],[8],[16]. By this fact we can have the following tables of comparison.

Algorithm	Speed	Remarks
RSA	$O(n^3)$	See [8]
ECC	$O(n^3)$	See [5] and [16]
AA_β	$O(n^2)$	Utilizes basic arithmetic operation of multiplication.

Table 2 Encryption and decryption speed for message block of length n

Algorithm	Speed	Remarks
RSA	$O(n^3)$	See [8]
ECC	$O(n^3)$	See [5] and [16]
AA_β	$O(n^2)$	Only involves the mathematical operation of multiplication, where n is the size of the two numbers to be multiplied.

Table 3 Generation of public keys

Another advantage is the ability of the cryptosystem to produce encrypt and decrypt keys of any length. One can also note that because encrypt and decrypt procedures are the basic arithmetic operation of addition and subtraction, it could encrypt messages of large block size with ease. As a result this algorithm is advantageous relative to RSA or ECC (because of better speed) and ECC (because of less computational effort to encrypt/decrypt messages of large block size).

References

[1] M. R. K. Ariffin and N. A. Abu, AA_β-cryptosystem: A chaos based public key cryptosystem, *Int. Jour. Cryptology Research*. **vol. 1, no. 2** (2009), 149–163.

[2] M. R. K. Ariffin, N. A. Abu and A. Mandangan, Strengthening the AA_β-cryptosystem, P*roc. Second International Cryptology Conference 2010*. (2010), 16–26.

[3] S. R. Blackburn, The Discrete Log Problem Modulo 1: Cryptanalyzing the Ariffin - Abu cryptosystem, *J. Mathematical Cryptology*, to appear, (2011), EPRINT.IACR.ORG/2010/114.PDF (PrePrint)

[4] R. Bose, Novel Public Key Encryption Techniques Based on Multiple Chaotic Systems, *Physic Review Letters*. **vol. 95, issue 9** (2005), id. 098702.

[5] A. E. Cohen and K. K. Parhi, Implementation of Scalable Elliptic Curve Cryptosystem Crypto-Accelerators for GF(2m), *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers 1*. (2004), 471–477.

[6] W. Diffie and M. E. Hellman, New Directions in Cryptography, *IEEE Transactions on Information Theory*. **vol. 22, no. 26** (1976), 644–654.

[7] J. Hoffstein, J. Pipher and J. H. Silverman, An Introduction to Mathematical Cryptography. *New York: Springer*. (2008), 352–358.

[8] J. Hermans *et. al.*, Speed Records for NTRU, *CT-RSA 2010, LNCS 5985*. (2010), 73–88.

[9] J. Hoffstein, J. Pipher, J. H. Silverman. NTRU: A Ring Based Public Key Cryptosystem in Algorithmic Number Theory (ANTS III) Lecture Notes in Computer Science 1423, *Springer-Verlag, Berlin*. (1998), 267–288.

[10] N. Koblitz, Elliptic Curve Cryptosystems, *Math. Comp*. **vol. 48, no. 177** (1987), 203–209.

[11] R. L. Rivest, A. Shamir and L. Adleman, A method for obtainning digital signatures and public key cryptosystems, *Commun. ACM*. **vol. 21, issue 2** (1978), 120–126.

[12] B. Schneier, Key length in Applied Cryptography. *New York: John wiley & Sons*. (1996), 151–168.

[13] M. Scott, When RSA is better than ECC. (2008, November 15) [Online]. Available: HTTP://WWW.DERKEILER.COM/NEWSGROUPS/SCI.CRYPT/2008-11/MSG00276.HTML

[14] S. S. Wagstaff, Cryptanalysis of Number Theoretic Ciphers, *Divisibility and Arithmetic*. (2003), 27–42.

[15] S. Vanstone, ECC holds key to next generation cryptography. (2006, March 18) [Online].Available: HTTP://WWW.DESIGN-REUSE.COM/ARTICLES/7409/ECC-HOLD-KEY-TO-NEXT-GEN-CRYPTOGRAPHY.HTML

[16] J. Wolkerstorfer and W. Bauer, A PCI-Card for Accelerating Elliptic Curve Cryptography, *Proceedings of Austrochip 2002, Graz, Austria, October 4*, (2002).

Received xxx; revised xxx

Author information

M.R.K. Ariffin, Al-Kindi Cryptography Research Laboratory, Laboratory for Theoretical Studies, Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia.
Email: rezal@math.upm.edu.my

N.A. Abu, Department of Computer Systems and Communications, Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
Email: nura@utem.edu.my