Malaysian Journal of Mathematical Sciences 7(S): 19-37 (2013)
Special Issue: The 3rd International Conference on Cryptology & Computer Security 2012 (CRYPTOLOGY2012)

MALAYSIAN JOURNAL OF MATHEMATICAL SCIENCES
Journal homepage: http://einspem.upm.edu.my/journal

A New Efficient Asymmetric Cryptosystem Based on the Integer Factorization Problem of $N = p^2q$

[1,2*]M. R. K. Ariffin, [1,2]M. A. Asbullah, [1,3]N. A. Abu and [1]Z. Mahad

[1]Al-Kindi Cryptography Research Laboratory,
Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia

[2]Department of Mathematics, Faculty of Science,
Universiti Putra Malaysia,
43400 UPM Serdang, Selangor, Malaysia

[3]Faculty of Information Technology and Communication,
Universiti Teknikal Malaysia, Melaka, Malaysia

E-mail: rezal@putra.upm.edu.my, ma_asyraf@putra.upm.edu.my, nura@utem.edu.my and zaharimahad@putra.upm.edu.my

*Corresponding author

ABSTRACT

In this paper, we introduce a new scheme based on the hardness of factoring integers of the shape $N = p^2q$. Our scheme uses a combination of modular linear and modular squaring. We show that the decryption is 1-to-1 which is a great advantage over Rabin's cryptosystem. Its encryption speed has a complexity order faster than RSA and ECC. For decryption its speed is better than RSA and is marginally behind ECC. Constructed using a simple mathematical structure, it has low computational requirements and would enable communication devices with low computing power to deploy secure communication procedures efficiently.

Keywords: Integer factorization problem, square root problem, asymmetric cryptosystem.

1. INTRODUCTION

The Rabin cryptosystem that utilizes the integer factorization problem of $N = pq$ coupled with the square root modulo problem is said to be an optimal implementation of RSA with the encryption exponent $e = 2$ (Rabin (1979)). However, the situation of a 4-to-1 mapping during decryption has deterred it from being utilized. Mechanisms to ensure its possible implementation have been proposed, however the solutions either still have a possible decryption failure or losing their computational advantages.

As a consequence other underlying cryptographic primitives have taken centre stage. 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 (DHKE) procedure, El-Gamal cryptosystem and elliptic curve cryptosystem (ECC) respectively (Diffie and Hellman (1976), Koblitz (1987)). 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 $C \equiv M^e (mod\, N\,)$ coupled with the inability to factor $N = pq$ for large primes p and q is its fundamental source of security (Rivest *et al.* (1978)).

It has been suggested 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 from RSA (Vanstone (2006)). Hence, the notion "cryptographic efficiency" is conjured. That is, to produce an asymmetric cryptographic scheme that could produce the same security level at a certain key length of the traditional RSA but shorter. 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 a task (Scott (2008)). Thus, it is prudent to have the notion of "cryptographic efficiency" which is less "computationally intensive" and be able to securely transmit large blocks of data (when needed).

In 1998 the cryptographic scheme known as NTRU was proposed with better "cryptographic efficiency" relative to RSA and ECC (Hoffstein *et al.* (2008) and Hermans *et al.* (2010)) NTRU has a textbook complexity order of $O(n^2)$ (Fast Fourier Transform (FFT) allows for $O(n \log n)$) for both encryption and decryption as compared to DHKE, ElGamal, Cramer-Shoup, RSA and ECC (all have a textbook complexity order of $O(n^3)$ or via FFT: $O(n^2 \log n)$).

From our literature review we list the following characteristics that must be "ideally" achieved (but not restricted to):

1. Shorter key length. If possible shorter than ECC 160-bits.

2. Speed. To have speed of complexity order $O(n^2)$ (or FFT implementation of $O(n \log n)$) for both encryption and decryption.
3. Able to increase data size to be transmitted asymmetrically. That is, not to be restricted in size because of the mathematical structure.
4. To be IND-CCA2 secure in the standard model.
5. Simple mathematical structure for easy implementation.

In this paper, we show how to efficiently design an asymmetric cryptosystem based on the hardness of factoring integers of the shape $N = p^2q$ and coupled with the square root problem as one of its cryptographic primitive. That is, we will efficiently redesign Rabin's cryptosytem that has decryption failure due to a 4-to-1 mapping. We will show that in our design for encryption, it does not involve "expensive" mathematical operation. Only basic multiplication is required neither without division nor modulo operation. In order to give a proper fundamental discussion on the merits of this new design, we will define the "Bivariate Function Hard Problem" (BFHP) and give an intuition on its existence via the RSA problem. The hardiness of factoring p^2q has been used in many systems such as the Okamoto-Uchiyama's scheme (Okamoto (1998)) and the Schmidt-Samoa' system (Schmidt (2006)). Also, experimental results on our scheme, RSA and ECC regarding the speed of execution are presented.

The layout of this paper is as follows. Definition of the BFHP and an intuition of its existence via the RSA problem will be presented in Section 2. The AA_β-BFHP will be detailed in Section 3. In this section we will also list previous designs to overcome the decryption failure of the Rabin cryptosystem for comparative purposes on the efficiently to get back the original message during decryption process. We will then proceed to define the AA_β-scheme in Section 4. A numerical example will also be given. Further analysis on the AA_β-BFHP is given in Section 5. These include the Coppersmith type attacks (Coppersmith (1996)), a Euclidean division attack and an analysis of a lattice based attack. Continuing in Section 6, we give a security reduction proof on the underlying hardness assumptions needed in our design. In Section 7 a table of comparison between the AA_β-scheme against RSA, ECC and NTRU is given. Experimental results on the speed and "data payload" between AA_β, RSA and ECC will also be produced. Finally, we shall conclude in Section 8.

2. BIVARIATE FUNCTION HARD PROBLEM

The following proposition gives a proper analytical description of the "Bivariate Function Hard Problem" (BFHP).

Proposition 1. Let $F(x_1, x_2, \dots, x_n)$ be a multivariate one-way function that maps $F : \mathbb{Z}^n \to \mathbb{Z}^+_{(2^{n-1}, 2^n - 1)}$. Let F_1 and F_2 be such functions (either identical or non-identical) such that $A_1 = F_1\,(x_1, x_2, \dots, x_n\,)$, $A_2 = F_2\,(y_1, y_2, \dots, y_n\,)$ and $\gcd(A_1\,, A_2) = 1$. Let $u, v \in \mathbb{Z}^+_{(2^{m-1}, 2^m - 1)}$.
Let

$$G(u, v) = A_1 u + A_2 v \tag{1}$$

If at minimum $m - n - 1 = k$, where 2^k is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, it is infeasible to determine (u, v) over $\mathbb{Z}$ from $G(u, v)$. Furthermore, (u, v) is unique for $G(u, v)$ with high probability.

Remark 1. Before we proceed with the proof, we remark here that the diophantine equation given by $G(u, v)$is solved when the parameters (u, v) over $\mathbb{Z}$ are found. That is, the BFHP is solved when the parameters (u, v) over $\mathbb{Z}$ are found.

Proof. We begin by proving that (u, v) is unique for each $G(u, v)$ with high probability. Assume there exists $u_1 \neq u_2$ and $v_1 \neq v_2$ such that

$$A_1 u_1 + A_2 v_1 = A_1 u_2 + A_2 v_2 \tag{2}$$

We will then have

$$Y = v_1 - v_2 = \frac{A_1(u_1 - u_2)}{A_2}$$

Since $\gcd(A_1\,, A_2) = 1$ and $A_2 \approx 2^n$, then the probability that Y is an integer is 2^{-n}. Then the probability that $v_1 - v_2$ is an integer solution not equal to zero is 2^{-n} . Thus assumption is false with high probability.

Next we proceed to prove that solving the Diophantine equation given by $G(u, v)$ is infeasible to be solved. The general solution for $G(u, v)$ is given by

$$u = u_0 + A_2 t \tag{3}$$

and

$$v = v_0 - A_1 t \tag{4}$$

for some integer t. To find u within the stipulated interval $u \in (2^{m-1}, 2^m - 1)$ we have to find the integer t such that the inequality $2^{m-1} < u < 2^m - 1$ holds. This gives

$$\frac{2^{m-1} - u_0}{A_2} < t < \frac{2^m - 1 - u_0}{A_2}.$$

Then the difference between the upper and the lower bound is

$$\frac{2^m - 1 - 2^{m-1}}{A_2} = \frac{2^{m-1} - 1}{A_2} \approx \frac{2^{m-2}}{2^n} = 2^{m-n-2}.$$

Since $m - n - 1 = k$, where 2^k is exponentially large for any probabilistic polynomial time (PPT) adversary to sieve through all possible answers, we conclude that the difference is very large and finding the correct t is infeasible. This is also the same scenario for v.■

Remark 2. In fact, since the pair (u_0, v_0) inherits the size of G, and from the equation $t = \frac{u-u_0}{A_2}$, we have at minimum the value of $t \approx 2^{m+n-n} = 2^m$.

2.1 Example

Let $A_1 = 191$ and $A_2 = 229$. Let $u = 41234$ and $v = 52167$. Then $G = 19821937$. Here we take $m = 16$ and $n = 8$. We now construct the parametric solution for this BFHP. The initial points are $u_0 = 118931622$ and $v_0 = -99109685$. The parametric general solution are: $u = u_0 + A_2 t$ and $v = v_0 - A_1 t$. There are approximately $286 \approx 2^9$ (i.e. $\frac{2^{16}}{229}$) values of t to try (i.e. difference between upper and lower bound), while at minimum the value is $t \approx 2^{16}$. In fact, the correct value is $t = 519172 \approx 2^{19}$. ■

Remark 3. It has to be noted that the BFHP in the form we have described has to be coupled with other mathematical considerations upon F_1, F_2, u, v to yield practical cryptographic constructions.

2.2 RSA BFHP

Definition 4. Let the tuple (M, e, d, p, q) be strong RSA parameters. Let $N = pq, ed \equiv 1(mod\ \varphi(N))$ and $\varphi(N) = (p - 1)(q - 1)$.
From $C \equiv M^e (mod\ N)$ we rewrite as

$$C(M,j) = M^e - Nj \tag{5}$$

where j is the number of times M^e is reduced by N until $C(M,j)$ is obtained. The problem of determining $(M,j)$from equation (5) is the RSA BFHP. The pair (M,j) is unique with high probability for each $C(M,j)$.

Lemma 1. The RSA BFHP is infeasible to be solved.

Proof. Let $X = M^e$. From

$$C(X,j) = X - Nj \tag{6}$$

the general solution is

$$X = X_0 - Nt$$

and

$$j = j_0 + t$$

for some $t \in \mathbb{Z}$. It is easy to deduce that the correct t belongs in the interval $u \in (2^{k(e-1)-1}, 2^{k(e-1)} - 1)$. Current RSA deployment has $k = 1024$. Hence, to solve the RSA BFHP is infeasible.■

Lemma 2. RSA problem $\equiv_p$ RSA BFHP

Proof. From $C \equiv M^e (mod\ N)$ if the RSA problem is solved then M is found. Hence, $j = \frac{M^e - C}{N}$ is also found. Thus, the RSA BFHP is solved.

From $C(X,j) = X - Nj$, if the RSA BFHP is solved means that (M,j) is found. Thus, the RSA problem is solved.■

Corollary 1. Solving RSA BFHP does not imply successful factoring of $N = pq$.

Proof. From Remark 1, if RSA BFHP is solved then (M,j) is found. That is

$$M = \sqrt[e]{C + Nj}$$

and

$$j = \frac{M^e - C}{N}.$$

It is obvious that the factoring of N is yet to be obtained.■

3. AA_β-BFHP

We now proceed to define parameters that forms the building blocks of the AA_β-BFHP.

Key Generation

INPUT: The size n of the prime numbers.
OUTPUT: A public key tuple (n, A_1, A_2) and a private key tuple (p, q, d).

1. Generate two random and distinct n-bit strong primes p, q such that $p, q \equiv 3 \ (mod\ 4)$ where $2^n < p, q < 2^{n+1}$.
2. Choose random d such that $d > (p^2q)^{\frac{4}{9}}$.
3. Compute integer e such that $ed \ \equiv 1(mod\ pq)$ and add multiples of pq until $2^{3n+4} \ < \ e \ < \ 2^{3n+6}$ (if necessary).
4. Set $A_1 \ = \ p^2q$. We have $2^{3n} \ < \ A_1 \ < \ 2^{3n+3}$.
5. Set $A_2 \ = e$.
6. Return the public key tuple (n, A_1, A_2) and a private key pair (pq, d).

We also have the fact that $2^{2n} \ < pq \ < \ 2^{2n+2}$. We also let $2^{4n} \ < \ u \ < \ 2^{4n+1}$ and $2^{2n-2} \ < \ v \ < \ 2^{2n-1}$.

Lemma 3. Let $C(u, v) \ = \ A_1u \ + \ A_2v^2$ as stated in the above key generation process. This equation has a unique solution set (u, v).

Proof. Suppose that there are two couples of solutions (u_1, v_1) and (u_2, v_2) of the equation $C = \ A_1u \ + \ A_2v^2$ with $(v_1 \neq v_2)$ and $v_i \ < \ 2^{2n} - 1$. Then $A_1u_1 \ + \ A_2{v_1}^2 \ = A_1u_2 \ + \ A_2{v_2}^2$. Using $A_1 = p^2q$ this leads to

$$(u_2 - \ u_1)p^2q = \ (v_1 \ + \ v_2)(v_1 - \ v_2)A_2.$$

Since $\gcd(p^2q;\ A_2) \ = \ 1$, then $p^2q| \ (v_1 \ + \ v_2)(v_1 - \ v_2)$ and the prime numbers p and q satisfy one of the conditions

$$\begin{cases} p^2|(v_1 \pm v_2) \\ q|(v_1 \mp v_2) \end{cases} \text{or} \ \begin{cases} pq\,|(v_1 \pm v_2) \\ p\,|(v_1 \mp v_2) \end{cases} \text{ or } \ p^2q|(v_1 \ \pm \ v_2)$$

Observe that p^2, pq and $p^2q \ > \ 2^{2n}$ while $|(v_1 \ \pm \ v_2)| < \ 2 \cdot 2^{2n-1} \ = \ 2^{2n}$: This implies that none of these conditions is possible. Hence the equation $C \ = \ A_1u \ + \ A_2v^2$ has only one solution with the defined parameters.■

Lemma 4. Let $C(u,v) = A_1u + A_2v^2$ as stated in the above key generation process. To determine the solution set (u,v) is a BFHP.

Proof. The proof is a direct implementation of Proposition 1.

We now proceed to give a proof of correctness. We will begin by computing $W \equiv Cd \equiv v^2 \, (mod\, pq)$: Then we have to solve $W \equiv v^2 \, (mod\, pq)$ using the Chinese Remainder Theorem.

Lemma 5. Let p and q be two distinct primes such that $p, q \equiv 3 \, (mod\, 4)$. Define x_p and x_q by

$$x_p \equiv W^{\frac{p+1}{4}} \, (mod\, p), x_q \equiv W^{\frac{q+1}{4}} \, (mod\, q).$$

Then the solutions of the equation $x^2 \equiv W \, (mod\, p)$ are $\pm x_p \, (mod\, p)$ and the solutions of the equation $x^2 \equiv W \, (mod\, q)$ are $\pm x_q \, (mod\, q)$.

Lemma 6. Let p and q be two distinct primes such that $p, q \equiv 3 \, (mod\, 4)$. Define x_p and x_q by

$$x_p \equiv W^{\frac{p+1}{4}} \, (mod\, p), x_q \equiv W^{\frac{q+1}{4}} \, (mod\, q).$$

Define $M_1 \equiv q^{-1} \, (mod\, p)$ and $M_2 \equiv p^{-1} \, (mod\, q)$. Then the solutions of the equation $v^2 \equiv W \, (mod\, pq)$ are

$$v_1 \equiv x_pM_1q + x_qM_2p \, (mod\, pq),$$

$$v_2 \equiv x_pM_1q - x_qM_2p \, (mod\, pq),$$

$$v_3 \equiv -x_pM_1q + x_qM_2p \, (mod\, pq),$$

$$v_4 \equiv -x_pM_1q - x_qM_2p \, (mod\, pq).$$

Proof. To solve the equation $v^2 \equiv W \, (mod\, pq)$, we use the Chinese Remainder Theorem. Consider the equations ${x_p}^2 \equiv W \, (mod\, p)$ and ${x_q}^2 \equiv W \, (mod\, q)$. Then the solution of the equation $v^2 \equiv W \, (mod\, pq)$, are the four solutions of the four systems

$$\begin{cases} v \equiv \pm x_p \, (mod\, p) \\ v \equiv \pm x_q \, (mod\, q) \end{cases}$$

Define $M_1 \equiv q^{-1}\ (mod\ p)$ and $M_2 \equiv p^{-1}\ (mod\ q)$. We will get explicitly

$$v_1 \equiv x_p M_1 q + x_q M_2 p\ (mod\ pq),$$

$$v_2 \equiv x_p M_1 q - x_q M_2 p\ (mod\ pq),$$

$$v_3 \equiv -x_p M_1 q + x_q M_2 p\ (mod\ pq),$$

$$v_4 \equiv -x_p M_1 q - x_q M_2 p\ (mod\ pq).$$

It can be seen that solving $v^2 \equiv W\ (mod\ pq)$, will give four solutions v_i for $i = 1, 2, 3, 4$ and as mentioned in Lemma 3 there is only one integer value v_i for its corresponding u. Thus, there is no decryption failure.

Remark 3. The Rabin cryptosystem is known to have decryption failure due to its 4-to-1 mapping. The following is a list that describes strategies to overcome this feature of the Rabin cryptosystem.

1. Redundancy in the message (Menezes *et al.* (1996)). This scheme has a probability decryption failure of approximately $\frac{1}{2^{l-1}}$where l is the least significant binary string of the message.

2. Extra bits (Kurosawa *et al.* (2001)). One will send 2 extra bits of information to specify the square root. The encryption process requires the computation of the Legendre and Jacobi symbol. This results in a computational overhead which is much more than just computing a single square modulo N.

3. Williams's technique (Williams (1980)). The encryption process requires the encrypter to compute a Jacobi symbol. Hence, losing the performance advantage of Rabin over RSA (as in point no.2).

Remark 4. The AA_β-BFHP is that upon obtaining $C = A_1u + A_2v^2$ determine the pair (u, v). Combining Lemma 3 and 6, it is clear that the AA_β-BFHP provides a platform for designing a scheme which employs the square root problem but with no decryption failure.

4. THE AA_β ENCRYPTION SCHEME

We assume that the communication is between party A (Along) and party B (Busu). Busu encrypts to Along.

4.1 Encryption

INPUT: The public key tuple (n, A_1, A_2) and the message M.
OUTPUT: The ciphertext C.

1. Message is an integer $M = 2^{4n}m_1 + m_2$ where we have the following condition for the pair (m_1, m_2): $2^{4n} < m_1 < 2^{4n+1}$ and $2^{2n-2} < m_2 < 2^{2n-1}$.
2. Compute $C = A_1 m_1 + A_2 {m_2}^2$
3. Send ciphertext C to Along.

4.2 Decryption

Decryption by Along is conducted in the following steps:

INPUT: The private key (pq, d) and the ciphertext C.
OUTPUT: The plaintext M.

1. Compute $W \equiv C_1 d (mod\ pq)$.
2. Proceed to solve W as in Lemma 6 to obtain a list m_{2_i} for $i = 1,2,3,4$.
3. For $i = 1,2,3,4$ compute $m_{1_i} = \frac{C - {m_{2_i}}^2 A_2}{A_1}$.
4. Sort the pair (m_{1_j}, m_{2_j}) for integer m_{1_j}.
5. Return the message $M = 2^{4n}m_1 + m_2$.

4.3 Example

Let $n = 16$. Along will choose the primes $p = 106243$ and $q = 79151$. The public keys will be $A_1 = 893422852703399$ and $A_2 = 11179696420225111$. The private keys will be $pq = 8409239693$ and $d = 7674272266$. The message Busu sends is formed by the integers $m_1 = 34209071375236753507$ and $m_2 = 896788005$. Consequently we have ${m_2}^2 = 804228725911880025$.

Then $C_1 = 39554199144517456173395858868378068$. To decrypt Along will first compute $W = 4493909651$. Along will then obtain the following root values $m_{2_1} = 7512451688, m_{2_2} = 6327417266$, $m_{2_3} = 2081822427$ and $m_{2_4} = 896788005$. Only $m_{1_4} = \frac{C - {m_{2_4}}^2 A_2}{A_1}$. will produce an integer value. That is $m_{1_4} = 34209071375236753507$.

5. FURTHER ANALYSIS ON THE AA_β-BFHP

5.1 Coppersmith type attack

Theorem 1. Let N be an integer of unknown factorization. Furthermore, let $f_N(x)$ be an univariate, monic polynomial of degree δ. Then we can find all solutions x_0 for the equation $f_N(x) \equiv 0(mod\ N)$ with

$$|x_0| < N^{\frac{1}{\delta}}$$

in time polynomial in $(log\ N, \delta\ )$.

Theorem 2. Let N be an integer of unknown factorization, which has a divisor $b > N^{\beta}$. Furthermore let $f_b(x)$ be an univariate, monic polynimial of degree δ. Then we can find all solutions x_0 for the equation $f_b(x) \equiv 0(mod\ b)$ with

$$|x_0| < \frac{1}{2} N^{\frac{\beta^2}{\delta} - \epsilon}$$

in time polynomial in $(logN, \delta, \frac{1}{\epsilon})$.

Claim 1: Attacking v

With reference to Theorem 1, let $N = A_1 = p^2q$ and $d' \equiv e^{-1}\ (mod\ N)$. Compute $W \equiv Cd' \equiv v^2(mod\ pq)$. Let $f_N(x) \equiv x^2 - W \equiv 0(mod\ N)$. Hence $\delta = 2$. Thus the root $x_0 = v$ can be recovered if $v < N^{\frac{1}{2}} \approx 2^{1.5n}$. But since $v \approx 2^{2n}$, this attack is infeasible.

Claim 2: Attacking d

With reference to Theorem 2, we begin by observing $f_b(x) \equiv ex - 1 \equiv 0(mod\ pq)$ where pq is an unknown factor $N = A_1 = p^2q$.
Since $pq > N^{\frac{2}{3}}$ we have $\beta = \frac{2}{3}$. From $f_b(x)$ we also have $\delta = 1$. By Theorem 2, the root $x_0 = d$ can be found if $|x_0| < N^{\frac{4}{9}}$. But since $d > N^{\frac{4}{9}}$, this attack is infeasible.

5.2 Euclidean division attack

From $C = A_1u + A_2v^2$, we observe

1. $\left\lfloor \frac{C}{A_1} \right\rfloor \neq u$
2. $\left\lfloor \frac{C}{A_1} \right\rfloor \neq v^2$

5.3 Analysis on lattice based attack

The square lattice attack has been an efficient and effective means of attack upon schemes that are designed based on Diophantine equations. The AA_β scheme has gone through analysis regarding lattice attacks while it went through the design process. Let $C = A_1u + A_2v^2$, be an AA_β-BFHP equation. Consider the Diophantine equation $A_1x_1 + A_2x_2 = C$. Introduce the unknown x_3 and consider the Diophantine equation

$$A_1x_1 + A_2x_2 - Cx_3 = 0.$$

Then $(u, v^2, 1)$ is a solution of the equation. Next let T be a number to be fixed later. Consider the lattice $\mathcal{L}$ spanned by the matrix:

$$\overline{M_0} = \begin{pmatrix} 1 & 0 & A_1T \\ 0 & 1 & A_2T \\ 0 & 0 & -CT \end{pmatrix}$$

Observe that

$$(x_1, x_2, x_3)\overline{M_0} = (x_1, x_2, T(A_1x_1 + A_2x_2 - Cx_3)).$$

This shows that the lattice $\mathcal{L}$ contains the vectors $(x_1, x_2, T(A_1x_1 +$ *$A_2x2-Cx3)$* and more precisely the vector-solution *$V0=(u, v^2, 0)$*. Observe that the length of V_0 satisfies

$$\| V_0 \| = \sqrt{u^2 + v^4} \approx 2^{4n}$$

On the other hand, the determinant of the lattice is $det(L) = CT$ and the Gaussian heuristics for the lattice $\mathcal{L}$ asserts that the length of its shortest non-zero vector is usually approximately $\sigma(\mathcal{L})$ where

$$\sigma(L) = \sqrt{\frac{dim(\mathcal{L})}{2\pi e}} det(\mathcal{L})^{\frac{1}{dim(\mathcal{L})}} = \sqrt{\frac{3}{2\pi e}} (CT)^{\frac{1}{3}}.$$

If we choose T such that $\sigma(\mathcal{L}) > \| V_0 \|$, then V_0 can be among the short non-zero vectors of the lattice $\mathcal{L}$. To this end, T should satisfy

$$T > (\frac{\pi e}{2})^{\frac{3}{2}} \cdot \frac{2^{12n}}{C} \qquad (7)$$

Next, if we apply the LLL algorithm to the lattice $\mathcal{L}$, we will find a basis (b_1, b_2, b_3) such that $\parallel b_1 \parallel \leq \parallel b_2 \parallel \leq \parallel b_3 \parallel$ and

$$b_i \leq 2^{\frac{n(n-1)}{4(n+1-i)}} det(\mathcal{L})^{\frac{1}{(n+1-i)}}, \text{ for } i = 1,2,3,4 \text{ and } n = 3$$

For $i = 1$, we choose T such that $\parallel V_0 \parallel \leq \parallel b_1 \parallel \leq 2^{\frac{1}{2}} \cdot (CT)^{\frac{1}{3}}$. Using the approximation$\parallel V_0 \parallel \approx 2^{4n}$, this is satisfied if

$$T > 2^{-\frac{1}{2}} \cdot \frac{2^{12n}}{C},$$

which follows from the lower bound of equation (7). We experimented this result to try to find $(u, v^2, 0)$. The LLL algorithm outputs a basis with a matrix in the form

$$\overline{M_1} = \begin{pmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & T \end{pmatrix}$$

If $(u, v^2, 0)$ is a short vector, then $(u, v^2, 0) = (x_1, x_2, x_3)\overline{M_1}$ for some short vector (x_1, x_2, x_3). We deduce the system

$$\begin{cases} a_{11}x_1 + a_{21}x_2 = u \\ a_{12}x_1 + a_{22}x_2 = v^2 \end{cases}$$

from which we can deduce that $x_3 = 0$. If we compute $\frac{A_1u - A_2v^2}{C}$, we get $x_2 = 1$ for some x_1. It follows that

$$\begin{cases} a_{11}x_1 + a_{21} = u \\ a_{12}x_1 + a_{22} = v^2 \end{cases}$$

This situation is similar to the AA_β-BFHP. In fact it is the general solution for the Diophantine equation for C, where $a_{11} = A_2$ and $a_{12} = A_1$.

5.4 Example with lattice based attack

We will use the parameters in the earlier example. Observe the lattice $\mathcal{L}$ spanned by the matrix

$$\overline{M_0} = \begin{pmatrix} 1 & 0 & A_1T \\ 0 & 1 & A_2T \\ 0 & 0 & -CT \end{pmatrix}$$

the length of the vector $V = (u, v^2, 0)$ is approximately $\parallel V \parallel \approx 35751905917344588937$. We will use $T = 2^{20n}$ which would result in the length of the vector V is shorter than the Gaussian heuristic of the lattice $\mathcal{L}$.

The LLL algorithm outputs the matrix $\overline{M_1}$ given by:

$$\begin{pmatrix} 11179696420225111 & -893422852703399 & 0 \\ 278692739853541622 & 3515767083866695990 & 0 \\ 213708068523785 & -17078430849056 & -T \end{pmatrix}$$

It follows that

$$\begin{cases} 11179696420225111x_1 + 278692739853541622 = u \\ -893422852703399x_1 + 3515767083866695990 = v^2 \end{cases}$$

which is exactly the general solution for Diophantine equation of C in example Section 4.3.

6. SECURITY REDUCTION AND COMPUTATIONAL HARD PROBLEM OF THE AA_β-BFHP

The following propositions describe the security reduction and computational hard problems enveloping the AA_β-BFHP.

Proposition 2. Solving AA_β-BFHP $\leq_p$ Factoring $A_1 = p^2q$

Proof. It is obvious that if one is able to factor $A_1 = p^2q$ then d can be computed, where $ed \equiv 1 (mod\ pq)$. One then proceeds to solve $Cd \equiv v^2 (mod\ pq)$. Since the solution set (u, v) is unique, only one choice of v from v_i where $i = 1, 2, 3, 4$ will provide an integer solution for $u = \frac{C - {v_i}^2 A_2}{A_1}$. ■

7. TABLE OF COMPARISON

The following is a table of comparison between RSA, ECC, NTRU and AA_β. Let $|K|$ denote public key size. The AA_β cryptosystem has the ability to encrypt large data sets (i.e. $8n$-bits of data per transmission). The ratio of $M{:}\,|K|$ suggests better economical value per public key bit being used. The table also states the complexity order for encryption and decryption via Fast Fourier Transform (FFT). We denote n as the minimum security parameter of each algorithm.

TABLE 1: Comparison table for input block of length n

Algorithm	Encryption Speed	Decryption Speed	Security parameter n	Ratio $M{:}C$	Ratio $M{:}\lvert K\rvert$
RSA	$O(n^2 log\, n)$	$O(n^2 log\, n)$	512	$n{:}n$	$2n{:}3n$
ECC	$O(n^2 log\, n)$	$O(n^2 log\, n)$	160	$n{:}2n$	$n{:}n$
NTRU	$O(n\, log\, n)$	$O(n\, log\, n)$	2008	Varies	$n{:}6n$
AA_β	$O(n\, log\, n)$	$O(n^2 log\, n)$	512	$8n{:}7n$	$8n{:}6n$

7.1 Empirical evidence

We now produce empirical results between RSA, ECC and AA_β algorithms, which should give a non-analytical perspective of the speed each algorithm is capable of. Experiment was conducted on the following environment: Maple 13 on Windows XP Professional, Core 2 Duo, P8400 @ 2.26 GHz and 956 MB RAM.

TABLE 2: RSA encryption and decryption time (in seconds)

Key Size n(n-bits)	Encryption Speed	Decryption Speed	No. of message blocks $(24576/\,n)$
1024	0.544	0.781	24
2048	0.836	1.394	12
4096	1.753	3.362	6

TABLE 3: ECC encryption and decryption time (in seconds)

Key Size (n-bits)	Encryption Speed	Decryption Speed	No. of message blocks $(24576/\,n)$
160	0.416	0.854	154
224	0.422	0.927	110
320	0.436	0.932	101

TABLE 4: AA_β encryption and decryption time (in seconds)

Length of prime (n-bits)	Key Size ($6n$-bits)	Encryption Speed	Decryption Speed	No. of message blocks $(24576/\,n)$
512	3072	0.142	0.489	8
1024	6144	0.143	0.752	4
2048	12288	0.144	1.578	2

8. CONCLUSION

Through the presentation of this work we have examined the square root problem that had difficulties to be executed under the circumstances of a 4-to-1 decryption scenario. An in depth dissection within the RSA problem lead us to the RSA-BFHP. We generalized the RSA-BFHP into the general BFHP and utilized its one-way property. The tightness of the BFHP definition and subsequent analytical results provided an avenue to construct and argue on the effectiveness of the methodology presented in this work. Furthermore, one cannot discount the many other possible designs based on the general BFHP statement.

Extending the results through complexity order analysis as well as "real" computational experiments, it could be seen that with an encryption and decryption speed of $O(n\,log\,n)$ for encryption and $O(n^2\,log\,n)$ for decryption, AA_β is able to provide an ideal platform for applications that rely on fast bulk encryption for the masses while at the same time has a relaxed environment for decryption. Linearly, a 1024-bit security parameter would allow AA_β encrypting 618,696,503 bits of data in 1 hour. On the other hand RSA would need 3.8 hours for the same amount of data.

In concluding, this result provides an avenue for more efficient designs to be produced. The BFHP could easily be extended to a multivariate situation. The opportunity to obtain a scheme that has both encryption and decryption having complexity order of $O(n^2)$ (or best case implementation of $O(n\,log\,n)$) and key length much shorter than the one prescribed for the integer factorization problem is there to be discovered.

ACKNOWLEDGEMENTS

The authors would like to thank the Ministry of Higher Education, Malaysia for research funding. The authors also would like to thank Prof. Dr. Abderrahmane Nitaj from the Département de Mathématiques, Université de Caen, France, Dr. Yanbin Pan of Key Laboratory of Mathematics Mechanization Academy of Mathematics and Systems Science, Chinese Academy of Sciences Beijing, China and Dr. Gu Chunsheng of School of Computer Engineering, Jiangsu Teachers University of Technology, Jiangsu Province, China for valuable comments and discussion on all prior AA_β designs.

REFERENCES

Coppersmith, D. 1996. Finding a Small Root of a Univariate Modular Equation, *Proceedings of Eurocrypt '96, Lecture Notes in Computer Science*. **1070**:155-165.

Coppersmith, D. 1996. Finding a Small Root of a Bivariate Modular Equation; Factoring with High Bits Known, *Proceedings of Eurocrypt '96, Lecture Notes in Computer Science*. **1070**:178-189.

Diffie, W. and Hellman, M.E. 1976. New Directions in Cryptography, *Proceedings of* IEEE Transactions on Information Theory. **22**(6): 644–654.

Galbraith, S.D. 2012. *Mathematics of Public Key Cryptography*. Cambridge University Press.

Hoffstein, J., Lieman, D., Pipher, J. and Silverman, J.H. 2008. *NTRU : A Public Key Cryptosystem*. Accessed 27th February 2013. Sourced from: http://grouper.ieee.org/groups/1363/lattPK/submissions/ntru.pdf

Hoffstein, J., Pipher, J. and Silverman, J.H. 2008. *An Introduction to Mathematical Cryptography*. New York: Springer.

Hoffstein, J., Pipher, J. and Silverman, J.H. 1998. NTRU: A Ring Based Public Key Cryptosystem in Algorithmic Number Theory. *Lecture Notes in Computer Science*. **1423**: 267–288.

Hermans, J., Vercauteren, F. and Preneel, B. 2010. Speed Records for NTRU. *Topics in Cryptography - CT-RSA 2010, Lecture Notes in Computer Science*. **5985**: 73–88.

Koblitz, N. 1987. Elliptic Curve Cryptosystems. *Mathematics of Computation*. **48**: 203–209.

Kurosawa, K., Ogata, W., Matsuo, T. and Makishima, S. 2001. IND-CCA Public Key Schemes Equivalent to Factoring $n = pq$. *Public Key Cryptography 2001*: 36-47.

Menezes, A.J., van Oorschot, P.C. and Vanstone, S.A. 1996. *Handbook of Applied Cryptography*, CRC Press.

Okamoto, T. and Uchiyama, S. 1998. A New Public-Key Cryptosystem as Secure as Factoring. *EUROCRYPT-98, Lecture Notes in Computer Science*. **1403**: 308-318.

Rabin, M.O. 1979. Digitalized signatures and public-key functions as intractable as factorization. *Technical Report MIT/LCS/TR-212, MIT Laboratory for Computer Science*.

Rackoff, C. and Simon, D.R. 1992. Non-Interactive Zero-Knowledge Proof of Knowledge and Chosen Ciphertext Attack. *CRYPTO 1991, Lecture Notes in Computer Science*. **576**: 433-444.

Rivest, R. L., Shamir, A. and Adleman, L. 1978. A method for obtainning digital signatures and public key cryptosystems. *Communication of the ACM*. **21**:120–126.

Schmidt-Samoa, K. 2006. A New Rabin-type Trapdoor Permutation Equivalent to Factoring. *Electronic Notes in Theoretical Computer Science (ENTCS)*. **157**(3): 79-94.

Schneier, B. 1996. *Key length in Applied Cryptography*. New York: Wiley & Sons.

Scott, M. 2008. *When RSA is better than ECC*. Accessed 17th January 2013. Sourced from http://www.derkeiler.com/ Newsgroups/ sci. crypt/2008-11/msg00276.html.

Vanstone, S. 2006. ECC holds key to next generation cryptography. Accessed 17th January 2013. Sourced from http://www.design-reuse.com/articles/7409/ecc-hold-key-to-next-gen-cryptography.html.

Wagstaff, S. S. 2003. *Cryptanalysis of Number Theoretic Ciphers*. Chapman & Hall.

Williams, H. C. 1980. A Modification Of The RSA Public Key Encryption Procedure. *IEEE Trans. Inf. Theory*. **26**(6): 726-729.