# A new security notion for asymmetric encryption Draft #10

Muhammad Rezal Kamel Ariffin[1,2]

[1] Al-Kindi Cryptography Research Laboratory,
Institute for Mathematical Research,
[2] Department of Mathematics, Faculty of Science,
Universiti Putra Malaysia (UPM), Selangor, Malaysia
rezal@upm.edu.my

**Abstract.** A new practical asymmetric design is produced with desirable characteristics especially for environments with low memory, computing power and power source.

KEYWORDS: asymmetric security, diophantine equations, bivariate function hard problem

## 1 Introduction

In this article we provide a new asymmetric encryption design based on the difficulty of solving ***solving a diophantine equation with infinitely many solutions*** and ***solving a system of diophantine equations with unknown exponent***. Further discussion on this problem will be provided in the following sections.

## 2 A new security notion for asymmetric encryption

The following 2 sub-sections provide definitions and discussion on the the so-called ***underlying security primitive*** which the our asymmetric scheme relies on.

### 2.1 Linear diophantine equations with infinitely many solutions

**Definition 1.** *To determine the preferred solution for a diophantine equation where that preferred solution is from a set of infinitely many solutions.*

To further understand and obtain the intuition of Definition 1, we will now observe a remark by Herrmann and May [1]. It discusses the ability to retrieve variables from a given linear Diophantine equation. But before that we will put forward a famous theorem of Minkowski that relates the length of the shortest vector in a lattice to the determinant[1]:

**Theorem 1.** *In an $\omega$-dimensional lattice, there exists a non-zero vector $v$ with*

$$||v|| \leq \sqrt{\omega}\det(L)^{\frac{1}{\omega}}$$

We now put forward the remark.

*Remark 1.* There is a method for finding small roots of linear modular equations $a_1x_1 + a_2x_2 + ... + a_nx_n \equiv 0 \pmod{N}$ with known modulus $N$. It is further assumed that $\gcd(a_i, N) = 1$. Let $X_i$ be upper bound on $|x_i|$. The approach to solve the linear modular equation requires to solve a shortest vector problem in a certain lattice. We assume that there is only one linear independent vector that fulfills the Minkowski bound (Theorem 1) for the shortest vector. Herrmann and May showed that under this heuristic assumption that the shortest vector yields the unique solution $(y_1, ..., y_n)$ whenever

$$\prod_{i=1}^{n} X_i \leq N.$$

If in turn we have

$$\prod_{i=1}^{n} X_i > N^{1+\epsilon}.$$

then the linear equation usually has $N^{\epsilon}$ many solutions, which is exponential in the bit-size of $N$. So there is no hope to find efficient algorithms that in general improve on this bound, since one cannot even output all roots in polynomial time.

We now put forward a corollary.

**Corollary 1.** *A linear diophantine equation $f(x_1, x_2, ..., x_n) = a_1x_1 + a_2x_2 + ... + a_nx_n = N$, with*

$$\prod_{i=1}^{n} x_i > N^{1+\epsilon}.$$

*is able to ensure secrecy of the preferred sequence $\mathbf{x} = \{x_i\}$.*

*Remark 2.* In fact if one were to try to solve the linear diophantine equation $N = a_1x_1 + a_2x_2 + ... + a_nx_n$, where $\prod_{i=1}^{n} X_i > N^{1+\epsilon}$, any method will first output a short vector $\mathbf{x} = \{x_i\}$ as the initial solution. Then there will be infinitely many values from this initial condition that is able to recontruct $N$.

### 2.2 System of diophantine equations with unknown exponent(s) and reduction moduli

It is well known that from:

$$A \equiv g^a \pmod{p}$$

if given the tuple $(A, g, p)$ to determine the unknown exponent $a$ (if the tuple are "strong") would be difficult. In fact this is the discrete log problem (DLP).

We now extend this feature to the following setting; given:

$$A_i \equiv \sum_{j=1}^{k} g_j^{a_j} \pmod{p}$$

If given the tuple $(A_i)$, determine $(a_j, g_j, p)$.

## 3 Bivariate Function Hard Problem (BFHP)

In this section we introduce a particular case of a linear diophantine equation in 2 variables that is able to secure its private parameters under some conditions. This section explores subsection 2.1 in more detail for the mentioned case.

**Definition 2.** *We define $\mathbb{Z}^+_{(2^{m-1},2^m-1)}$ as a set of positive integers in the interval $(2^{m-1}, 2^m - 1)$. In other words, if $x \in (2^{m-1}, 2^m - 1)$, $x$ is an $m$-bit positive integer.*

**Proposition 1.** *Let $A = f(x_1, x_2, ..., x_n)$ be a one-way function that maps $f : Z^n \to Z^+_{(2^{m-1},2^m-1)}$. Let $f_1$ and $f_2$ be such function (either identical or non-identical) such that $A_1 = f(x_1, x_2, ..., x_n), A_2 = f(y_1, y_2, ..., y_n)$ and $gcd(A_1, A_2) = 1$. Let $u, v \in Z^+_{(2^{n-1},2^n-1)}$. Let $(A_1, A_2)$ be public parameters and $(u, v)$ be private parameters. Let*

$$G(u, v) = A_1u + A_2v \tag{1}$$

*with the domain of the function $G$ is $Z^2_{(2^{n-1},2^n-1)}$ since the pair of positive integers $(u, v) \in Z^2_{(2^{n-1},2^n-1)}$ and $Z^+_{(2^{m+n-1},2^{m+n}-1)}$ is the codomain of $G$ since $A_1u + A_2v \in Z^+_{(2^{m+n-1},2^{m+n}-1)}$.*

*If at minimum $n - m - 1 = k$, 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.*

Before we proceed with the proof of the above proposition we would like to put forward 2 remarks.

*Remark 3.* We remark that the preferred pair $(u, v) \in \mathbb{Z}$, is the $prf$-solution for (1). The preferred pair $(u, v)$ is one of the possible solutions for (1) from:

$$u = u_0 + A_2t \tag{2}$$

and

$$v = v_0 - A_1t \tag{3}$$

for any $t \in \mathbb{Z}$.

*Remark 4.* Before we proceed with the proof, we remark here that the diophantine equation given by $G(u,v)$ is solved when the preferred parameters $(u,v) \in \mathbb{Z}$ are found. That is the BFHP is $prf$-solved when the preferred parameters $(u,v) \in \mathbb{Z}$ are found.

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

$$A_1u_1 + A_2v_1 \neq A_1u_2 + A_2v_2 \tag{4}$$

We will then have

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

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 $v_1 = v_2$ with probability $1 - 2^{-n}$.

We next proceed to prove that to $prf$-solve the diophantine equation given by (1) is infeasible. The general solution for $G(u,v)$ is given by (2) and (3) for some integer $t$.

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

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

Then, the difference between the upper and the lower bound is $\approx \frac{2^{n-2}}{2^m}$.

Since $n - m - 1 = k$ 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 $v$.

*Example 1.* 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: $u = u_0 + A_2t$ and $v = v_0 - A_1t$. 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 $t = 519172 \approx 2^{19}$.

## 4 A new asymmetric primitive

In this section we provide the reader with a working cryptographic primitive that is based upon the BFHP.

• **Key Generation by Along**

INPUT: The size $n$ of the parameters.
OUTPUT: A public key tuple $(n, e_1, e_2, e_3)$ and private keys $(d_1, d_2, d_3, p)$.

1. Generate private random $n$-bit prime,$p$.
2. Generate $e$ where $gcd(e, p-1) = 1$. For reasons to be observed later the value of $e$ is with reference to the amount of data the user intends to relay.
3. Compute private $d_1 \equiv e^{-1} \pmod{p-1}$.
4. Generate secret random $n$-bit $g_1, u_1, u_2, h_2, h_3 \in \mathbb{Z}_p$.
5. Compute secret $h_3 = u_1 - h_1$ and $h_4 = u_2 - h_2$.
6. Compute secret $d_{u2} \equiv u_2^{-1} \pmod{p-1}$.
7. Compute secret $g_2 \equiv g_1^{u_1 d_{u2}} \pmod{p}$.
8. Compute public values $e_2 \equiv a g_2^{h_2} \pmod{p}$ and $e_3 \equiv g_1^{h_3} \pmod{p}$.
9. Compute private $d_3 \equiv e_2^{-1} \pmod{p}$.
10. Compute private $k = e_3 d_3 \pmod{p}$.
11. Compute secret $a \equiv k(g_1^{h_1} g_2^{h_2} - g_1^{h_3} g_2^{h_4})^{-1} \pmod{p}$.
12. Compute public values $e_1 \equiv a g_1^{h_1} \pmod{p}$.
13. Compute private $d_2 \equiv a_1 g_2^{h_4} \pmod{p}$.
14. Return the public key tuple $(n, e_1, e_2, e_3)$ and private key tuple $(d_1, d_2, d_3, p)$.

• **Encryption by Busu**

INPUT: Along's public key set $(n, e_1, e_2, e_3)$ and the message $M$ tuple $(b_0, b_1, b_2)$ where $b_0 \approx 2^{n-1}$ and $b_1, b_2 \approx 2^{(e-2)n}$.
OUTPUT: A ciphertext pair $(C_1, C_2)$.

1. Compute the first ciphertext $C_1 = b_0^e + b_1(e_1 e_3 + 1) + b_2(e_1 e_2)$.
2. Compute the second ciphertext $C_2 = b_1 e_2 + b_2 e_3$.
3. Send the ciphertext pair $C = (C_1, C_2)$.

• **Decryption by Along**

INPUT: The ciphertext pair $C = (C_1, C_2)$ and private key tuple $(d_1, d_2, d_3, p)$.
OUTPUT: The message tuple $M = (b_0, b_1, b_2)$.

1. Compute $b_0 \equiv (C_1 - C_2(d_2 + d_3))^{d_1} \pmod{p}$.
2. Solve the simultaneous equations $C_1 - b_0^e = b_1(e_1 e_3 + 1) + b_2(e_1 e_2)$ and $C_2 = b_1 e_2 + b_2 e_3$ to obtain $(b_1, b_2)$.
3. Return the message tuple $M = (b_0, b_1, b_2)$.

**Proposition 2.** *The decryption process is correct.*

*Proof.* From $g_1^{u_1} - g_2^{u_2} \equiv g_1^{h_1} g_1^{h_3} - g_2^{h_2} g_2^{h_4} \equiv 0 \pmod p$, we have

$$\begin{aligned}(C_1 - C_2(d_2+d_3))^{d_1} &\equiv (b_0^e + b_1 + a[b_1[g_1^{u_1} - g_2^{u_2}] + b_2[g_1^{h_1} g_2^{h_2} - g_1^{h_3} g_2^{h_4}]] - [b_1 + b_2 k])^{d_1} \\ &\equiv (b_0^e + b_1 + b_2 k - [b_1 + b_2 k])^{d_1} \\ &\equiv (b_0^e)^{d_1} \\ &\equiv b_0 \pmod p.\end{aligned}$$

We obtain the exact $b_0$ since $b_0 < p$, which ensures that no modular reduction has occurred. Next, to obtain $(b_1, b_2)$ is trivial.

In the next section we will point out locations where the fundamental source of security situated.

## 5 The fundamental source of security

We will dissect the mathematical structures introduced in the above so-called "cryptosystem". We will begin at looking at Along's parameters first.

### 5.1 Security of the ciphertext

- Observe the ciphertext given by $C_1 = b_0^e + b_1(e_1e_3 + 1) + b_2(e_1e_2)$. We have $C_1 \approx 2^{(e+2)n}$ while $b_0^e b_2 b_3 \approx 2^{3en}$. Thus, $b_0^e b_2 b_3 > C_1$.
- We have $b_1, b_2 \approx 2^{en}$ while $e_2, e_3 \approx 2^n$, thus the equation $C_2 = b_1e_2 + b_2e_3$ is "protected" by BFHP.
- We also have $C_2 \approx 2^{(e+1)n}$ while $b_1b_2 \approx 2^{2en}$. Thus, $b_1b_2 > C_2$.
- To solve the simultaneous equations of $C_1, C_2$ it is a system of 2 equations with 3 variables.

### 5.2 Security of the public key

**Security type-1**
Observe the following public key equation:

$$e_1e_3 \equiv a g_1^{u_1} \pmod p \tag{5}$$

This is an equation with the following unknown tuple $(a, g_1, u_1, p)$. From $e_2 \equiv g_2^{h_2} \equiv (g_1^{u_1 d_{u_2}})^{h_2} \pmod p$, the adversary does not obtain any helpful information to "study" equation (5).

**Security type-2**
Now from another relation:

$$e_1 e_3 - e_2(a g_2^{h_4}) \equiv 0 \pmod{p} \tag{6}$$

the adversary can assume the following strategy:

- Set $g_1 = g_1', u_1 = u_1', p = p'$.
- Compute $a' \equiv (e_1 e_3)(g_1^{u_1})'^{-1} \pmod{p'}$.
- Now, choose random $u_2 = u_2'$ and compute $d_{u2'} \equiv (u_2)'^{-1} \pmod{p' - 1}$.
- Now, compute $g_2' \equiv (g_1^{u_1})'^{d_u 2'} \pmod{p'}$.
- From $e_2 = g_2'^{h_2} \pmod{p'}$, the adversary would face the DLP to determine $h_2 = h_2'$ from his own choice of $(g_2', p')$.
- If the adversary is able to determine $h_2'$ (i.e. then he would be able to compute $h_4' = u_2' - h_2'$) from the adversaries own choice of parameters, and the probability that
$$e_1 e_3 - e_2(a' g_2'^{h_4'}) \equiv 0 \pmod{p'}$$
is $2^{-4n}$.
- A faster approach for the adversary is to check whether
$$e_1 e_3 - a' g_2'^{u_2'} \equiv 0 \pmod{p'}.$$
In this case the probability is still $2^{-4n}$.
- Solving equation (6) could also be viewed as ***solving a "system" of diophantine equations with unknown exponent***.

**Security type-3**
Observe the following system of equations:

$$e_1 e_3 - e_2 d_2 \equiv 0 \pmod{p} \tag{7}$$
$$e_1 e_2 - e_3 d_2 \equiv k \pmod{p} \tag{8}$$
$$e_2 d_3 + e_3 d_3 \equiv 1 + k \pmod{p} \tag{9}$$

When we eliminate the pair $(d_2, k)$, we have the equation

$$e_2 d_3 (e_2 + e_3) + e_1 e_3^2 - e_1 e_2^2 - e_2 \equiv 0 \pmod{p} \tag{10}$$

- To solve equation (10), we set $p = p'$ and compute $d_3'$.
- Then from (9) obtain $k'$.
- Then from (8) obtain $d_2'$.
- Then check whether (7) holds or not. The probability that (7) holds is $2^{-n}$.

## 6 Subset sum - like problem?

To obtain parameters that satisfy equation (5) "mimics" the subset sum problem.

## 7 Collision type attacks

We dedicate this section to discuss the possibility of designing a collision type attack on our new scheme.

## 8 Achieving IND-CCA2

It is obvious that the new scheme achieves IND-CPA. But how about IND-CCA2?

## 9 Conclusion

This paper presents a new cryptosystem that has advantages in the following areas against known public key cryptosystems:

1. It has a complexity order of $O(n^2)$ during encryption and $O(n^3)$ during decryption.
2. *Mathematically, an adversary does not have any advantage to attack the published public key or the ciphertext.*
3. Does the new scheme produce "cylic-type" features that would allow a collision type attack to be designed?
4. If a collision type attack cannot be designed, how do we propose to evaluate the scheme in order to suggest a minimum key length?

## 10 Acknowledgment

I would like to acknowledge Prof Abderrahmane Nitaj of University of Caen, France and Dr Yanbin Pan of the Chinese Academy of Science, China for all discussion and feedbacks while designing the scheme.

## References

1. Herrmann, M., May, A.: Solving Linear Equations Modulo Divisors: On Factoring Given Any Bits. In ASIACRYPT 2008, volume 5350 of Lecture Notes in Computer Science, pp. 406-424. Springer-Verlag (2008)