A new security notion for asymmetric encryption Draft #6

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 a ***certain problem upon a polynomial***. 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 Solving a multi-variable polynomial of unknown degree

Definition 1. *Let $p_\delta(x_1, x_2, ...x_n)$ denote a multi-variable polynomial of unknown degree δ. The challenge is to determine any tuple $(x_1, x_2, ..., x_n)$ which is a solution to the polynomial. That is, to obtain a tuple that results in $p_\delta(x_1, x_2, ...x_n) = 0$.*

2.2 Linear diophantine equations with infinitely many solutions

Before we discuss this subsection we will first 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 relates the length of the shortest vector in a lattice to the determinant[1]:

Theorem 1. *In an ω-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 $|y_i|$. 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^{ϵ} 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 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.

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.

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*

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

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* $A_1 u + A_2 v \in Z^{+}_{(2^{m+n-1},2^{m+n}-1)}$*.*

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:

$$u = u_0 + A_2 t \tag{2}$$

and

$$v = v_0 - A_1 t \tag{3}$$

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

$$A_1 u_1 + A_2 v_1 \neq A_1 u_2 + A_2 v_2 \tag{4}$$

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: $u = u_0 + A_2 t$ 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 $t = 519172 \approx 2^{19}$.

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 and a pair of public values co-prime to each other $(g_1, g_2) \approx 2^n$.

OUTPUT: A public key tuple (n, e, g_1, g_2, L_{J2}) and private keys (d, v, j, J_1, J_2).

1. Generate random n-bit secret parameter $(a_1, a_2, h_1, h_2, \delta)$.
2. Compute secret parameter $G_1 = h_1 g_1 - h_2 g_2, G_2 = h_1 g_2 - h_2 g_1, H_1 = a_1 G_2 - a_2 G_1, H_2 = a_1 G_1 - a_2 G_2$.
3. Compute secret parameter $j = gcd(H_1, H_2), J_1 = \frac{H_1}{j}, J_2 = \frac{H_2}{j}$.
4. Compute public parameter: length of J_2 denoted by L_{J2}.
5. Compute secret parameter $u = a_1 h_1 + a_2 h_2, v = a_1 h_2 + a_2 h_1$. Ensure that $gcd(v, H_2) = 1$.

6. Compute private key $d = 3^{-1} \pmod{(\phi(J2))}$. Ensure that d exists. Otherwise repeat the above processes again.
7. Compute public key $e = \frac{u+J_2^{\delta}}{v} \pmod{H_2}$.
8. Return the public key pair (n, e, g_1, g_2, L_{J2}) and private keys (d, v, j, J_1, J_2).

Remark 5. Let $s = gcd(eg_1 - g_2, eg_1 - g_2)$.Ensure that $3^{-1} \pmod{\phi(\frac{eg_1-g_2}{s})}$ does not exist. Reasons will be clear in following paragraphs.

• Encryption by Busu

INPUT: The message M tuple (b_1, b_2) where $b_1 \approx 2^{L_{J2}-1}$ and $b_2 \approx 2^{3L_{J2}}$, and Along's public key set (n, e, g_1, g_2, L_{J2}).

OUTPUT: A ciphertext C.

1. Compute secret parameter $B_1 = g_1b_2 + g_2b_1^3, B_2 = g_1b_1^3 + g_2b_2$.
2. Compute the ciphertext $C = B_1e - B_2$. Equivalently $C = b_1^3(eg_2 - g_1) + b_2(eg_1 - g_2)$.
3. Send the ciphertext pair C.

• Decryption by Along

INPUT: The ciphertext C and private key tuple (d, v, j, J_1, J_2).

OUTPUT: The message tuple $M = (b_1, b_2)$.

1. Compute $b_1 = \left(\frac{Cv}{jJ_1}\right)^d \pmod{J_2}$.
2. Compute $b_2 = \frac{C-b_1^3(eg_2-g_1)}{eg_1-g_2}$.
3. Return the message tuple $M = (b_1, b_2)$.

Proposition 2. *The decryption process is correct.*

Proof.

$$\begin{aligned} w \equiv Cv(\tfrac{1}{j}) &\equiv [B_1(u + J_2^{\delta}) - B_2v](\frac{1}{j}) \\ &\equiv [(g_1b_2 + g_2b_1^3)(a_1h_1 + a_2h_2) - (g_1b_1^3 + g_2b_2)(a_1h_2 + a_2h_1) + B_1J_2^{\delta}](\frac{1}{j}) \\ &\equiv [b_1^3(a_1(h_1g_2 - h_2g_2) - a_2(h_1g_1 - h_2g_1)) + b_2(a_1(h_1g_1 - h_2g_2) - a_2(h_1g_2 - h_2g_1)) + B_1J_2^{\delta}](\frac{1}{j}) \\ &\equiv [b_1^3H_1 + b_2H_2 + B_1J_2^{\delta}](\frac{1}{j}) \\ &\equiv b_1^3J_1 \pmod{J_2} \end{aligned} \tag{5}$$

Then,

$$\left(\tfrac{w}{J_1}\right)^d \equiv \left(\tfrac{Cv}{jJ_1}\right)^d \equiv b_1 \pmod{J_2} \tag{6}$$

From (6), since $b_1 < J_2$, then no modular reduction has occurred. Thus, we have obtained the full value of b_1. To obtain 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 $C = B_1 e - B_2 = b_1^3(eg_2 - g_1) + b_2(eg_1 - g_2)$. We have ensured that $3^{-1} \pmod{\phi(\frac{eg_1 - g_2}{s})}$ does not exist, where $s = gcd(eg_1 - g_2, eg_1 - g_2)$.

- We have $b_1^3, b_2 \approx 2^{3L_{J2}} \approx 2^{6n}$ while $eg_2 - g_1, eg_1 - g_2 \approx 2^{4n}$.
- We have $C \approx 2^{10n}$ while $b_1^3 b_2 \approx 2^{12n}$. Thus, $b_1^3 b_2 > C$.

5.2 Security of the public key

Observe equation (5). the objective is clear to the adversary. That is, to launch a passive attack via any candidate value to obtain

$$w' \equiv b_1^3 J_1' \pmod{J_2'} \tag{7}$$

For discussion sake, let us observe if the public is given by

$$e = \frac{u}{v} \pmod{H_2}$$

That is,

$$u = ev + H_2 t$$

for some $t \in \mathbb{Z}$. Let $t = t_0$. Since (g_1, g_2) are public, by setting $h_1 = h_1', h_2 = h_2'$ we can compute $G_1' = h_1' g_1 - h_2' g_2, G_2' = h_1' g_2 - h_2' g_1$. Following this, from the relation:

$$a_1 h_1' + a_2 h_2' = e(a_1 h_2' + a_2 h_1') + (a_1 G_1' - a_2 G_2')t_0$$

we can have the following:

$$a_1(h_1' - eh_2' - G_1' t_0) = a_2(eh_1' - h_2' - G_2' t_0)$$

This is a linear diophantine equation in 2 variables. We can easily solve by

$$a_1 = (eh_1' - h_2' - G_2' t_0)r$$

and

$$a_2 = (h'_1 - eh'_2 - G'_1 t_0)r$$

for some $r \in \mathbb{Z}$. These "attacked values" can be utilized to launch a passive attack. That is, the adversary can achieve equation (7) in polynomial time via the same decryption process as outlined by Proposition 2.

However, in our scheme the public key is given by:

$$e = \frac{u + J_2^{\delta}}{v} \pmod{H_2}$$

This can interpreted as

$$u + J_2^{\delta} = ev + H_2 t$$

Via the same procedure we will set $h_1 = h'_1, h_2 = h'_2, t = t_0$ and then proceed to compute G'_1, G'_2. Since by definition $J_2 = \frac{H_2}{j}$ $(j = gcd(H_1, H_2))$ and by utilizing the candidate value $H'_2 = a_1 G'_1 - a_2 G'_2$ we will have

$$a_1 h'_1 + a_2 h'_2 + (\frac{a_1 G'_1 - a_2 G'_2}{j})^{\delta} = e(a_1 h'_2 + a_2 h'_1) + (a_1 G'_1 - a_2 G'_2)t_0$$

This will lead us to the problem of solving a polynomial of unknown degree δ in 2 variables.

6 Collision type attacks

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

7 Achieving IND-CCA2

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

8 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 $O(n^2)$ both during encryption and decryption.
2. Mathematically, an adversary does not have any advantage to attack the published public key (i.e. because of the unknown degree of the polynomial) 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?

9 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)