JSIAM Letters Vol.6 (2014) pp.85–88 ©2014 Japan Society for Industrial and Applied Mathematics JSIAM Letters

A key exchange protocol based on Diophantine equations and S-integers

Attila Bérczes[1], Lajos Hajdu[1], Noriko Hirata-Kohno[2], Tünde Kovács[1,2] and Attila Pethő[1]

[1] University of Debrecen, 4032 Debrecen, Egyetem tér 1, Debrecen, Hungary
[2] Nihon University, 4-8-24 Kudan-Minami, Tokyo 102-8275, Japan

E-mail *hirata@math.cst.nihon-u.ac.jp*

Received June 9, 2014, Accepted September 22, 2014

Abstract

The aim of this article is to present a cryptosystem with a new key exchange protocol based on Diophantine equations of polynomial type. Our protocol is inspired by that of H. Yosh whose security comes from a translation of Diophantine equations. We suggest here a key exchange protocol relying on the hardness of solving Diophantine equations in the ring of S-integers.

Keywords cryptography, key exchange protocol, S-integers, Diophantine equation

Research Activity Group Algorithmic Number Theory and Its Applications

1. Introduction

The starting point of public key cryptography is considered in the article of Diffie and Hellman [1] where the authors describe a new kind of cryptography, including the need of a key distribution system, known as the Diffie-Hellman key exchange protocol. The theory of public key cryptography has gone through a vast development since the introduction of their protocol. Some protocols turned out to be un-secure, and others were considered to be safe. However, a breakthrough due to the continuous efforts may break the security of any protocol, so creating new key exchange protocols remains one of the primary tasks in the theory of cryptography. Indeed, key exchange protocols are mainly based on mathematical problems, which are sufficiently difficult.

In 2011, H. Yosh [2] suggested the use of a key exchange protocol, the security of which is based on the hardness of solving Diophantine equations (indeed, the meaning of key exchange protocol here is slightly different from the usual terminology, but we follow the way proposed in [2]). In [3], N. Hirata-Kohno and A. Pethő analyzed the protocol due to Yosh, revealing several weaknesses of the protocol, and suggested a modification of it. They removed partially the weaknesses and suggested a choice of the parameters, which is secure against ciphertext-only attack.

We give here a new key exchange protocol based on S-integer solutions to Diophantine equations with an example, relying again on the idea by Yosh, but additionally combined with the complexity of S-integers. In our new protocol, the public key size is much less than in the previous versions, but provides at least the same level of security.

2. The key exchange protocol of H. Yosh

Let R be a ring. The protocol of Yosh is defined in the case $R = \mathbb{Z}$, but the idea works in the same way for different rings, therefore we shall present the protocol in a general case. In [3] Hirata-Kohno and Pethő simplified the protocol of Yosh, according to their needs, however that is essentially the protocol of Yosh. We shall describe it now in details.

Alice and Bob are willing to agree in a secret key using only unsecured channels for their communications. In order to do this they perform the following steps:

(i) Alice chooses elements $r_1, \dots, r_m \in R$ and constructs a polynomial Diophantine equation with coefficients in R:

$$f(X_1, \dots, X_m) = 0, \quad \text{in } X_1, \dots, X_m \in R \qquad (1)$$

such that the m-tuple $(r_1, \dots, r_m) \in R^m$ is a solution to the equation (1).

(ii) Alice keeps the m-tuple $(r_1, \dots, r_m) \in R^m$ secret, and sends the polynomial $f(X_1, \dots, X_m)$ to Bob via the unsecured channel. Consequently, the polynomial $f(X_1, \dots, X_m)$ has to be considered public.

(iii) Bob chooses randomly a polynomial $g(X_1, \dots, X_m) \in R[X_1, \dots, X_m]$ and chooses random elements $a_i \in R$ and $0 \le b_i \in \mathbb{Z}$ $(1 \le i \le n)$ with $b_1, \dots, b_n$ odd, and defines the function

$$T_{a_i,b_i}(X) := (X + a_i)^{b_i} \quad (1 \le i \le n)$$

so as to be invertible. Then Bob computes the polynomial

$$H(X_1, \dots, X_m) := T_{a_n,b_n}\left(\dots T_{a_1,b_1}\left(g(X_1, \dots, X_m)\right)\dots\right)$$

and takes a random element

$$h(X_1, \dots, X_m) \in H(X_1, \dots, X_m) + f(X_1, \dots, X_m) \cdot R[X_1, \dots, X_m].$$

(iv) Bob then sends g and h to Alice through the unsecured channel, but he keeps the elements $a_i \in R$

and $b_i \in \mathbb{Z}_{\geq 0}$ $(1 \leq i \leq n)$ secret.

(v) Alice, in the possession of g and h, computes the values $s = g(r_1, \ldots, r_m)$ and $u = h(r_1, \ldots, r_m)$, and sends the element u to Bob through the unsecured channel.

(vi) For $1 \leq i \leq n$, Bob computes the inverse functions $T^{-1}_{a_i,b_i}$ to the bijective polynomial functions T_{a_i,b_i}, and obtains the value

$$s = T^{-1}_{a_1,b_1}\left(\ldots T^{-1}_{a_n,b_n}(u)\ldots\right),$$

which should be the shared secret of Alice and Bob.

3. Previous results concerning the security of the protocol of Yosh

In [3] the authors proved the correctness of the protocol of Yosh, simplified it and gave a careful analysis of the security of the simplified protocol.

They also suggested a finite field version. For the choice of the polynomial f the most important requirement is that it has to be extremely hard to solve the equation $f(X_1, \ldots, X_m) = 0$ in R^m. More precisely the authors proved in [3]:

Proposition 1 ([3, Proposition 3]) *If the adversary can compute more than $2m$ solutions to* (1)*, not necessarily $(r_1, \ldots, r_m)$, then he/she can compute the element s and breaks the protocol.*

Compared to Proposition 1, there is an important drawback to the finite field version of the protocol. Indeed, if we choose random values $r_2, \ldots, r_m$ from the finite field for $X_2, \ldots, X_m$ then a question to decide whether the equation

$$f(X_1, r_2, \ldots, r_m) = 0$$

has a solution in X_1 from the finite field or not, can be answered in probabilistic polynomial time [4], and in the case, a solution also can be found in probabilistic polynomial time. This might enable the attacker to find many solutions to the equation (1), which in view of Proposition 1 undermines the security of the protocol. So we cannot consider the finite field version of the protocol safe.

Thus in the present paper we suggest a variant of the protocol of Yosh, which works over the rational integers and the ring of S-integers, in the case when considerably more solutions to (1) are needed to break the protocol.

4. Our new key exchange protocol on the ring of S-integers

In many cases, even if it is not possible to completely solve a Diophantine equation, it may be feasible to find several "small" solutions by chance. This makes the protocol of Yosh un-secure, both in its original form and in the modified form analyzed in [3]. This weakness might be compensated by choosing the parameter n large, but as pointed out in [3], this becomes impossible by practical considerations. Further, by the same reason the positive integers b_i $(1 \leq i \leq n)$ must be also very small. Thus the number of the free parameters $a_i \in R$ and $b_i \in \mathbb{Z}_{\geq 0}$ $(1 \leq i \leq n)$ cannot be sufficiently increased in the protocol of Yosh.

We mention that the polynomial functions T_{a_i,b_i} in the protocol of Yosh are of a special form only because this form may guarantee that their composite function is invertible. So, in our new key exchange protocol, first we suggest to choose a general polynomial function T which is invertible, instead of the function $T_{a_n,b_n}(\ldots T_{a_1,b_1}(X)\ldots)$.

Second, let $S = \{p_1, \ldots, p_k\}$ be a finite set of distinct rational primes with a suitable k. Consider a rational number a/b with $a, b \in \mathbb{Z}$ and $\gcd(a, b) = 1$, such that the (possibly empty) set of prime divisors of b is contained in S. This rational number is a so-called S-integer (corresponding to the specific set S). Denote by $\mathbb{Z}_S$ the set of S-integers. Clearly, this set $\mathbb{Z}_S$ is a subring of $\mathbb{Q} \subset \mathbb{R}$ and $\mathbb{Z}_S$ contains $\mathbb{Z}$. The elements of $\mathbb{Z}_S$ have the property that in their denominators, the exponents of the primes lying in S can be arbitrarily large.

In this article we choose $R = \mathbb{Z}_S$ and we present the following modification of the protocol of Yosh. The main idea is that Alice considers $r_1, \ldots, r_m \in \mathbb{Z}_S$ and Bob chooses $T \in \mathbb{Z}_S[X]$ in the step of the construction of the polynomial T, that makes the key exchange protocol possibly more secure, relying on the difficulty of finding solutions in $\mathbb{Z}_S$ by random search.

Choosing a solution in $\mathbb{Z}_S$, we note that it is an easy task to find a Diophantine equation which vanishes at this selected solution, but it is not at all easy to find a solution to a given Diophantine equation in S-integers. This is a typical one-way function to make a key exchange protocol.

Our new key exchange protocol is as follows. Alice and Bob choose a finite set of distinct rational primes $S = \{p_1, \ldots, p_k\}$ with a suitable large k. They keep this set S and proceed as follows.

(i) Alice chooses elements $r_1, \ldots, r_m \in \mathbb{Z}_S$ and constructs a polynomial Diophantine equation in m unknowns with coefficients in $\mathbb{Z}$:

$$f(X_1, \ldots, X_m) = 0, \quad \text{in } X_1, \ldots, X_m \in \mathbb{Z}_S \qquad (2)$$

such that the m-tuple $(r_1, \ldots, r_m) \in \mathbb{Z}_S^m$ is a solution to the equation (2) (note that the coefficients of $f(X_1, \ldots, X_m)$ are in $\mathbb{Z}$).

(ii) Alice keeps the m-tuple $(r_1, \ldots, r_m) \in \mathbb{Z}_S^m$ secret, and sends the polynomial $f(X_1, \ldots, X_m)$ to Bob via the unsecured channel. Consequently, the polynomial $f(X_1, \ldots, X_m)$ has to be considered public knowledge.

(iii) Bob chooses randomly a polynomial in m variables $g(X_1, \ldots, X_m) \in \mathbb{Z}[X_1, \ldots, X_m]$ and chooses another random polynomial function $T(X) \in \mathbb{Z}_S[X]$ such that $T : \mathbb{R} \mapsto \mathbb{R}$ is strictly monotonically increasing, namely invertible. Bob then computes the polynomial

$$H(X_1, \ldots, X_m) = T\left(g(X_1, \ldots, X_m)\right),$$

and takes a random element

$$h(X_1, \ldots, X_m)$$

$$\in H(X_1,\dots,X_m) + f(X_1,\dots,X_m)\cdot\mathbb{Z}_S[X_1,\dots,X_m].$$

(iv) Bob sends g and h to Alice through the unsecured channel, but he keeps the polynomials $T(X)$ and $H(X_1,\dots,X_m)$ secret.

(v) Alice, in the possession of g and h, computes the values $s = g(r_1,\dots,r_m)$ and $u = h(r_1,\dots,r_m)$, and sends the element u to Bob through the unsecured channel, while she keeps the value s secret.

(vi) Knowing that the polynomial function $T : \mathbb{R} \mapsto \mathbb{R}$ is strictly monotonically increasing continuous function, indeed bijective, Bob computes the value

$$s = T^{-1}(u),$$

which should be the shared secret of Alice and Bob.

To ensure that $T : \mathbb{R} \mapsto \mathbb{R}$ is strictly increasing we need that dT/dX is positive on $\mathbb{R}$. This is fulfilled if the degree of T is odd and the coefficients are well chosen. Here we have to mention that in the last step of the protocol, to compute s one can use the secant method for the polynomial $T(X) - u$, since we know that s is the only real root of $T(X) - u = 0$.

Proposition 2 *The protocol described in Section 4 is correct.*

Proof Alice can compute s because she knows g and $r_1,\dots,r_m$. As $f(r_1,\dots,r_m) = 0$, we have

$$u = h(r_1,\dots,r_m) = H(r_1,\dots,r_m).$$

Since $H(r_1,\dots,r_m) = T(g(r_1,\dots,r_m)) = T(s)$ and $T(X)$ is invertible, we have

$$s = T^{-1}(u).$$

Bob can computes s by the secant method (this proof is essentially same as in [3, Proposition 1]).

(QED)

5. Security aspects

We analyze our protocol from mathematical and cryptographical point of view. It was proved in 1971 by Y. Matijasevic (see [5]) that the solvability of polynomial Diophantine equations in integers, thus in S-integers too, is algorithmically not decidable. Nevertheless, there are also large classes of Diophantine equations which can be solved by algorithms (see e.g. [6,7]). However, as in our protocol, if a polynomial is constructed with a prescribed solution, then this solution can be computed in at most exponential time in the size of the solution.

In order to have our protocol efficient enough, we have to choose the form of f such that its parameters are easy to compute, when a solution vector is given. On the other hand, by Proposition 1, we have to choose f such that it is hard to find solutions $(r_1,\dots,r_m) \in \mathbb{Z}_S^m$ to the equation

$$f(X_1,\dots,X_m) = 0.$$

These requirements are obviously contradictory. We argued in [3] that diagonal polynomials may satisfy both requirements.

The parameters g, T and r can be chosen randomly, thus h is a random element of the $T(g) + f\mathbb{Z}_S$. Besides f also g, h and u are public objects, and the relation $h(r_1,\dots,r_m) = u$ is public as well. Thus already a passive adversary knows that $(r_1,\dots,r_m)$ satisfies the "system" of equations

$$f(r_1,\dots,r_m) = 0, \quad h(r_1,\dots,r_m) = u. \tag{3}$$

When $m > 4$, if h is chosen as a random polynomial and f as a diagonal one such that this system defines a non-singular algebraic variety in $\mathbb{R}^m$ of codimension 2, we may expect that it is at least similarly hard to find an S-integer solution to (3) as to (1).

We have to mention a weak point of our protocol. The key pairs of public key cryptosystems are stable objects, they can be used several times. This property is used in multiple-user setting such as a client server model. However, the public keys in the protocol of Yosh do not have this property and the polynomial f and its roots are only for a single action. If Alice would create with k partners common keys using always the same f and $r_1,\dots,r_m$ then denoting by $h_1,\dots,h_k$ the corresponding polynomials computed in Step (iii) and setting $u_i = h_i(r_1,\dots,r_m), i = 1,\dots,k$ the passive adversary would get $k+1$ independent equations

$$f(r_1,\dots,r_m) = 0, \quad h_i(r_1,\dots,r_m) = u_i, \quad i = 1,\dots,k$$

for $r_1,\dots,r_m$. If $k+1 \geq m$ then these determine uniquely $r_1,\dots,r_m$. With this respect the protocol of Yosh behaves as a one time pad, consequently, in the present form, it cannot be applied in multiple-user setting. We should concentrate us on this problem against multiple-user setting in our future work.

We also point out that there might exist a way to obtain the value $g(r_1,\dots,r_m) = s$ without precisely knowing $r_1,\dots,r_m$, but only f, h, g and u being given. An investigation about such a possibility is an important and essential problem, which is to be considered in our situation.

6. Example

Finally we present an example as follows. Let $S := \{167, 359, 379\}$. We perform the following steps.

(i) Alice chooses the polynomial f with the following coefficients.

$$f = c_1X_1^2 + c_2X_2^5 + c_3X_3^3 + c_4X_4^7 + c_5X_5^4 + c_6,$$

$$c_1 = 4806529705,$$

$$c_2 = -6205175372,$$

$$c_3 = 925478963,$$

$$c_4 = -768530557342240919,$$

$$c_5 = 1746745227,$$

$$c_6 = 4946407506070084575251776766468057476355317931641.$$

Alice keeps an S-integer solution $(r_1, r_2, r_3, r_4, r_5)$

to $f = 0$ secret which is

$$r_1 = \frac{4747053250}{167},$$
$$r_2 = 17914675,$$
$$r_3 = \frac{1640439652}{379},$$
$$r_4 = \frac{9078809}{359},$$
$$r_5 = 3039073006.$$

Note that for $(r_1, r_2, r_3, r_4, r_5)$ we have at least one index i such that $r_i \in \mathbb{Z}_S \backslash \mathbb{Z}$.

Actually, Alice first generates randomly the solution $(r_1, \ldots, r_5)$ then computes the coefficients $c_1, \ldots, c_6$, which ensure $f = 0$.

In this example, we simply construct f such that the terms $c_1 r_1^2, c_2 r_2^5, c_3 r_3^3, c_4 r_4^7, c_5 r_5^4$ are all integers. However, we can construct f such that these terms are not simultaneously all integers but we have $f(r_1, r_2, r_3, r_4, r_5) = 0$. This step to construct f means solving a linear Diophantine equation, that can be done by the generalized Euclidean algorithm (see [8, p. 31]).

(ii) Bob sets

$$\begin{aligned} g = {} & 234578 - 29879731X_2 + 26864732X_5 \\ & - 48958473X_1X_2 + 7145266643X_3^2 \\ & + 5537433896X_2X_4 \end{aligned}$$
$$\begin{aligned} T = {} & 476538X^5 + 703764X^4 + 893596X^2 \\ & + 31980091X + 43626626, \end{aligned}$$
$$h \equiv T(g) \mod f.$$

(iii) Alice computes s, u and gets the following result.

$$s = \frac{95969333849894392973555800718295 1}{8611708873},$$
$$u = \frac{u_1}{u_2},$$

where

$$\begin{aligned} u_1 = {} & 3879358709869226733566718595284408 25 \\ & 0487184287276298373175190144568713 55 \\ & 5545632009950458737433436138617944 26 \\ & 3882902166006837623102344920359076 05 \\ & 5037950450389851860575611 41, \end{aligned}$$
$$\begin{aligned} u_2 = {} & 4736381742001944320338406709171444 09 \\ & 67619738975593. \end{aligned}$$

(iv) Using the secant method, Bob computes

$$T^{-1}(u) = s.$$

The example shows clearly that the new protocol is superior to the protocol of [3]. In the present example f has one more variable than in [3]. In both examples T has five parameters, but in [3] only three parameters were free, because the other two parameters could assume only very small values, while in the present case all the five parameters are essentially free. They can be arbitrary large satisfying the mild assumption dT/dX is positive on $\mathbb{R}$. Thus by Section 5, the present example is at least as secure as the example in [3].

On the other hand the size of the public key of this example is much smaller than of it of [3]. Indeed, we presented in both cases explicitly the secret keys: $r_1, \ldots, r_m, T(H), s$, and the public keys: f, g, u. The only missing data is h because this polynomial has a long form and we supposed that it might be waste of paper to give all of the form here explicitly, thus we try to explain it as follows. We computed both examples with MAPLE 13, which gave us the size of the internal representation of h, which is a multivariate polynomial. As such objects do not have a canonical representation, the most honest way to compare the size of two such polynomials is to give the size of their internal representation in the same computer algebra system. The polynomial h of [3] has 2107 terms of form $aX_1^{n_1}X_2^{n_2}X_3^{n_3}X_4^{n_4}$, where a denotes an integer and $n_1, \ldots, n_4$ non-negative integers. Moreover the internal representation in MAPLE 13 has length 800327. In contrast, the same parameter in the present example has only 269 terms and its internal representation in MAPLE 13 has length 18240.

Acknowledgments

The research was mainly supported by the Funding Program for Next Generation World-Leading Researchers (NEXT Program, JSPS), whose grant number is GR 087. It was also supported by JSPS P12806 for the fourth author, and partially supported by the University of Debrecen, by grants K100339 and NK104208 of the Hungarian National Foundation for Scientific Research, and by the European Union and the European Social Fund through project Supercomputer, the national virtual lab (grant no.: TAMOP-4.2.2.C-11/1/KONV-2012-0010). We are obliged to sincerely thank the referee for his/her excellent work giving essential advices for our investigation.

References

[1] W. Diffie and M. Hellman, New direction in cryptography, IEEE Trans. Inform. Theory, **22** (1976), 644–654.

[2] H. Yosh, The key exchange cryptosystem used with higher order Diophantine equations, IJNSA Journal, **3** (2011), 43–50.

[3] N. Hirata-Kohno and A. Pethő, On a key exchange protocol based on Diophantine equations, Infocommunications Journal, **5** (2013), 17–21.

[4] M. Mignotte, Mathematics for Computer Algebra, Springer-Verlag, New York, 1992.

[5] M. Davis, Y. Matijasevic and J. Robinson, Hilbert's tenth problem, Diophantine equations: positive aspects of a negative solution, in: Proc. of Mathematical Developments Arising from Hilbert Problems, F. E. Browder ed., Proc. Sympos. Pure Math., Vol. 28, pp. 323–378, AMS, Providence, R.I., 1976.

[6] A. Baker, Transcendental Number Theory, Cambridge Univ. Press, Cambridge, 1975.

[7] J. H. Silverman, The Arithmetic of Elliptic Curves, 2nd ed., Graduate Texts in Mathematics, Vol. 106, Springer-Verlag, New York, 2009.

[8] L. J. Mordell, Diophantine Equations, Pure and Applied Mathematics, Vol. 30, Academic Press, New York, 1969.