Analysis for Memory Reduction of the UOV Scheme with the Application of PRNG

Jihyun Kim and Howon Kim

***Abstract*—In case we use a quantum computer, we have to construct the countermeasure. Due to the high efficiency and fast computation time, the multivariate quadratic public key systems are considered as an alternative to RSA or ECC based systems. However, the large key size is a fatal disadvantage of the multivariate quadratic public key systems. For this reason, the multivariate quadratic public key systems are not widely used. In this paper, we measure how much the private key memory size can be reduced by using a secure pseudo-random number generator in the UOV scheme. Almost 93% of the private key memory size is reduced.**

***Index Terms*—Multivariate quadratic public key system, pseudo-random number generator, UOV scheme.**

I. Introduction

In modern society, public key cryptosystems are used in everywhere around us such as e-commerce, online banking and communication. We used to use RSA scheme which is the most popular based on the difficulty of factoring large integers. RSA is considered to be secure but it will be different when it comes to using quantum computers [1]. Other public key schemes such as ECC and El Gamal are the same. In case when the quantum era arrives, we have to establish the countermeasures. Multivariate schemes are considered as an alternative.

Multivariate cryptosystem is based on the problem of solving Multivariate Quadratic equations (MQ-problem) over finite fields. MQ-problem is NP-complete. There are several public key schemes based on MQ-problem like [2], [3] and [4]. Although multivariate cryptosystem is quite fast, it is not widely used due to the large key size. For this reason, it is hard to be used for memory constrained devices such as smart cards and wireless sensor nodes. Therefore reducing the key size is a significant issue.

In this paper, we measure how much the private key size can be reduced when we apply a pseudo-random number generator to Unbalanced Oil and Vinegar Signature (UOV) Scheme. We use a secure pseudorandom number generator for generation the private key. By this way we confirm that almost 93% of the private key size can be reduced.

In subsection A from Section I, we describe the multivariate quadratic scheme. The most important part of the multivariate quadratic scheme is a central map. We have a description of UOV as a central map in Section II and also modified UOV can be found. We give a brief account of BBS algorithm in Section III. In Section IV, we explain how to apply PRNG to UOV scheme detailedly and we give the result of a practical experiment in section V. In section VI, we make mention of consideration and conclude our paper in Section VII.

Multivariate Quadratic Scheme

Let F be a finite field and q be a number of elements of F. Let E be an extension of the ground field F. X is a set of input variables where $n = |X|$ and Y is a set of output variables where $m = |Y|$.

The main idea of the MQ scheme is $F^n \to F^m$. Fig. 1 shows a flow of the MQ scheme. S, Q and T are a private keys of the MQ scheme and P is a public keys of this scheme.

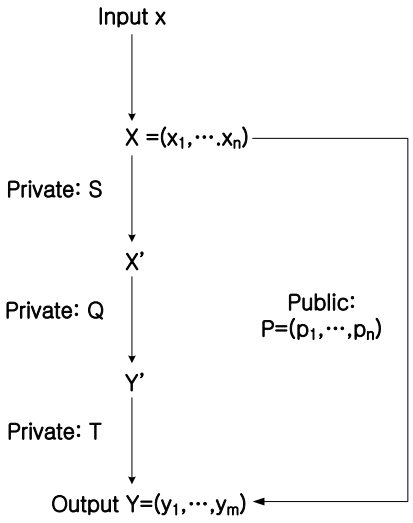

Fig. 1. Flow of the MQ scheme.

The main stream is occurred in the private key Q and we call it a central map. S and T are an affine transformation and they can be represented in the form of an invertible matrix $M_s \in F^{n \times n}$ and a vector $v_s \in F^n$,i.e, we have $S(X) = M_s X + v_s$. M_t and V_t can also be shown like this over $F^{m \times m}$ and F^m [5]. To get the public key P, we use a composite function with the affine transformations $S : F^n \to F^n$, $T : F^m \to F^m$ and the central map $Q : F^n \to F^m$. Now the public key P can be represented in $P = T \circ Q \circ S$.

In the MQ Scheme, we define the polynomial vector $P = (p_1, \ldots p_m)$. According to the polynomial vector, we have a form that denotes a i-th vector of the public key.

$$y_r(x) = \sum_{i \in v, j \in 0} r_{ij}^{(r)} x_j x_j + \sum_{i,j \in v}^{n} \varepsilon_j^{(r)} x_j + \delta^{(r)}, r = 1, \ldots 0 \quad (1 \le i \le m) \quad (1)$$

Manuscript received November 10, 2012; revised January 25, 2013.

The authors are with the Department of Computer Engineering, Pusan National University South Korea (e-mail: jihyunkim@pusan.ac.kr, howonkim@pusan.ac.kr).

DOI: 10.7763/IJCCE.2013.V2.205

$\alpha_{i,j,k}, \beta_{i,j}$ and ω_i are the coefficients of the equations (1) over F. The public key can also be expressed in matrix after graded lexicographical ordering.

1) Encryption and Signature Verification. In the MQ scheme, encryption process and signature verification process are same because of using public key. We choose an input vector $X \in F^n$ and therewith perform the polynomials (1). Then we can obtain an output vector $Y \in F^m$ which is treated as ciphertext or the original of the signature.
2) Decryption and Signature Generation. For decryption and signature generation, we have to perform the inverse process of encryption and signature verification process with the private keys and a given $Y \in F^m$. The affine transformation S and T can easily be inverted as long as we utilize the matrices, i.e, $Y' = M_T^{-1}(Y - v_t)$. However the way how to invert the private key Q depends on the structure of the central map. In section 2, we will discuss about obtaining pre-image of Y' in UOV scheme.

II. The Unbalanced Oil and Vinegar Signature Scheme

In this section, we will give a description about UOV Scheme. The Oil and Vinegar Signature Scheme was proposed by J. Patarin in [6]. But it was broken by A. Kipnis and A. Shamir [7]. A Kipnis and J Patarin found that if we have significantly more "vinegar" unknowns that "oil" unknowns, then the attack of [7] does not work and the security of this more general scheme is still an open problem.

A. Basic UOV Scheme

Let K be a finite field. Let *v* be a number of vinegar variables and o be a number of oil variables. Let n be a total number of variables and set *n = o + v*.

As appears by Fig. 1, X' will be the input of the UOV scheme as an element of K^n and Y' will be the output of the UOV scheme as an element of K^o, where $X' = \{x_1', ..., x_n'\}, Y' = \{y_1', ..., y_o'\}$. The input variable set X' can be divided into two set which are vinegar variable set and oil variable set. Therefore we are able to present X' this way, $X' = \{x_1', ..x_v', x_{v+1}', ..x_{o+v}'\}$. We define o quadratic polynomials $Q(X') = Y'$ by

$$y_r'(x) = \sum_{i \in v, j \in 0} r_{ij}^{(r)} x_j' x_j' + \sum_{i,j \in v}^{n} \varepsilon_j^{(r)} x_j' + \delta^{(r)}, r = 1, ...0 \quad (2)$$

It should be noted that there is no terms composed of only oil variables.

The coefficients $\gamma_{ij}^{(r)}, \lambda_{ij}^{(r)}, \varepsilon_i^{(r)}, \delta^{(r)} \in K$ are the private key of the UOV scheme. Private S, the other private key of the UOV scheme, can be presented in a linear invertible matrix. Because the private T is not needed for the security we don't use it. As a result, we can get the public key of the UOV scheme by

$$P = Q \circ S \quad (3)$$

Remark 1. Basically the MQ scheme needs two affine maps and one central map but in the UOV scheme the second affine map T is not needed for the security. So we have $P = Q \circ S$

1) Signature Generation. If we randomly choose the v unknowns of K, we could get a system of o linear equation in the o variables because there is no terms which are consist of only oil variables. Due to that, it is possible to compute the equations by Gaussian reductions. If there is no solution, we should try again with new random vinegar variable. After finding a X', we can get the signature by $X = S^{-1}(X')$. X is the signature.
2) Signature Verification. A signature X is valid if and only if all public equation P is satisfied.

B. Cyclic UOV Scheme

Petzoldt A., Bulygin S. and Buchmann J suggested a new idea to reduce the public key size of multivariate cryptosystems by using a partially cyclic public key in [8]. They reduced the size of the public key by up to 83%.

In this section we review that how the authors of [8] reduce the size of public key.

The authors of [8] found the relation between the coefficients of the quadratic terms of P and Q after choosing private S which is a linear invertible matrix.

$$p_{ij}^{(r)} = \sum_{k=1}^{n} \sum_{l=k}^{n} \alpha_{kl}^{ij} \cdot y'^{(r)}_{kl} = \sum_{k=1}^{v} \sum_{l=k}^{n} \alpha_{kl}^{ij} \cdot y'^{(r)}_{kl}$$
$$(i \le i \le j \le n, r = 1, ..., o) \quad (4)$$

with

$$\alpha_{kl}^{ij} = \begin{cases} s_{ki} \cdot s_{li} & (i = j) \\ s_{ki} \cdot s_{lj} + s_{kj} \cdot s_{li} & otherwise \end{cases} \quad (5)$$

$p_{ij}^{(r)}$ and $y'^{(r)}_{kl}$ are the coefficients of the quadratic terms in r-th polynomial of *P* and *Q*. The second equation in (4) is established owing to excluding oil terms and s which is shown in (5) is an element of an affine map *S*.

Let *L* be the length of the non-zero quadratic terms in *Q* and *L'* be the length of the quadratic terms in the public polynomials.

$$L = \frac{v \cdot (v+1)}{2} + o \cdot v \quad L' = \frac{n \cdot (n+1)}{2} \quad (6)$$

The authors of [8] defined a matrix A which has L rows and L columns for the equation (5).

A = $\left(\alpha_{kl}^{ij}\right)$ ($1 \le k \le v$, $k \le l \le n$ for the rows, $1 \le i \le v$, $i \le j \le n$ for the columns)

$$\mathbf{A} = \begin{bmatrix} \alpha_{11}^{11} & \alpha_{11}^{12} & \cdots & \alpha_{11}^{vn} \\ \alpha_{12}^{11} & \alpha_{12}^{12} & \cdots & \alpha_{12}^{vn} \\ \vdots & & & \vdots \\ \alpha_{vn}^{11} & \alpha_{vn}^{12} & \cdots & \alpha_{vn}^{vn} \end{bmatrix} \quad (7)$$

So (4) is represented by

$$quadP = quadQ \cdot \mathbf{A} \quad (8)$$

where $quadP$ and $quadQ$ are submatrices of the public key P matrix and private Q matrix. If the matrix A is invertible, $quadQ$ could be unique. Thus the equation (8) can be bijective.

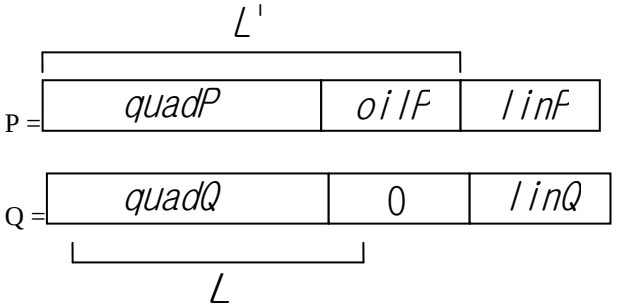

Fig. 2. Matrix layout of public key P and private key Q.

Fig. 2 shows the matrix layout of public key P and private key Q. P and Q are matrices and each of them consists of the coefficients of the polynomials.(constant coefficients of each polynomial are not included) oil/F is a submatrix of the P and it consists of the coefficients of the terms which are composed of oil variables. As we mentioned before, there are no oil terms are in the polynomial (2). $linF$ and $linQ$ are the linear coefficients of each polynomials.

To insert a partially circulant matrix into the public key, the authors of [8] chose a vector $b = (b_1, , , b_L) \in_R K^L$ for an anchor of the public key and shifted a vector b by

$$v_P^{(i)} = S^{i-1}(b) \quad i = 1,...,o \quad (9)$$

where $S^i(b)$ is the circular right shift of the vector b by i-th positions.

In this manner, they could reduce the size of the UOV scheme public key.

III. The Blum Blum Shub Generator

The Blum-Blum-Shub(BBS) pseudorandom generator is a very simple and strongly secure pseudorandom generator based on the hardness of integer factoring[8][9]. Let Blum prime number is a prime number p with $p \equiv 3 (\mathrm{mod}\, 4)$. Let m is a product of two big Blum prime numbers p and q. Choose the random seed s where $s \in_R [1, m-1]$. Then we can get the initial value x_0 by

$$x_0 \equiv s^2 (\mathrm{mod}\, m) \quad (10)$$

Then we can define the sequence.

$$x_{i+1} \equiv x_i^2 (\mathrm{mod}\ m) \quad (11)$$

Even parity bit, odd parity bit or least significant bit can be the output of the Blum Blum Shub pseudorandom generator. In our paper, we use the least significant bit.

Algorithm 1: The BBS algorithm

$$Input: s$$
$$x = s^2 \bmod m$$
$$for\ i = 0,...do$$
$$x_{i+1} = x_i^2 \bmod m$$
$$b_i = x \bmod 2$$
$$b = b + b_2 \times 2$$
$$end\ for$$
$$Output: b$$

Alg. 1 shows the BBS algorithm. The input s is a random seed and the output b is a random value which is composed of a series of the random bits. For example, let p = 31, q = 23 and s = 4. Then the BBS generator creates the sequence $x_1, x_2, x_3, \cdots, x_8$ = 16, 256, 653, 35, 512, 473, 560, 593. Therefore the random bits are 0, 1, 1, 0, 1, 0, 1, 0 and the output is 86.

IV. UOV Scheme with PRNG

As we explained in Section I, reducing the key size is a hot research topic. If the UOV scheme is applied to the practical networks, because all clients must have the private key for generating a signature basically, reducing the private key size would be very important. We generate the private key by BBS pseudorandom generator which is considered as a secure pseudorandom number generator [9].

In this section, we explain how to apply PRNG to the UOV scheme.

A. Construction

Let K be a finite field. Let v be a number of the vinegar variables and o be a number of the oil variables. Let p and q are Blum prime numbers. Let n be a total number of variables and set $n = o + v$.

We denote $u = (v \cdot (v+1))/2 + o \cdot v + n + 1$ and $m = p * q$. Alg. 1(PrKG, Private Key Generation) is used in generating a private key. Now we can generate the private key by

$$q_{i,j+1} = \Pr KG(q_{i,j}) \quad (1 \le i \le o,\ 1 \le j \le u) \quad (12)$$

We are able to get an initial value $q_{1,1}$ by

$$q_{1,1} = \Pr KG(r) \quad (13)$$

where $r \in_R [1, m-1]$ is a random seed.

Now we are able to construct matrix Q as the private key by virtue of (12).

$$\mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1u} \\ q_{21} & q_{22} & \cdots & q_{2u} \\ \vdots & & & \vdots \\ q_{O1} & q_{O2} & \cdots & q_{OU} \end{bmatrix} \quad (14)$$

Also, we can obtain $quadQ$ and $linQ$. We denote $H = [quadP \mid oilP]$. Because there is no correlation between H and $linP$, we can get the public key respectively[4]. To obtain H, we define a matrix $A' \in K^{LXL'}$ like (7).

A' = $\left(\alpha_{kl}^{ij}\right)$ $(1 \le k \le v,\ k \le l \le n$ for the rows, $1 \le i \le j \le n$ for the columns)

$$\mathbf{A}' = \begin{bmatrix} \alpha_{11}^{11} & \alpha_{11}^{12} & \cdots & \alpha_{11}^{nn} \\ \alpha_{12}^{11} & \alpha_{12}^{12} & \cdots & \alpha_{12}^{vn} \\ \vdots & & & \vdots \\ \alpha_{vn}^{11} & \alpha_{vn}^{12} & \cdots & \alpha_{vn}^{nn} \end{bmatrix} \quad (15)$$

We can easily get H by

$$H = quadQ \cdot \mathbf{A}' \quad (16)$$

$linP$ is also simply computed due to having the same dimension between $linQ$ and affine map S.

$$linP = linQ \cdot S \quad (17)$$

The public key will be $P = [H \mid linP]$

B. Key Generation Procedure

1) Compute the L, L' and u.
2) Generate the oXu matrix $\mathbf{Q}$ as the private key following (12)
3) Obtain $quad\mathbf{Q}$ and $lin\mathbf{Q}$.
4) Choose an $n \times n$ affine matrix S at random over the field. If it is note invertible, choose again.
5) Compute a LXL' matrix $\mathbf{A}'$ following (15).
6) Compute H and $linP$ following (16) and (17).
7) Obtain the public key by $P = [H \mid linP]$.

- Signature Generation. The private key generation process is required prior to sign the message. The signature generation procedure is almost same with the basic UOV's in subsection 2.1 except adding the private key generation process.
- Signature Verification. There is no difference with the basic UOV's in subsection 2.1.

Key Size. Let $K = GF(2^m)$. The public key size is $\frac{o \times (L'+n+1) \times m}{8} byte$, and the size of the private key is only $\frac{(3+n^2+n) \times m}{8} byte$

3 is come from a, b and r in (11) and (12). $n^2 + n$ is for the affine map *S*.

V. Practical Experiment

We compute the key size in each scheme following presented (o, v) in the Table I. We assume $K = GF(2^8)$. In [8], the basic UOV scheme with *o*=24, *v*=48 is considered to be secure. Therefore we begin with *o*=25, *v*=50. Table I shows the result of the measurement. In the UOV scheme with a PRNG, the private key size was reduced 94.79%.

TABLE I: Average Reduction Rate in Each Scheme

(o,v)	Key size(kB)	Basic UOV	UOV (PRNG)
(25,50)	Public key size	73.15	73.15
	Private key size	70.72	5.70
(30,60)	Public key size	125.58	125.58
	Private key size	119.82	8.19
(35,70)	Public key size	198.49	198.49
	Private key size	187.57	11.13
(40,80)	Public key size	295.24	295.24
	Private key size	276.96	14.52
(45,90)	Public key size	419.22	419.22
	Private key size	391.01	18.36
(50,100)	Public key size	573.80	573.80
	Private key size	532.70	22.65
(55,110)	Public key size	762.36	762.36
	Private key size	705.05	27.93
(60,120)	Public key size	988.26	988.26
	Private key size	911.04	32.58
Average Reduction rate(%)	**Public key size**	-	-
	Private key size	-	**94.79**

The average reduction rate will go up if the sizes of *o* and *v* are longer than before (Fig. 3 and Fig. 4).

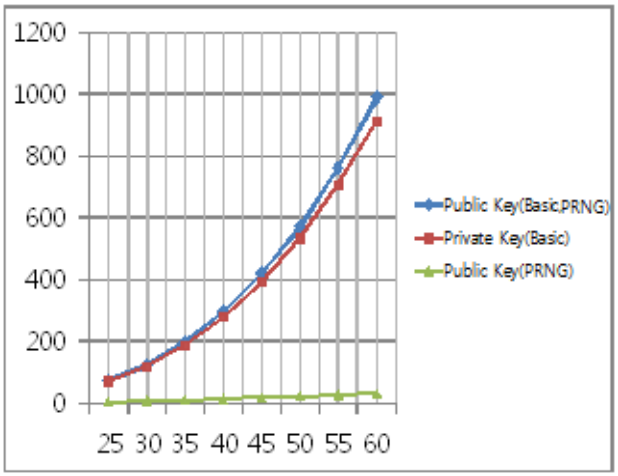

Fig. 3. This figure shows change of the public key and private key size rate on increasing a number of the oil variables. The x-axis is a number of the oil variables and the y-axis is the key size(kB).

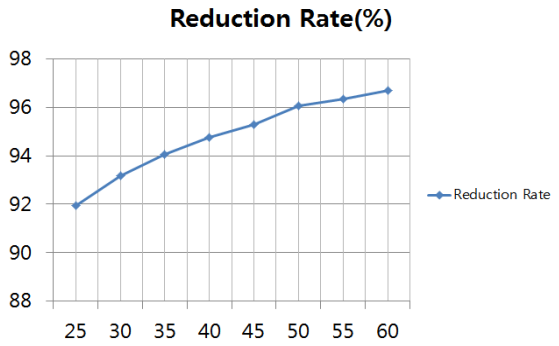

Fig. 4. This figure shows change of the reduction rate on increasing a number of the oil variables. The x-axis is a number of the oil variables and the y-axis is the reduction rate (%).

VI. Consideration

By the practical experiments, we got to know almost 95% of the private key memory can be saved. However, although the reduction rate for saving a memory is wonderful, a fundamental problem is still remained. Because when we make a signature for certain data, we have to generate a private key. And a necessary memory size is the same as before. If we generate both the private key and the signature on the fly, we can save the memory as much as we expected.

VII. Conclusion

In this paper, we reviewed the multivariate quadratic schemes such as the basic UOV scheme and the cyclic UOV scheme. And we applied BBS generator to the UOV scheme and measured that how much a private key memory size can be reduced. We confirmed that 94.79% of the private key memory size was saved.

However, there is still problem. We saved the memory but the private key size was not changed. Therefore our next research topic is adapting a PRNG which generate the private key on the fly for UOV scheme. We expect our study could contribute to solve the key size issue in MQ scheme.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No.2010-0026621).

References

[1] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. 35nd Annual Symposium on Foundations of Computer Science*, S. Goldwasser, ed.., IEEE Computer Society Press, 1994, pp. 124-134.

[2] A. Kipnis, J. Patarin, and L. Goubin, "Unbalanced oil and vinegar signature schemes," *Advances in Cryptology — EUROCRYPT 1999, Lecture Notes in Computer Science*, vol. 1592, pp. 206-222, Springer, 1999.

[3] T. Matsumoto and H. Imai. "Public quadratic polynomial-tuples for efficient signature verification and message-encryption," *Advances in Cryptology — EUROCRYPT 1988, Lecture Notes in Computer Science*, vol. 330, pp. 419-545, 1988.

[4] J. Patarin. "Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): Two new families of asymmetric algorithms," *Advances in Cryptology — EUROCRYPT 1996, Lecture Notes in Computer Science*, vol. 1070, pp. 33-48, Springer, 1996.

[5] C. Wolf. (2006). Introduction to multivariate quadratic public key systems and their applications. [Online]. Available: http://www5.rz.rub.de

[6] J. Patarin, "The oil and vinegar signature scheme," *Dagstuhl Workshop on Cryptography*, 1997.

[7] A. Kipnis and A. Shamir, "Cryptanalysis of the oil and vinegar signature scheme," in *Proc. CRYPTO'98, LNCS*, Springer, vol. 1462, 1998, pp. 257-266.

[8] A. Petzoldt, S. Bulygin, and J. Buchmann, "A multivariate signature scheme with a partially cyclic public key," in *Proc. SCC 2010*, 2010, pp. 229-235.

[9] L. Blum, M. Blum, and M. Shub, "A simple unpredictable pseudo-random number generator," *SIAM Journal on Computing*, vol. 15, pp. 364-383, May 1986.

Jihyun Kim received the BSEE degree from Pusan National University, Pusan, Republic of Korea in 2010, and he received the MS degree program in department of Computer Engineering at Pusan National University in 2012. He is Ph. D course in same major and university. His research interests include multivariate quadratic schemes, sensor networks, smart grid and malware protection systems.

Howon Kim He received the BSEE degree from Kyungpook National University, Daegu, Republic of Korea, in 1993 and the MS and PhD degrees in electronic and electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, Republic of Korea, in 1995 and 1999, respectively. From July 2003 to June 2004, he studied with the COSY group at the Ruhr-University of Bochum, Germany. He was a senior member of the technical staff at the Electronics and Telecommunications Research Institute (ETRI), Daejeon, Republic of Korea. He is currently working as an associate professor with the Department of Computer Engineering, School of Computer Science and Engineering, Pusan National University, Busan, Republic of Korea. His research interests include RFID technology, sensor networks, information security, and computer architecture. Currently, his main research focus is on mobile RFID technology and sensor networks, public key cryptosystems, and their security issues. He is a member of the IEEE, and the International Association for Cryptologic Research (IACR).