Public Key Cryptosystems Constructed Based on Random Pseudo Cyclic Codes, K(IX)SE(1)PKC, Realizing Coding Rate of Exactly 1.0

Masao KASAHARA
Faculty of Informatics, Osaka Gakuin University, Suita-shi, 564-8511 Japan.
kasahara@ogu.ac.jp

Abstract

In this paper, we present a new class of public-key cryptosystems, K(IX)SE(1)PKC realizing the coding rate of exactly 1.0, based on random pseudo cyclic codes. We show that K(IX)SE(1)PKC is secure against the various attacks including the attack based on the Gröbner bases calculation (GB attack).

Keyword

Public key cryptosystem, Error-correcting code, Code based PKC, Multivariate PKC, Gröbner bases, PQC.

1 Introduction

Most of the multivariate PKC’s are constructed by the simultaneous equations of degree larger than or equal to 2 [1]∼[6]. The present author recently proposed several classes of multivariate PKC’s that are constructed by many sets of linear equations[7]∼[13].

It should be noted that McEliece PKC[14] can be regarded as the first member of the class of the linear multivariate PKC.

In this paper, we present a new class of public key cryptosystem, K(IX)SE(1)PKC based on pseudo cyclic codes, realizing the coding rate of exactly 1.0. We show that K(IX)SE(1)PKC is secure against the attacks including the attack based on the Gröbner bases calculation (GB attack)[15].

Throughout this paper, when the variable \( v_i \) takes on a value \( \tilde{v}_i \), we shall denote the corresponding vector \( \mathbf{v} = (v_1, v_2, \cdots, v_n) \) as

\[
\mathbf{\tilde{v}} = (\tilde{v}_1, \tilde{v}_2, \cdots, \tilde{v}_n).
\] (1)

The vector \( \mathbf{v} = (v_1, v_2, \cdots, v_n) \) will be represented by the polynomial as

\[
v(x) = v_1 + v_2 x + \cdots + v_n x^{n-1}.
\] (2)

The \( \tilde{u}, \tilde{u}(x) \) et al. will be defined in a similar manner.

2 K(IX)SE(1)PKC over \( \mathbb{F}_{2^m} \)

2.1 Construction

Let us define a few symbols.

\[
\begin{align*}
G(x): & \text{ Random polynomial for generating random pseudo cyclic code over } \mathbb{F}_{2^m}, \\
& R_0 + R_1 x + \cdots + R_{g-1} x^{g-1} + R_g x^g, \\
& \text{where } R_i (i = 1, \cdots, g - 1), R_0 \neq 0 \\
& \text{and } R_g \neq 0 \text{ take on an element of } \mathbb{F}_{2^m} \\
e_Y : & \text{ Exponent (period, order) of } Y(x). \\
\sharp \{A_i\} : & \text{ Order of the set } \{A_i\}. \\
H(A_i) & \text{ Ambiguity of } A_i, \log_2 \sharp \{A_i\} \text{ (bit).} \\
H(A|B) & \text{ Ambiguity (Conditional Entropy) of } A \text{ when } B \text{ is given (bit).} \\
[R_{ij}]_{a \times b} : & \text{ Random matrix, where } R_{ij} (i = 1, \cdots, a; j = 1, \cdots, b) \text{ takes on 0 or 1} \\
& \text{equally likely in a random manner.} \\
H([R_{ij}]_{a \times b}) & \text{ Ambiguity of } [R_{ij}]_{a \times b} \cong ab \text{ (bit).} \\
C & \text{ Ciphertext, } (C_I, C_H). \\
C_I & \text{ First ciphertext.} \\
C_H & \text{ Second ciphertext.} \\
N_V & \text{ Total number of variables.} \\
N_E & \text{ Total number of equations.}
\end{align*}
\]

Let the message vector \( \mathbf{A} \) over \( \mathbb{F}_2 \) be represented by

\[
\mathbf{A} = (A_1, A_2, \cdots, A_N).
\] (3)

Throughout this paper we assume that the messages \( A_1, A_2, \cdots, A_N \) are mutually independent and equally likely. Let \( \mathbf{A} \) be transformed into

\[
\mathbf{A} \cdot H_I = \mathbf{a} = (a_1, a_2, \cdots, a_N),
\] (4)

where \( H_I \) is an \( N \times N \) non-singular random matrix over \( \mathbb{F}_2 \).

Let \( \mathbf{a} \) be partitioned into

\[
\mathbf{a} = (m_1, m_2, \cdots, m_n),
\] (5)

where \( m_i \) is given by

\[
m_i = (a_{i1}, a_{i2}, \cdots, a_{im}).
\] (6)
In the followings let us regard \( m_i \) as an element of \( \mathbb{F}_2^m \).
Let us partition the components of \( \mathbf{a} \) into
\[
\mathbf{m}_A = (m_{g+1}, m_{g+2}, \ldots, m_{g+f}),
\]
\[
\mathbf{m}_B = (m_{g+f+1}, m_{g+f+2}, \ldots, m_n),
\]
and
\[
\mathbf{m}_C = (m_1, m_2, \ldots, m_g),
\]
respectively, where \( n \) is given by
\[
n = g + 2f.
\]
We let \( f \) be given by
\[
g < f,
\]
due to the reason mentioned in 2.4.
From \( \mathbf{m}_A \) and \( \mathbf{m}_B \), we obtain
\[
(m_A(x)m_B(x))^\alpha \equiv p(x) \mod P(x),
\]
where \( P(x) \) is a primitive polynomial of degree \( f \) over \( \mathbb{F}_2^m \), and \( \alpha \) is given by
\[
\alpha = 1 + 2 + 2^2 + \cdots + 2^f < 2^m - 1.
\]
Let \( p(x) \) be represented by
\[
\mathbf{p} = (p_1, p_2, \ldots, p_f).
\]
The first ciphertext \( C_I(x) \) is given by
\[
C_I(x) = p(x).
\]

**Remark 1:** All the components of \( \mathbf{p} \) are calculated at the sending end from Eq.(12), for the given \( \mathbf{m}_A \) and \( \mathbf{m}_B \). Namely all the components are not represented by a set of equations of degree \( B \).

Regarding \( \mathbf{m}_A \) over \( \mathbb{F}_2^m \) as an \( mf \)-tuple over \( \mathbb{F}_2 \), \( \mathbf{m}_A \) is transformed into
\[
\mathbf{m}_A H_B = \mathbf{m}'_A = (m'_{g+1}, m'_{g+2}, \ldots, m'_{g+f}),
\]
where \( H_B \) is an \( mf \times mf \) non-singular random matrix over \( \mathbb{F}_{2^m} \).
It should be noted that any component \( m'_i \) of \( \mathbf{m}'_A \) is an element of \( \mathbb{F}_{2^m} \).
Let \( r(x) \) be given by
\[
r'_i(x)x^g \equiv r(x) \mod G(x)
\]
\[
= r_1 + r_2 x + \cdots + r_g x^{g-1}.
\]
The code word, \( w(x) \), generated by the generator polynomial \( G(x) \), can be represented by
\[
w(x) = r(x) + m'_A(x)x^g.
\]
Regarding the vector \( \mathbf{r} = (r_1, r_2, \ldots, r_g) \) over \( \mathbb{F}_{2^m} \) as a \( gm \)-tuple over \( \mathbb{F}_2 \), it is transformed into
\[
(r_1, r_2, \ldots, r_g) H_{III} = \mathbf{t} = (t_1, t_2, \ldots, t_g),
\]
where \( H_{III} \) is a \( gm \times gm \) random non-singular matrix over \( \mathbb{F}_2 \).
We see that the ambiguity of \( H_{III} \) over \( \mathbb{F}_2 \) is given approximately by
\[
|H_{III}| \simeq g^2 m^2 \quad (\text{bit}),
\]
an extremely large value for \( gm \gtrsim 80 \).
According to the transformation given by Eq.(19), the code word \( w(x) \) is transformed into
\[
w'(x) = t(x) + m'_A(x)x^g \neq 0 \mod G(x),
\]
for \( r(x) \neq 0 \).
The \( w'(x) \) is publicized.
At the sending end the message vector \( \mathbf{m}_C \) is transformed into
\[
\{m_C(x)\}^3 = \tau(x).
\]

**Remark 2:** All the components of \( \mathbf{\tau} \) are calculated at the sending end from Eq.(23), for the given \( m_C(x) \). Namely all the components of \( \mathbf{\tau} \) are not given by a set of quadratic equations.

With this \( \tau(x) \), at the sending end, the word \( u(x) \) is constructed by
\[
u(x) = w'(x) + \tau(x)x^g.
\]
The second ciphertext \( C_H(x) \) is given by
\[
C_H(x) = u(x).
\]
We have the following set of keys.

| Public key | \( \mathbf{m}_A, \mathbf{m}_B, \mathbf{m}_C, \mathbf{w}', P(x), \alpha, g, 3 \) |
| Secret key | \( H_I, H_{II}, H_{III}, G(x) \) |

### 2.2 Encryption and Decryption

#### [Encryption]

**Step 1:** The vector \( \tilde{\mathbf{p}} \) is calculated from Eq.(12) for the given \( \mathbf{m}_A \) and \( \mathbf{m}_B \).

**Step 2:** The ciphertext \( \tilde{C}_I(x) \) is given by \( \tilde{p}(x) \) from Eq.(15).

**Step 3:** The \( w'(x) \) is calculated from Eq.(21).

**Step 4:** Given \( \tilde{m}_C(x) \), the \( \tilde{\tau}(x) \) is calculated from Eq.(23).

**Step 5:** The ciphertext \( \tilde{C}_H(x) \) is given by \( \tilde{u}(x) = w'(x) + \tilde{m}_C(x)x^g \) from Eqs.(24) and (25).
Step 1: The $\tilde{r}(x)$ is inverse transformed to $\tilde{r}(x)$ by $\tilde{r} \cdot H^{-1}_1$, yielding $\tilde{u}(x) + \tilde{m}_C(x)x^d$.

Step 2: The $\tilde{m}_C(x)$ is decoded by
\[ \tilde{C}_B(x) = \{ \tilde{w}(x) + \tilde{m}_C(x) \}^d \equiv \tilde{m}_C(x) \mod G(x), \] \[ \text{(26)} \]
where $d$ is the inverse element of $3$ modulo $e_G$, yielding $\tilde{w}(x)$.

Step 3: From $\tilde{w}(x)$, the transformed message $\tilde{m}_A'(x)$ is decoded.

Step 4: The vector $\tilde{m}_A$ is obtained by $\tilde{m}_A'H^{-1}_1$.

Step 5: Letting $e_P$ be the period of $P(x)$, the message $\tilde{m}_B(x)$ is obtained by
\[ \tilde{m}_B(x) \equiv \tilde{p}(x)^{\beta} \tilde{m}_A'(x) \mod P(x), \] \[ \text{(27)} \]
where $\beta$ is given by
\[ a \beta \equiv 1 \mod e_P. \] \[ \text{(28)} \]

Step 6: From $\tilde{m}_A$, $\tilde{m}_B$ and $\tilde{m}_C$, the original message, $\tilde{A}$, is decoded by
\[ (\tilde{m}_A, \tilde{m}_B, \tilde{m}_C)H^{-1}_B = \tilde{A} \equiv (\tilde{A}_1, \tilde{A}_2, \ldots, \tilde{A}_N). \] \[ \text{(29)} \]

### 2.3 Examples

In Table 1, we present two examples of K(X)SE(1)PKC over $\mathbb{F}_{2^m}$.

Let us show a schematic diagram of Example II.

Let us discuss on several possible attacks on K(X)SE(1)PKC.

**Attack 1: Exhaustive attack on $G(x)$ over $\mathbb{F}_{2^m}$**

The generator polynomial $G(x)$ can be represented by
\[ G(x) = R_0 + R_1x + \cdots + R_{g-1}x^{g-1} + R_gx^g, \] \[ \text{(35)} \]
where we assume that $R_i(i = 0, 1, \cdots, g)$ takes on an element of $\mathbb{F}_{2^m}$ equally likely except that $R_0$ and $R_g$ are required to be nonzero element.

As a result, the probability of estimating $G(x)$ correctly in an exhaustive manner, $P_C[\tilde{G}(x)]$, is given by
\[ P_C[\tilde{G}(x)] = (2^m - 1)^{-2} \cdot 2^{-(g-1)m} \] \[ \approx 2^{-(g+1)m}. \] \[ \text{(36)} \]
Letting $m$ and $g$ satisfy
\[(g+1)m \gtrsim 80,\] (37)
the probability $P_C[\hat{G}(x)]$ is given by
\[P_C[\hat{G}(x)] \lesssim 2^{-80},\] (38)
a sufficiently small value.

For example, $P_C[\hat{G}(x)]$’s are given by $2^{-128} = 2.94 \times 10^{-39}$ for Example I and $2^{-192} = 2.76 \times 10^{-60}$ for Example II in Table 1, extremely large values. We see that K(IX)SE(1)PKC would be secure against the Attack 1 provided that Eq. (37) is satisfied.

\section*{Attack 2: Attack on $m'_A$ based on $t$}

The $t$ is given by a linear transformation of $r$ and is given as it is in word $u$. In order to be secure against the Attack 2, the relation, $g < f$ (Eq. (11)), should be strictly satisfied. The conditional entropy, $H(m'_A|t)$ is given by
\[H(m'_A|t) = (f - g)m \text{ (bit)},\] (39)

For examples I and II in Table 1, the conditional entropy $H(m'_A|t)$ is given by
\[H(m'_A|t) = 128 \text{ (bit)}\] (40)
a sufficiently large value.

It is easy to see that once $m'_A$ is disclosed, $m'_C(x)$ is disclosed.

We conclude that K(IX)SE(1)PKC is secure against Attack 2. □

\section*{Attack 3: Attack on $m_B(x)$ by estimating $m_A(x)$}

We assume here that Eq. (30) holds, namely $d_A = d_B$. By estimating $m_A(x)$ in an exhaustive manner for a given $p(x)$, $m_B(x)$ can be disclosed. The probability of disclosing $m_B(x)$ by estimating $m_A(x)$ is given by
\[P_C[\hat{m}_B(x)] = 2^{-(d_A+1)m}.\] (41)

For examples I and II, the probability $P_C[\hat{m}_B(x)]$’s are given by $2^{-7432} = 3.71 \times 10^{-68}$ and $2^{-4496} = 2.54 \times 10^{-116}$ respectively, extremely small values.

We see that K(IX)SE(1)PKC is secure against Attack 3. □

\section*{Attack 4: GB attack on the ciphertext}

The ciphertext $C_B(x)$ can be represented by a set of simultaneous equations of degree $B$ in the variables $A_1, A_2, \cdots, A_N$. The ciphertext $C_B(x)$ can be represented by a set of linear and quadratic equations in the variables $A_1, A_2, \cdots, A_N$. Namely the GB attack should solve the following sets of simultaneous equations.

\section*{SE(II) : The $g m$ linear equations and $f m$ quadratic equations in the variables $A_1, A_2, \cdots, A_N$.}

The degree $B$ takes on 223 for Example I and, 255 for Example II, in Table 1, extremely large values. The number of variables $N$ takes on 544 for Example I and, 640 for Example II, also large values.

We conclude that K(IX)SE(1)PKC is secure against Attack 4. □

\section*{3 Conclusion}

In this paper we have presented K(IX)SE(1)PKC based on random pseudo cyclic codes. We have shown that our proposed K(IX)SE(1)PKC can be made sufficiently secure against the various attacks including the attack based on the Gröbner bases calculation.

\section*{References}


[12] M. Kasahara: “New Class of Public Key Cryptosystem Constructed Based on Pseudo Cyclic Codes over $F_2$ and over $F_{2^m}$ ($m \geq 7$) Realizing Coding Rate of 1.0”, SITA, (2010-12).

