Comparison between NTRU, Polynomial RSA, and PH-RSA

Banen Najah Abass$^1*$ and Hassan Rashed Yassein$^2$

$^1$Department of Mathematics, Faculty of Education for Girls, University of Kufa, Al-Najaf, Iraq
$^2$Department of mathematics, College of Education, University of Al-Qadisiyah, Al-Qadisiyah, Iraq

Abstract. It has become necessary to find new encryption methods that guarantee data integrity as a result of the advancement of technology. For this reason, we presented in this research a comparison between the three cryptosystems: NTRU, polynomial RSA, and PH-RSA in terms of the level of security so that it is easier for the user to choose the appropriate method according to the nature of the transmitted data.

1 Introduction

As a result of the wide application of the Internet in various fields of life as well as the continuous development of technology, most of whose data is stored on the Internet, information has become more vulnerable to hacking. As a result, we constantly need to be aware of the most effective encryption methods in order to make it difficult for hackers to penetrate our data.

The RSA encryption technique, which was proposed by Rivest et al. in 1979 is considered to be one of the public-key encryption methods primarily depends on the utilization of parameters derived from prime integers [1]. El Gamal was first suggested as a cryptographic algorithm by ElGamal in 1985 [2], it was based on the Diffie-Hellman key exchange [3]. Hoffstein et al. [5] founded NTRU in 1996 using a truncated polynomial ring as their foundation. Because of its many benefits, NTRU is an active choice for a wide range of applications, which encourages academics to work on further developing it. In 2012, Ishwary and Kumar introduced a number of enhancements to the protection of sensitive information by utilizing the RSA algorithm [5]. In 2015, Gafitoiu put out the idea of a polynomial RSA, which would replace integers with polynomials [6]. Yassein and Al-Saidi presented theHXDTRU cryptosystem, which is based on hexadecnnion algebra [7], in the year 2016. Al-Saidi and Yassin were also introduced in 2021 as a new alternative to the NTRU cryptosystem based on highly dimensional algebra with dense lattice structures [8]. In the same year, Yassin and others presented a new design of NTRU encryption with improved security and performance [9]. In 2023 Atea and Yassein presented an encryption method based on linking the concepts of NTRU and modified RSA using a polynomial ring $Z[x]/N(x)$ by creating two public keys [10]. In 2024, Abbas and Yassein introduced a new polynomial hexadecimal RSA (PH-RSA) encryption scheme by mixing NTRU and RSA.

* Corresponding author: Banenn.alkuzai@uokufa.edu.iq

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).
polynomial with high security [11]. In this paper, we compare NTRU, polynomial RSA, and PH-RSA cryptosystems in terms of security.

2 NTRU Cryptosystem

NTRU public key cryptosystem is based on a ring truncated polynomials of degree $N-1$, which is as follows

$$c_0 + c_1 x + c_2 x^2 + \cdots + c_{n-2} x^{N-2} + c_{N-1} x^{N-1}$$

In NTRU, The basic operations take place in the ring $K = \mathbb{Z}[x] / (x^N - 1)$ can be represented in modulo $p$ and $q$ by

$$K_q = \mathbb{Z}_p[x] / (x^N - 1)$$

and

$$K_q = \mathbb{Z}_q[x] / (x^N - 1)$$

respectivey.

Now, let $F, G$ are two truncated polynomials in $K$ such that

$$F = \sum_{i=0}^{N-1} f_i x^i$$

and

$$G = \sum_{i=0}^{N-1} g_i x^i$$

The addition (+) and convolution multiplication($\circ$) defined as

$$F + G = \sum_{i=0}^{N-1} (f_i + g_i) x^i$$

and

$$U_{\beta} = F \circ G = \sum_{\beta} 1 f_j G_{\beta - 1} \sum_{i=0}^{N-1} f_i G_{N+i+1} = \sum I_{i+j=\beta \pmod{N}} F_i G_j$$

2.5.1 Public Parameters

The public key components in the NTRU are parameters used $\square, \square$, and $\square$ are positive integers. Note that $p$ and $q$ need not be prime, but we will $\gcd(p,q) = 1$ and $q$ will always be considerably larger than $p$ and four sets of polynomials of degree $N - 1$ with integer coefficients are defined as following

$$L_f = \{ f \in K | f \text{ has } d_f \text{ coefficients equal to } +1, d_f - 1 \text{ equal to } -1, \text{ the rest } 0 \}$$

$$L_g = \{ g \in K | g \text{ has } d_g \text{ coefficients equal to } +1, d_g \text{ equal to } -1, \text{ the rest } 0 \}$$

$$L_\phi = \{ \phi \in K | \phi \text{ has } d_\phi \text{ coefficients equal to } +1, d_\phi \text{ equal to } -1, \text{ the rest } 0 \}$$

$$L_m = \{ m \in K | m \text{ has coefficients lying between } -\frac{p}{2} \text{ and } \frac{p}{2} \}$$

Where $\ell_{(d_x, d_y)} = \{ f \in K | f \text{ has } d_x \text{ coefficients equal } 1, d_y \text{ coefficients equal } -1, \text{ the rest } 0 \}$.

I. Key Generation

By choosing $f, g$, from the sets $L_f, L_g$, we will build the public key is $H = F_q^{-1} \circ g \pmod{q}$, so $f, g$ are private keys in the NTRU cryptosystem. Notice that the multiplicative inverse of the polynomials $f$ is denoted by $F_p^{-1}$ and $F_q^{-1}$ such that $F_p^{-1} \ast f = I \pmod{p}$ and $F_q^{-1} \ast f = I \pmod{q}$.

II. Encryption
Using the following formula $E = p(H \ast r) + m \pmod{q}$ can be encrypted the plaintext $m$, where $r$ is a random polynomial in $L_r$, called the blinding polynomial and polynomial $m$ is a message in $L_m$.

III. Decryption

Once the message has been transmitted encrypted, the recipient decrypts it using the following stages:

1. Calculate $C = f \ast E \pmod{p}$
   
   $= f \ast (p(H \ast r) + m) \pmod{q}$
   
   $= p(f \ast F_q^{-1} \ast g \ast r) + f \ast m \pmod{q}$
   
   $= p(g \ast r) + f \ast m \pmod{q}$.

   The polynomial coefficients $p(g \ast r) + f \ast m \pmod{q}$ are in the interval $(-\frac{q}{2}, \frac{q}{2}]$, that it stably, if its coefficients are reduced to mod $q$.

2. Calculate $D = C \pmod{p}$
   
   $= p(g \circ r) + f \circ m \pmod{p}$
   
   $= f \circ m \pmod{p}$

3. Calculate $F_p^{-1} \circ D \pmod{p} = m \pmod{p}$.

The result should then be adjusted between the interval $(-\frac{p}{2}, \frac{p}{2}]$.

3 Polynomial RSA Cryptosystem

The public parameters here are represented subset $l_A$ and $l_B$ defined as the Table 1 below

| $l_A$ | $A(x) \in Z_p[X] \ | B(x)$ has $d_a$ coefficients equal 1, $d_a - 1$ coefficients equal $-1$, the rest 0 |
|-------|------------------------------------------------------------------------------------------------------------------|
| $l_B$ | $B(x) \in Z_p[X] \ | B(x)$ has $d_b$ coefficients equal 1, $d_b$ coefficients equal $-1$, the rest 0 |

Because it is a public-key encryption method, the algorithm for this method is represented by three stages, namely, encryption and decryption, preceded by the creation of the key[26,50].

I- Key generation

It takes the following steps:

- Step1: take two polynomials $A(x), B(x)$
- Step2: calculate $N(x) = A(x) \times B(x)$
- Step3: find $S$ the number of invertible polynomial modulo $N(x)$

And $S = (p^m - 1)(p^n - 1)$ where $n, m$ is the degree of the polynomial $A(x), B(x)$ respectively

- Step4: choose $E$ such that $0 < E < S$ and gcd$(E, S) = 1$
- Step5: find $D$ invers of $E$, $DE \equiv 1 \pmod{S}$
Private keys \((A(x), B(x), D(x))\) and public keys \((N(x), E(x))\).

II - Encryption
Let us need to encrypt the message \(M(x)\) then the cipher text take the form \(C(x) = [M(x)]^E \mod N(x)\).

III - Decryption
After receiving the encrypted messages from the sender, the original message can be obtained by
\[
M(x) = (C(x))^D \mod N(x) = ([M(x)]^E)^D \mod N(x) = ([M(x)]^{sk+1}) \mod N(x) = [M(x)]^{sk} M(x) \mod N(x) = M(x) \mod N(x).
\]

4 PH-RSA Cryptosystem

The PH-RSA encryption system depends on polynomial RSA, but the polynomial ring \(\mathbb{Z}_p[x]\) is replaced by hexadecion algebra HD and the subsets \(L_F\) and \(L_G\) are defined as in (3). The cryptosystem phases of PH-RSA are as follows.

The PH-RSA encryption system consists of three phases: key generation, encryption and decryption.

1- Key Generate
Choose \(P(x), Q(x) \in HD\) when \(P(x) = f_0(x) + \sum_{i=1}^{15} f_i(x)x_i\) and \(Q(x) = g_0(x) + \sum_{i=1}^{15} g_i(x)x_i\),
Take \(R = HD/\langle N(x) \rangle\).
Choose \(e \in \mathbb{Z}_s = \{0,1,2,\ldots,s-1\}\) such that \(gcd(e,s) = 1\).
Find \(d \in \mathbb{Z}_s\).

2- Encryption: Plaintext is encrypted using the formula
\[
C(x) = \left[ m_0(x) + \sum_{i=1}^{15} m_i(x)x_i \right]^e \mod N(x).
\]

3- Decryption: The plaintext is restored using formula
\[
C[x]^d \mod N(x) = C[x]^d = \left[ m_0(x) + \sum_{i=1}^{15} m_i(x)x_i \right]^{ed} \mod N(x) = \left[ m_0(x) + \sum_{i=1}^{15} m_i(x)x_i \right]^{sk+1} \mod N(x) = \left[ (m_0(x) + \sum_{i=1}^{15} m_i(x)x_i)^{sk} \right] \cdot \left[ m_0(x) + \sum_{i=1}^{15} m_i(x)x_i \right] \mod N(x) = \left[ m_0(x) + \sum_{i=1}^{15} m_i(x)x_i \right] \mod N(x).
\]
5 Comparison Between NTRU, polynomial RSA, and PH-RSA

The space security and algebraic structure of NTRU, polynomial RSA, and PH-RSA are shown in Table 2.

Table 2. NTRU, polynomial RSA, and PH-RAS key spaces

<table>
<thead>
<tr>
<th></th>
<th>Algebraic structure</th>
<th>Space security</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polynomial RSA</td>
<td>$Z_p[x] /&lt; N(x) &gt;$</td>
<td>$(n_1!) \over (d_1^2)! (n_2-2d_1)!$ or $(n_2!) \over (d_1^2)! (n_2-2d_2)!$</td>
</tr>
<tr>
<td>PH-RSA</td>
<td>$H /&lt; N(x) &gt;$</td>
<td>$(n_1!) \over (d_1^2)! (n_2-2d_1)!)^16$ or $(n_2!) \over (d_1^2)! (n_2-2d_2)!)^16$</td>
</tr>
<tr>
<td>NTRU</td>
<td>$Z[x] / (x^N - 1)$</td>
<td>$(N!) \over (d_1^2)! (N-2d_2)!$</td>
</tr>
</tbody>
</table>

6 Conclusion

In this paper, we compared the NTRU encryption method with the polynomial RSA encryption system as well as the PH-RSA encryption method. The fact that the PH-RSA cryptosystem is more secure in comparison to the NTRU and polynomial RSA cryptosystems. This makes it harder for a hacker to readily enter the data because it takes a lengthy time for them to do so.

References