Asymmetric Cryptographic Protocol with modified Approach

Dr. Pushpa.R.Suri† and Priti Puri††,

Department of Computer Science & Applications,Kurukshetra University,Kurukshetra,India

Summary

The paper starts with general introduction about the cryptography. Paper flows with the one upcoming method of public key cryptography as NTRU. The basic function of NTRU is given with previous work done. Some new modified approach to the basic NTRU method is proposed. In that approach, entire polynomial ring (a huge set of polynomials)divide into small subsets of polynomials. Different subsets of polynomials can be run concurrently to generate the keys for encryption and decryption. By this approach, many processes can work in parallel. At the same time more than one person can do encryption and decryption with different sections and the process will hidden from each other..

Key words:

Decryption, NTRU, polynomial, public key, XOR, private key.

1. Introduction

The increasing use of electronic means of data communications, coupled with the growth of computer usage, has extended the need to protect information or data security. In data security, one important branch is Cryptology. Cryptology is divided into two parts as Cryptography and Cryptanalysis. The art of cryptography as a means for protecting private information against unauthorized access is as old as writing itself. A cipher conceals the plaintext *M* by transforming it into a disguised form, called the cipher text C, so that only the authorized receiver can transform it back to the original plaintext. The process of transforming plaintext into cipher- text is called encryption or enciphering, and the inverse transformation from cipher text to plaintext is called decryption or deciphering.

1.1 Public Key Encryption/Asymmetric key Cryptography

Whitfield Diffie and Martin Hellman proposed Public Key Encryption[2] in 1976; by this method each user of the network has their own individual private key and a public key. The public key is distributed to all members of the network, while only the user holds the private key. A message encrypted with the public key of a person can only be decrypted with Private Key of the same person and vice-versa. Public Key - K_1, Private key- K_2, Message- M then

$$D_{k2} (E (_{k1} (M)) = D_{k1} (E_{k2} (M)) = M \quad (1)$$

1.2 NTRU Basic Algorithm

An NTRU [7] cryptosystem purposed by three mathematicians in 1996,depends on three public integer parameters (N, p, q) and four sets L_f ,L_g ,L_r ,L_x of polynomials of degree N-1 with integer coefficients. Here p and q need not be prime, but assume that gcd (p, q)=1, and q will always be considerably larger than p. This algorithm works in the ring R= Z [X]/(X^N-1) where Z represents the set of integers. An element F∈R will be written as a polynomial or a vector,

$$F = \sum_{i=0}^{N-1} F_i x_i = [F_0, F_1, \ldots\ldots, F_{N-1}] \quad (2)$$

*The multiplication is denoted by * in ring R.*

$$F*G=H \text{ with } H_k = \sum_{i-j\equiv k(\text{mod } N)} F_i G_j \quad (3)$$

By a multiplication modulo q, we mean to reduce the coefficients modulo q .It is fundamentally different from both RSA and elliptic curve cryptography, and it has some efficient advantages over them.

The NTRU algorithm is actually probabilistic in nature; i.e. there is a small chance of decryption failure. With the appropriate choice of parameters the decryption probability can be made to be on the order of 10-25 or less.

The "moderate security" version of NTRU that was presented at Crypto '96(with the above values for N, p, q) was broken by Coppersmith and Shamir [7], who used lattice-basis reduction methods [1] to find short vectors in a lattice that arises when one tries to find the plaintext from the NTRU cipher text and public key. Subsequently there have been other successful attacks on certain versions of NTRU [7] [6]. In response, the inventors of NTRU have adopted new parameters and padding schemes that they believe can resist all known attacks. They offer valuable cash prizes to anyone who can break their

Manuscript received April 5, 2007

Manuscript revised April 25, 2007

“challenges" with N-parameter equal to 251, 347 and 503. In 2001 an NTRU signature scheme was proposed at Euro crypt [5], but both that scheme and a revised version were broken soon after [4]. A new revised signature scheme is now available on the NTRU website, but at present the prospects for commercial adoption of an NTRU-based signature scheme are unclear.

2. Suggested Scheme

This paper has given some idea of utilizing the polynomial ring in different aspects. In this scheme, ring polynomial is divided into n sections. Each section has m random polynomials for key generation, encryption and decryption. For large random polynomial generation, each section has random number generator which will give the random coefficients of polynomials. These different sections will work simultaneously. At the same time, many polynomials from different sections are used and it can give the maximum utilization of polynomials of a ring and random number generator can handle the randomness of polynomials of each section. By using this concept, the speed of the algorithm can be increased and parallel processing can be possible. If n number of different sections is maintained for algorithm then n number of random number generators is also required.

2.1 Security of Suggested Scheme:

The multiple sections, n, work together and section numbers (to which section message will move) are hidden from senders. At the time of encryption, Sender, B_1, will encrypt the message with A_1’s public key. At decryption end, receiver, A_1, firstly XOR the section number from the encrypted message and then message will go to that section area. In that section, random polynomials are generated with the help of random number generator and message will decrypt by using the private key. Section number XORing will increase security to the basic approach as doubled because intruder should know the section number and the private key of the particular receiver to decrypt the message.

$A_1, A_2, \ldots, A_n$ Receivers.
$B_1, B_2, \ldots, B_n$ Senders
$1, 2, 3, \ldots, n$ Different Sections
$h_1, h_2, \ldots, h_n$ Public keys related to different sections
$e_1, e_2, \ldots, e_n$ Encrypted messages for different sections
$n_1, n_2, \ldots, n_n$ Different section numbers

Section 1.

A_1, public key ($h_1 \oplus n_1$) $\longrightarrow$ B_1

A_1 $\longleftarrow$ B_1, e_1

$A_1, n_1 \oplus e_1$ $\longleftarrow$ B_1

Section 2.

A_2, public key ($h_2 \oplus n_2$) $\longrightarrow$ B_2

A_2 $\longleftarrow$ B_2, e_2

$A_2, n_2 \oplus e_2$ $\longleftarrow$ B_2

....

....

Section n.

A_n, public key ($h_n \oplus n_n$) $\longrightarrow$ B_n

A_n $\longleftarrow$ B_n, e_n

$A_n, n_n \oplus e_n$ $\longleftarrow$ B_n

2.2 Key creation:

We can divide the whole ring R’s polynomials into n sections, R_1, $R_2 \ldots\ldots R_n$ and each section have at least m random polynomials and m >n, m will be greater than n. $q_1 \ldots q_n$ is large modulus to which each coefficient is reduced (Non-secret). $p_1 \ldots . p_n$ is small modulus to which each coefficient is reduced (Non-secret). $f_1 \ldots . f_n$ polynomial (private key). $g_1 \ldots .. g_n$ polynomial used to generate the public key $h_1 \ldots . h_n$ from $f_1 \ldots f_n$ (Secret but discarded after initial use)
$h_1 \ldots h_n$ polynomial (public key)
$r_1 \ldots r_n$ random “blinding” polynomial (Secret but discarded after initial use)

For first section:

A_1, user, chooses 2 random polynomials f_1 and g_1, that are in the defined ring R_1 as $f_1, g_1 \in L_{g1}$ and whose inverses exist in the ring modulo key parameters p_1 and q_1. These inverses are denoted F_{p1} and F_{q1} respectively, and need to be computed for the chosen private key f_1. If the inverse of either does not exist another f_1 is chosen and the process is repeated. That is,

$$F_{q1} * f_1 \equiv 1 (\mathrm{mod}\ q_1) \text{ and } F_{p1} * f_1 \equiv 1 (\mathrm{mod}\ p_1) \tag{4}$$

Next A_1 generates the public key as the polynomial h_1:

$$h_1 = \{f_{q1} * g_1 \oplus \text{section number}\} \mathrm{mod}\ q_1. \tag{5}$$

Section number is an integer, can be 1 2…. n and g_1 is also a randomly chosen polynomial in R_1. Each user’s encryption and decryption will be handled within that section only. Now XOR the section number with this public key h_1.

Similarly for other sections,

$F_{q2} * f_2 \equiv 1 (\mathrm{mod}\ q_2)$ and $F_{p2} * f_2 \equiv 1 (\mathrm{mod}\ p_2)$ (6)

$h_2 = \{f_{q2} * g_2 \oplus \text{section number}\}\ \mathrm{mod}\ q_2.$ (7)

……………………

……………………

$F_{qn} * f_n \equiv 1 (\mathrm{mod}\ q_n)$ and $F_{pn} * f_n \equiv 1 (\mathrm{mod}\ p_n).$ (8)

$h_n = \{f_{qn} * g_n \oplus \text{section number}\}\ \mathrm{mod}\ q_n.$ (9)

2.3 Encryption

For first section:
Sender B_1, wishing to send a binary message x from the set of plaintexts L_{x1} to A_1, begins by randomly choosing a polynomial r_1 that is in R_1. B_1 then performs the encryption by computing

$$e_1 \equiv p_1 r_1 * h_1 + x_1\ (\mathrm{mod}\ q_1). \quad (10)$$

Scalar multiplication of integer p_1 with polynomial r_1 is simply multiplication of each coefficient of r_1 with p_1. Multiplication of r_1 with h_1 is wrapped around the ring size**.**

Similarly for other sections:

$e_2 \equiv p_2 r_2 * h_2 + x_2\ (\mathrm{mod}\ q_2).$ (11)

………………………........

….………………………

$e_n \equiv p_n r_n h_n + x_n (\mathrm{mod}\ q_n)$ (12)

2.4 Decryption:

A_1 begins to decrypt the message e_1 by first XORing the section number

$$\{e_1 \oplus \text{section number}\}\mathrm{mod}\ q_1$$

$$\equiv [p_1 r_1 * \{h_1 \oplus \text{section number}\}\mathrm{mod}\ q_1 + x_1\ (\mathrm{mod}\ q_1)] \quad (13)$$

After getting the section number, the message will go to that particular section and decryption will done by particular private key of that section,

$$a_1 \equiv f_1 * e_1\ (\mathrm{mod}\ q_1) \quad (14)$$

and then reducing the coefficients of a_1 to be between $-q_1/2$ to $q_1/2$. Then he obtains the message x_1 by computing $F_{p1} * a_1$ (mod p_1), where F_{p1} is part of his private key and is the multiplicative inverse of f_1 mod p_1 as derived in the key generation.

$$\begin{aligned} a_1 &\equiv f_1 * e_1\ (\mathrm{mod}\ q_1), \\ &\equiv f_1 * p_1 r_1 * h_1 + f_1 * x_1\ (\mathrm{mod}\ q_1) \\ &\equiv f_1 * p_1 r_1 * F_{q1} * g_1 + f_1 * x_1 (\mathrm{mod}\ q_1) \\ &\equiv p_1 r_1 * g_1 + f_1 * x_1 (\mathrm{mod}\ q_1). \end{aligned} \quad (15)$$

This means that when a reduces the coefficients of $f_1 * e_1$ mod q_1 into the interval from $-q_1\backslash 2$ to $+q_1\backslash 2$,he recovers exactly the………. polynomial $a_1 = p_1 r_1 * g_1 + f_1 * x_1$.
Similarly for other sections,

$\{e_2 \oplus \text{section number}\}\mathrm{mod}\ q_2$

$\equiv [p_2 r_2 * \{h_2 \oplus \text{section number}\}\mathrm{mod}\ q_2 + x_2\ (\mathrm{mod}\ q_2)]$ (16)

$a_2 \equiv p_2 r_2 * g_2 + f_2 * x_2 (\mathrm{mod}\ q_2)$ (17)

………………………

……………………….

$\{e_n \oplus \text{section number}\}\mathrm{mod}\ q_n$

$\equiv [p_n r_n * \{h_n \oplus \text{section number}\}\mathrm{mod}\ q_n + x_n\ (\mathrm{mod}\ q_n)]$ (18)

$a_n \equiv f_n * e_n\ (\mathrm{mod}\ q_n)$ (19)

$\equiv p_n r_n * g_n + f_n * x_n (\mathrm{mod}\ q_n)$ (20)

This means that when a reduces the coefficients of $f_n * e_n$ mod q into the interval from $-q_n\backslash 2$ to $+q_n\backslash 2$,he recovers exactly the polynomial $a_n = p_n r_n * g_n + f_n * x_n$.

3. Comparison of RSA, ECC, NTRU, BRAIDGROUP SYSTEM

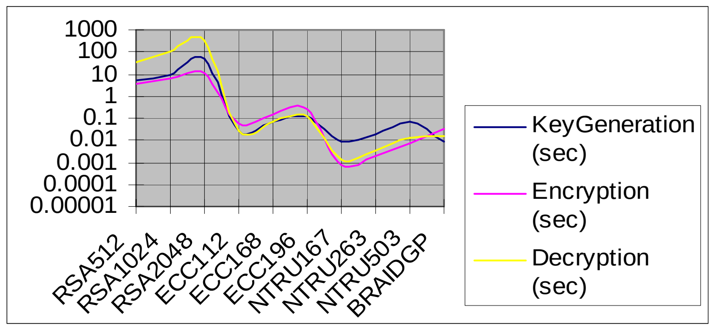

Figure 1.

Comparisons of four methods of public key cryptography, RSA [9], ECC [8], and NTRU & Braided Group Systems [10] are given with graph. ECC is mathematical based and more secure due to Elliptic curve discrete logarithm (ECDLP). The NTRU encryption and decryption are roughly two orders of magnitude faster than in ECC, with comparable security levels are used The NTRU keys are an order of magnitude bigger than ECC keys and NTRU message expansion is two times bigger than ECC using ElGamal scheme.

4. Conclusion

In this paper, we have given some introduction about public key cryptography with earlier work done in NTRU method. After that we have suggested a new scheme, some modification to the exiting algorithm for the multi users simultaneously. We can increase the security by XORing the section numbers. At the receiving end where receiver firstly has to give his section number then he can decrypt the message with his private key only. This is as to encrypt the private key and different sections can work together. Meet in the middle attack can be avoided but brute force attack can work. But due to section number still the security of messages can be doubled. Section numbers are XORed which can easily implemented on hardware also.

This scheme can be used in the organizations for security purposes or in security organizations where more than one user are receiving the different encrypted messages simultaneously and have to decrypt and send the encrypted messages again to different organizations. This scheme can work as warehouse to receive the messages from different places and send to different sections and workout simultaneously. The whole organization can be divided into the different sections and each user has his own section of NTRU and he can response to the sender without effecting the other persons work who are using the same algorithm with different sections of this scheme.

Case study

This application can be used in security related organization. We can make a system where sender, B, can take the public key of receiver, A, and encrypt the message and send it to the A, who is one part of organization. At receiving side, NTRU with sections is installed at the Server. Different section numbers are assigned to different clients. When message will receive at the server, section number is XORed with encrypted message and message will send to the concerning client, who are using the particular section of polynomial ring. With the help of section numbers, server will detect the particular client to whom this message will send. Then, client will use his private key to decrypt the message. For sending messages, client can use sender's public key, same procedure can repeat at sender level, B. We can divide the sections of NTRU according to number of clients.

References

[1] A. K. Lenstra, H. W. Lenstra, Jr. and L. Lovasz, Factoring polynomials with integer coefficients, Mathematics Annalen, 261 (1982), pp. 513-534.

[2] C. Gentry, J. Jonsson, M. Szydlo and J. Stern, Cryptanalysis of the NTRU signature scheme (NSS) from Euro crypt 2001, Advances in Cryptology ASIACRYPT 2001, Lecture Notes in Computer Science, 2248 (2001), Springer-Verlag, pp. 1-20.

[3] C. Gentry and M. Szydlo, Analysis of the revised NTRU signature schemeR-NSS, Advances in Cryptology , EUROCRYPT 2002, Lecture Notes in Computer Science, 2332 (2002), Springer-Verlag, pp. 299-320.Security and Privacy ACISP 2002, Lecture Notes in Computer Science, 2384 (2002), Springer Verlag, pp. 176-189.

[4] D. Coppersmith and A. Shamir, Lattice attacks on NTRU, Advances in Cryptology EUROCRYPT '97, Lecture Notes in Computer Science, 1233 (1997), Springer-Verlag, pp. 52-61.

[5] E. Jaulmes and A. Joux, A chosen cipher text attack against NTRU, Advances in Cryptology CRYPTO 2000, Lecture Notes in Computer Science, 1880 (2000), Springer-Verlag, pp. 20-35.

[6] J. Hoffstein, J. Pipher and J. Silverman, NSS: an NTRU lattice-based signature scheme, Advances in Cryptology, EUROCRYPT 2001, LNCS, 2045 (2001), Springer-Verlag, pp. 211-228.

[7] Karu and Loikkanen, Practical Comparison of Fast Public-key Cryptosystems, 2000

[8] V. Miller, Uses of elliptic curves in cryptography, advances in Cryptology CRYPTO '85, Lecture Notes in Computer Science, 218 (1986), Springer-Verlag, pp. 417-426

[9] http://www.rsasecurity.com/

[10] I. Anshel, M. Anshel and D. Goldfield, New key agreement protocol in braid group cryptography, Topics in Cryptography CT-RSA 2001, LNCS, 2020 (2001), Springer-Verlag, pp. 13-27

Dr. Pushpa Suri Phd, Senior Lecturer,, Department of Computer Science & application ,kurukshetra university, kurukshetra, India.

Priti Puri, M.tech,, Computer Science & Engineering, doing P.hd in Network Security & Cryptography, Department of Computer Science & Application ,kurukshetra university, kurukshetra,India.