International Journal of Computer Applications Technology and Research
Volume 2– Issue 4, 499 - 503, 2013, ISSN: 2319–8656

An Improved Post-Quantum Cryptographic Scheme Based on NTRU

Sachin Kumar
Delhi University
Delhi, India

Shobha
Delhi University
Delhi, India

Saibal K. Pal
DRDO, Metcalfe House, Delhi, India

Abstract: In this paper we report a more secure and efficient encryption algorithm based on the NTRU cryptographic scheme. NTRU is lattice based scheme resistant to quantum computing, hence it falls under the class of post quantum cryptosystems. It is based on shortest vector problem (svp)[9].The main characteristics of the system are low memory and low computational requirements but it provides high level of security. We present modifications in the NTRU scheme for making it more secure and efficient particularly for applications in wireless and constrained devices. In the original scheme, repetitions in the plaintext message lead to repetitions in the cipher text, which is a source of weakness in the system. To overcome this problem each byte of the input has been digested with different operations that produce different encrypted text even for repeated content of the Plain text message. The second modification is enhancing the public key scheme that makes this system more robust. These two modifications in the NTRU scheme makes it secure even for use in the Quantum Computing environment.

Keywords: Public-key cryptosystem, NTRU, polynomial inverse, convolution product, post-quantum cryptography

1. INTRODUCTION

Post quantum cryptography is a fascinating area of research challenge. In existence of Quantum computer post quantum cryptography will be critical for the future interest as it is well known that quantum computer may destroy RSA, DSA, and ECDSA. Quantum computers can potentially break most of conventional cryptosystems based on the integer factorization problem and discrete log problem which are actually deployed in practice at present. Certain classical cryptosystems inspired by computational problems of nature that entirely different from the integer factorization and discrete log are potentially much harder to solve, will remain unaffected by the threat of Quantum Computing. So those are called QUANTUM-RESISTANT or more clearly 'POSTQUANTUM' cryptosystems [1].We has some question to answer like-

- Is there any need to worry about the threat of quantum computers?
- Why should focus not continue on RSA or other resistant cryptosystem for classical computers?

Now suppose a situation when someone announces that quantum computer is no more a mystery means it is constructed then computers using crypto systems will be unsecure. In such case we need to have some crypto systems resistant to quantum attacks. The reasons to work on Post Quantum Cryptography are [2]-

- Time is required to improve the efficiency of post-quantum cryptography.
- Time is required to build confidence in post-quantum cryptography.
- Time is required to time to improve the usability of post-quantum cryptography [1]

These reasons are suggesting that cryptographic community should work on the crypto systems that can provide the security in quantum computer environment. Following are some recommended areas of crypto systems resistant on quantum computing-

1) Hash-based cryptography- Most of the application requires the unbroken digital signature in quantum environment. Some hashed based schemes are found to be practical to post quantum cryptography. There are many example but classic one is **Merkle's tree hash-public-key signature system (1979)**, building upon a one-message-signature idea of Lamport and Diffie. [3]

2) Code-based cryptography-This category include classic example of **McEliece's hidden-Goppa-code** public-key encryption system proposed by McEliece in 1978. Reason to be resistant in quantum computer is that it is based on Goppa code which has been unbroken till recent research done. No attack of significant affects has been detected on the code based cryptography that's why it is most suitable candidate for post quantum cryptography[3].

3) Lattice-based cryptography- Lattice based cryptography has promises to the post quantum cryptography because they enjoy the very strong proof based on implementation as well provide very high level of security with simplicity. We will be discussing the one of such encryption scheme in this paper and improved version. [3]

4) Multivariate – quadratic - equations cryptography- In recent years these crypto systems have been considered resistant to attacks and based on the quadratic equation over finite field. All of them use facts that MQ problem is N-P complete. One of many interesting examples is Patarin's "HFEv–" public-key-signature system (1996), generalizing a proposal by Matsumoto and Imai.2 Daniel J. Bernstein. [3]

5) Secret-key cryptography- In this category of cryptography the leading example is the Daemen–Rijmen "Rijndael" cipher (1998), which was renamed "AES," the Advanced Encryption Standard. [3]

In our document we are working on the lattice based cryptography. We are proposing NTRU encryption more secure, resistant and efficient.

2. DESCRIPTION OF MODEL

The NTRU, a lattice based cryptosystem, the encryption basically depends on the mixing of polynomial having small coefficients with reduction modulo p and q, where p and q are some constants. The encryption and decryption of NTRU is O (N^2) when the block of massage is O(N), as compared to RSA having O(N^3) .The key generation is very easy and fast of O(N) as compared to RSA having O(N^2). System validity depends on the probability theory because it uses the random polynomial that is why each element has many possible encryptions [8].

Some defined notations, parameter [7] and definitions that are followed in entire system are-

Definition of a lattice: Let v1, v2…. vk be a set of vectors in Rm. The set of all linear combinations a1v1 + a2v2…+...+ akvk , such that each ai ∈ Z, is a lattice. We call it more formally as the lattice generated by v1, v2, . . . ,vk.

Bases and the dimension of a lattice• Let L = { a1v1 + a2v2 + . . . + anvn| ai ∈ Z, I = 1 , . . . n } and v1, v2 . . . ,vn are n independent vectors, then we call that v1, v2,. . . ,vn is a basis for Lattice and that L has dimension n which is equal to cardinality of a vector[4].

N: (Degree constant). A positive integer which defines the dimension of the vector.

q: (Large Modulus).A positive integer. The associated NTRU lattice is a convolution modular lattice of modulus q.

p: (Small Modulus). An integer or a polynomial.

Df, Dg :(Private Key Spaces). Sets of small polynomials from which the private keys are taken.

Dm (Plain text Space**)**: Set of polynomials that represent encrypts able messages.

Dr (Blinding Value Space).Set of polynomials from which the temporary blinding value used during encryption is selected.

Center (centering method). It is way of performing mod q reduction on cipher text.

Convolution product: The Ring of Convolution Polynomials is R =Z[X] / (XN −1). Multiplication of Polynomials (* between polynomials) in this ring corresponds to the convolution product of their associated vectors, defined by

$$(f * g)(X) = \sum_{k=0}^{N-1} \Big(\sum_{i+j=k \pmod N} f_i g_j \Big) X^k.$$

Operation between two polynomials refers to the convolution product while for the constant and the polynomial it is simple multiplication. There is one more notation Rq = (Z/*q*Z)[*X*] /(*XN*−1) convolution operation in Rq can also be called as modular convolutions[3].

Definition1.A binary polynomial is one whose coefficients are all in the set {0,1}. A trinary polynomial is one whose coefficients are all in the set {0,±1}.

Definition2. Following are definition of the polynomial spaces $B_N(d)$,$T_N(d)$,$T_N(d1,d2)$-
Polynomials in space $B_N(d)$ have d number of coefficients equal to 1 and the other coefficients are 0. Polynomials in space $T_N(d)$ have d +1 number of coefficients equal to 1, have d number of coefficients equal to −1, and the other coefficients are 0. Polynomials in space $T_N(d1,d2)$ have d1 number of coefficients equal to 1, have d2 number of coefficients equal to −1, and the other coefficients are 0.

NTRU Encryption Algorithm:

NTRU Encrypt consists of three basic functions-

- Key Generation
- Encryption of plain text
- Decryption of cipher text

NTRU Encrypt key generation consists of the following operations:

1) Randomly generate polynomials f and g in df, dg respectively.

2) Invert f in Rq to obtain fq, invert f in Rp to obtain fp, and check that g is invertible in Rq [5].

3) The public key h = p *g * fq (mod q). The private key is the pair (f, fp).

NTRUEncrypt Encryption:-

NTRUEncrypt Encryption consists of the following operations-

1) Randomly select a “small” polynomial r from dr.

2) Calculate the cipher text e as e ≡ r * h + m (mod q).

NTRUEncrypt Decryption:-

NTRUEncrypt decryption consists of the following operations:

1) Calculate a ≡ center (f * e), where the center operation reduces its input into the interval [A, A+q−1] where A is an integer which decide the domain of the interval.

2. Recover m by calculating m ≡ fp * a (mod p).

3. CONTRIBUTION

In this paper we have proposed a new way of doing encryption in the NTRU system. We have extended the key and have done some complexion on input massage and even on the public key. In out implementation we are applying operation of the each byte and order of the byte. On first byte we are exchanging the first four bit with last four bit and in the second byte we are exchanging the first two bit to the next two bit. This sequence is also followed in reverse order in decryption. For illustration take a byte sequence 11110110.When occurred at the first number it is converted to

11111001in Fig-1.When this comes on the second or even places it is converted to 11111001in Fig-2. On the second improvement in the encryption scheme we have some complex operation on the key itself due to which public key has changed. In previous implementation we have the public key.

$$h = p * g * fq \pmod q.$$

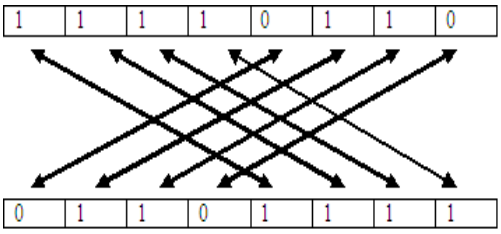

Fig 1: for odd placed byte

Where fq is the inverse of 'f' under modulo 'q' and * is convolution product of the two polynomial

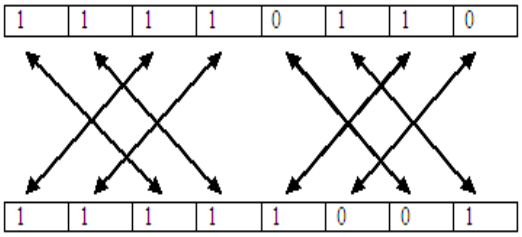

Fig 2: for even placed byte

Public Key Generation - Generate two polynomials randomly of degree N separately. Let these are r1 and r2 and do **Xoring** for each polynomial with other randomly generated polynomial. Convolution product is implemented for r1 and r2 with modulo q (say s) and find the convolution product of 's' and polynomial g. Now obtained product is multiplied by constant p modulo q(say t).We obtain convolution of g and fq and add 's' and 't' in it in modulo q .Result is h polynomial which is public key. Mathematically:-

r1 = r1 ^ random polynomial;

r2 = r2 ^ random polynomial;

s = r1 * r2 (modulo q);

t = ((s * g) * q) * p) % p;

h = p * g * fq (mod q)

h = (p * g * fq + s + t)(mod q)

Encryption:

1) Randomly select a "small" polynomial r belongs to Dr

2) Calculate the cipher text e as $e \equiv r * h + m \pmod q$ where m is message text.

Decryption:

In the decryption side we have private key (f, fp) (mod p). Now compute

a = f * e

Where e is an encrypted polynomial. Now obtain center of the polynomial from –q/2 to +q/2. This centering process is only for the maintaining the coefficient in the range between these –q/2 to +q/2. Now obtain the massage plain text by

m = fp * a (mod p)

Mathematical proof:

a = f * e

a = f * (p * r * h + f * m) (mod q)

m = fp * a (mod p)

m = fp * (f * p*r * h + f * m) (mod q) (mod p)

m = fp * (f * p * r *(p * g * fq + s + t + f * m) (mod q) + m) (mod q)(mod p)

m = fp * (f * p * p * r * g * fq + f * p * r * t + f * p * r * s + f * m)(mod q)(mod p)

m= (p *p * r * g * fq + p * r * r * s + fp * f * m)(mod q)(mod p)

m=m

Because these terms are multiple of p and when we take mod under p they get reduce to zero hence we get the original message.

4. IMPLEMENTATION

Public key generation:

Input: f and g polynomial.

Output: public key polynomial

1. Set: r1

2. Set: r2

3. Set r1 = r1 ^ random polynomial

4. Set r2= r1 ^ random polynomial

5. Set s = r1*r2;

6. Set t = p * g * s (mod q)

7. Set h = (g * fq + t + s)(mod q)

Here h is public key that provides more security when text is repeated more times

Digesting Input:

Input: g (polynomial of degree N with coefficient 0 or 1 only), m (plain text massage polynomial)

Output: Plain text massages polynomial.

0. Start:

1. Set: x=0,y=0

2.0 If (x mod 2 equals 0)

2.1 Set m + = g;

2.2 Set d = deg (N);

2.3.0 While (y < deg (N) / 2)

2.3.1 Set t = m [y];

2.3.2 Set m [y] = m [d + y]

2.3.3 Set m [d + y] = m [y];

3.0 Else then

3.1 Set y to zero

3.2 Set m - = g;

3.2.0 While (y < deg (N))

3.2.1 Set t = m [y];

3.2.2 Set m [y] = m [deg (N) – y];

3.2.3 Set m [deg (N) – y] = t;

4. End:

Undigesting Input:

Input: g (polynomial of degree N with coefficient 0 or 1 only), d (decrypted massage polynomial) [8].

Output: Plain text massages polynomial.

0. Start

1. Set: x = 0, y = 0;

2.0 If (x mod 2 equals 0)

2.1 Set d = deg (N);

2.2.0 While (y < deg (N) / 2)

2.2.1 Set t = m [y];

2.2.2 Set m [y] = m [d + y];

2.2.3 Set m [d + y] = m [y];

2.2.4 Set m + = g;

3.0Else then

3.1Set y to zero

3.1.0 While (y < deg (N))

3.1.1 Set t = m [y];

3.1.2 Set m[y] = m [deg (N) – y];

3.1.3Set m [deg (N) – y] = t;

3.1.4Set m - = g;

4. End;

5. OBSERVATION

Each public key cryptosystem has its own weakness and provide security based on some type of hard problem. Here in this NTRU encryption has some of its characteristics such as very less memory and computational cost. Security is based on the hard problem and the selection of the parameter set. in this scheme we have integer parameter N , P , Q and four set choosing the number of one and two in the polynomial like df ,dg ,dr etc.

TABLE 1
Comparison between NTRU and improved NTRU

Operation/entity	**NTRU**	**Improved Model**
Plain text block	$Nlog_2P$	$Nlog_2P$
Encrypted text block	$Nlog_2Q$	$Nlog_2Q$
Encryption speed	$O(N^2)$	$O(N^2)$
Decryption speed	$O(N^2)$	$O(N^2)$
Massage expansion	Log_pQ to 1	Log_pQ to 1
Private key length	$2Nlog_2P$ bits	$2Nlog_2P$ bits
Public key length	$Nlog_2Q$ bits	$Nlog_2Q$ bits

Now we are comparing the previous NTRU and the our model of on same set of parameter on each component like encryption and decryption and key size and other operation

We observe that both of the NTRU previous and our model have the same level of key length, and cost of operation encryption and decryption. While the change in the public key makes it more secure and digesting on the input makes it more complex to break

In the literature we have much public key crypto system with different type of hard problem including RSA based on difficulties of factoring problem, mackliece public key system on error detecting code and many others. When the modified scheme was compared with the other cryptosystem on key size and the operations we have concluded following TABLE II. This table concludes that NTRU message encryption varies even for long massage .Principle of expansion of the massage is exchange of the public key in massage block this is not significant problem. We have this solution for problem and this solution can also be implemented even for long massage with expansion of only after the first massage block. With this approach, from the sending side massage is with the polynomial with 0,1,-1 under modulo p=3,and interpreted as P1 for next massage block. The next massage block is p1*e1+m where m is first block of massage and m1 mod q

can be reduced exactly next block e2=p2*e2+m2 where p2 is calculated by squaring the m1 and reducing it by p=3. This process continues for massage of arbitrary length, hence this continues for massage of arbitrary length.

TABLE 2
Comparisons with other public key crypto system

Operation	NTRU (improved)	R S A	Mackliece
Encryption	N^2	N^2	N^2
Decryption	N^2	N^3	N^2
Public key	N	N	N^2
Private key	N	N	N^2
Massage expansion	varies	1-1	2-1

6. CONCLUSION AND FUTURE GUIDELINE

This document aims to meet the requirement of more secure and efficient NTRU. Security is achieved by introducing some more complex problem into the existing implementation and efficiency can be achieved by having some reduced implementation of polynomial multiplication of inverse computation. The most time consuming operation in NTRU are product of the polynomials because that is used for even for all operation like key generation, encryption and decryption.

In order to achieve the security we have achieved our goal to certain extent by modified algorithms with digesting function introduction. Comparison shown in this paper with other public key cryptosystem is much satisfying as cost remains the same and security increases highly. This is more important to build the confidence in post quantum cryptography. As this crypto system include very low computational requirement because polynomial coefficient are very small integer hence it is applicable for devices like mobile and embedded system with less computational power and which require sufficient amount of security[10].

For suggestion there can be two main areas one is reducing the cost of multiplication of polynomial which eventually will make the scheme more efficient. And the second one is using other scheme for security for case of repeated text. For this scheme digesting function-

F(x1+x1.x2+x2.x3…xn-1.xn) can be computed and undigesting correspondingly at decryption side where x1, x2, x3…xn are the byte of the text.

7. REFERENCES

[1] Introduction to post-quantum cryptography by Daniel.J. Bernstein Department of Computer Science, University of Illinois at Chicago.

[2] Post quantum cryptography by Daniel J. Bernstein, Johannes Buchmann Erik Dahmen Editors.

[3] Practical lattice-based cryptography: NTRUEncrypt and NTRUSign Jeff Hoffstein, Nick Howgrave-Graham, Jill Pipher,William Whyte.

[4] Handbook of Applied Cryptography, by A. Menezes, P. van Oorschot, and S. Vanstone, CRC Press, 1996.

[5] The NTRU Public Key Cryptosystem – A Tutorial, http: //www.ntru.com.

[6] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Cli_ord Stein.Introduction to Algorithms. MIT Press and the McGraw-Hill Book Company, second edition, 2001.

[7] N. Howgrave-Graham, J. H. Silverman, W. Whyte, Choosing Parameter Sets for NTRUEncrypt with NAEP and SVES-3, CT-RSA 2005.

[8] P. Shor, Polynomial time algorithms for prime factorization and discrete logarithms on a quantum computer, Preliminary version appeared in Proc. of 35th Annual Symp. on Foundations of Computer Science, Santa Fe, NM, Nov 20-22, 1994. Final version published in SIAM J. Computing 26 (1997) 1484.Published in SIAMJ .Sci.Statist.Comput.26:1484, 1997 e-Print Archive: quant-ph/9508027.

[9] M. Ajtai The shortest vector problem in L2 is NP-hard for randomized reductions (extended abstract) in Proc. thirtieth ACM symp on Th. of Comp.,1998, pp.10–19

[10] D. Bailey, D. Co±n, A. Elbrit, J. Silverman, and A. Woodbury, \NTRU in Constrained Devices," in Workshop on Cryptographic Hardware and Embedded Systems | CHES 2001 (C» . Ko»c, D. Naccache, and C. Paar, eds.)