A Proposal for SMS Security Using NTRU Cryptosystem

Ashok Kumar Nanda and Lalit Kumar Awasthi

Computer Science & Engineering Department,
National Institute of Technology, Hamirpur, Himachal Pradesh, India - 177005
ashokkumarnanda@yahoo.com, lalit@nith.ac.in

Abstract. Short Message Service (SMS) is getting more popular now-a-days. It will play a very important role in the future business areas of mobile commerce (M-Commerce). Presently many business organizations are using SMS for their business purposes. SMS's security has become a major concern for both business organizations and customers. There is a need for an end to end SMS Encryption in order to provide a secure medium for communication. Security is main concern for any business company such as banks who will provide these mobile banking services. Till now there is no such scheme that provides complete SMSs security. The transmission of an SMS in GSM network is not secure at all. Therefore it is desirable to provide SMS security for business purposes. In this paper, we have analyzed Number Theory Research Unit (NTRU) Crypto algorithm and NTRUSign (NTRU Signature) algorithm. We have compared theoretically the performance metrics like key size, key generation time, encryption time, decryption time, CPU computational power, speed, efficiency, memory space and security strength between NTRU and RSA. This theoretical results encouraged us to simulate the NTRU cryptosystem and NTRUSign algorithm on mobile phones using full size of SMS as future work.

Keywords: M-Commerce, NTRU Cryptosystem, NTRUSign, Performance analysis, RSA, SMS Encryption, SMS Security.

1 Introduction

In earlier times mobile phones used to be a craze, symbol of money and success but nowadays every common people finds its necessity of their individual life. Mobile phones are having a great influence in every one's live and are very convenient to keep with us. Mobile phones are a faster and more effective way to transfer information. Indeed, it is a resource that gives its user's great advantages. These days mobile phones are not just used for phone calls but they are about messaging, videos, songs, games, alarm clock, notes, calendar, reminder, etc. So one equipment, lots' of uses! Day by day mobile subscribers are increasing. The Fig. 1 shows the growth of mobile subscribers of world during 2009 to 2016F. 'F' stands for forecast value.

In 1992, the first SMS technology enables the sending and receiving of messages between two mobile phones. SMS message contains at most 140 bytes (1120 bits) of

K. Singh, A.K. Awasthi, and R. Mishra (Eds.): QSHINE 2013, LNICST 115, pp. 706–718, 2013.

data, so one SMS message can contain up to 160 characters (if 7-bit character encoding is used) and 70 characters (if 16-bit Unicode UCS2 character encoding is used). SMS provides more convenient for mobile phone users to communicate with each other using text messages via mobile phones either from mobile phones or Internet connected computers. One major advantage of SMS is that it is supported by 100% GSM mobile phones. Almost all subscription plans provided by wireless carriers include inexpensive SMS messaging service. The mobile messaging market is growing rapidly and is a very profitable business for mobile operators. It can be seen from Fig. 2 that the growth rate of SMS in world during 2000 to 2016F. 'F' stands for forecast value.

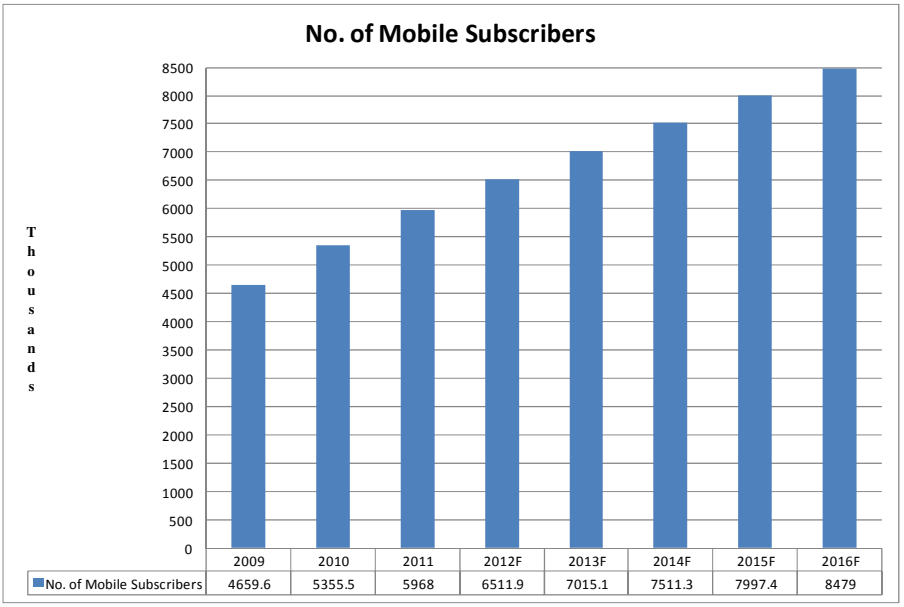
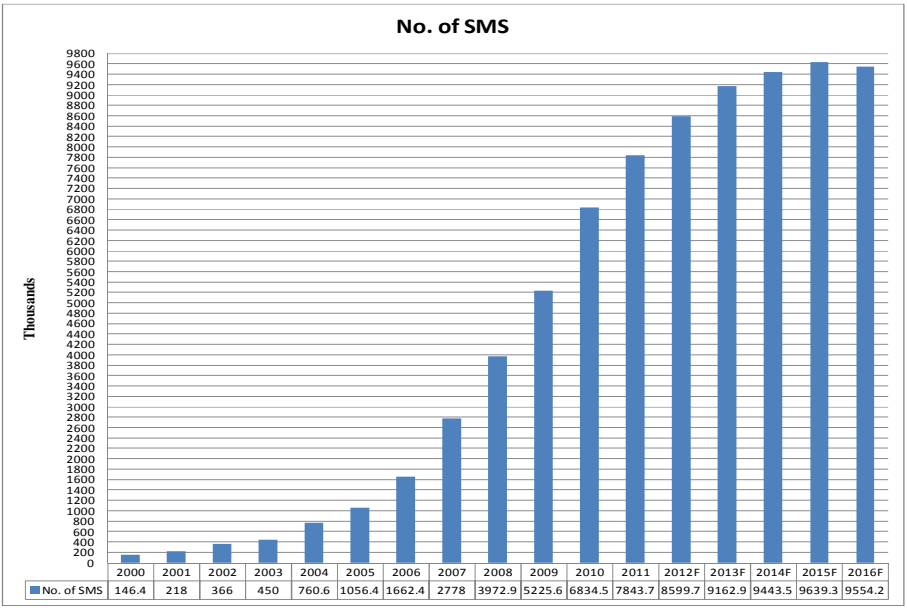


Fig. 1. Growth of Mobile Subscribers – World from 2009 to 2016F (F stands for Forecast) Source: Portio Research Ltd

SMS is getting more popular now-a-days. It will play a very important role in the future business areas of mobile commerce (M-Commerce). Many financial and business organizations are using SMS more for their business purposes. SMS has a variety of advantages and disadvantages for M-Commerce purpose. SMS's security has become a major concern for both business organizations and customers. There is a need fast key generation, encryption and decryption with optimization of memory size, CPU energy consumption. Security is main concern for any business company such as banks who will provide these mobile banking services to their customer. Currently there is no such scheme that provides complete SMSs security.

Many people of United States, EU5 (UK, Germany, France, Spain and Italy) and Japan prefer information exchange as text message (SMS) as compared to instant

message by mobiles is shown in below mentioned Table 1. The major advantages of SMS are: i) SMS is a personal like phone call but a person can read at any time without any disturbance to the work ii) Messages are instantly recorded so that one can refer at any time iii) It is relatively less SPAM free iv) SMS is discreet in nature v) SMS bills are considered as negligible vi) SMS is more convenient for deaf and hearing-impaired people to communicate vii) SMS is a store-and-forward service viii) SMS doesn't overload the network as much as phone calls ix) It is possible to send SMS many people at a time x) easy to use xi) common messaging tool among consumers xii) works across all wireless operators xiii) no specific software required to installation. There are very few and negligible disadvantages are : i) Consumes more time to type as compared to phone call ii) No proper authentication of SMS sender iii) Length of SMS is maximum 140 - 160 characters iv) Reliability and versatility can be compromised when using SMS v) does not support sending media, including videos, pictures, melodies or animations vi) does not offer a secure environment for confidential data during transmission. The same Table - 1 indicates that few people access financial services such as bank account information and financial news or stock quotes using SMS because SMS are not fully secure in wireless environment due to its broadcast nature.



	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012F	2013F	2014F	2015F	2016F
No. of SMS	146.4	218	366	450	760.6	1056.4	1662.4	2778	3972.9	5225.6	6834.5	7843.7	8599.7	9162.9	9443.5	9639.3	9554.2

Fig. 2. Growth of SMS – World from 2000 to 2016F (F stands for Forecast) Source: Portio Research Ltd

Presently researchers proposed some security concepts regarding SMS security. Most of the proposals are software frames to be installed on mobile device and /or on the SIM cards to implement security. When SMS used for M-Commerce the following services are required [1]:

Table 1. Mobile behavior in United States, EU5 (UK, Germany, France, Spain and Italy) and Japan – October, November, December 2010 Percent of total mobile audience (Age 13+)

	US	Europe	Japan
Used Messaging			
Sent Text Message	68%	82.7%	41.6%
Instant Messaging	17.2%	14.2%	3.6%
Accessed Financial Services			
Bank Accounts	11.4%	8%	7%
Financial news or stock quotes	10.2%	8%	16.5%

Source: comScore MobiLens (Feb 2011)

a) Confidentiality: only the valid communicating users can view the SMS.
b) Integrity: the SMS can't be tampered by the intruders. The system should be able to find out such alteration.
c) Non-repudiation: no party can deny the receiving or transmitting the data communicating between them.
d) Authentication: each party has to have the ability to authenticate the other party.
e) Authorization: it has to be ensured that, a party performing the transaction is entitled to perform that transaction or not.

We realized that security is most essential for mobile phone users and network operators to avoid different threats at different levels. The transmission of an SMS in GSM network is not secure at all. Therefore it is desirable to secure SMS for business purposes by additional encryption.

Rest of paper is organized as follows. Section 2 provides an overview of related work. Section 3 brief explaining about NTRU cryptosystems. Section 4 analysis and compares the theoretically results. Section 5 discusses about NTRUSign Scheme and followed by discussion and future work.

2 Related Work

Challa and Pradhan in 2007 [2] compared RSA and NTRU using 'C' language for measuring encryption, decryption speeds. They performed test for encryption and decryption using key size as 128 bits, 256 bits, 512 bits, 1Kb, 2Kb, 5Kb and 10Kb for both RSA & NTRU respectively. They used data size 22 bits and 10 bits for encryption and decryption methods respectively for RSA and 51 bits and 20 bits for encryption and decryption methods respectively for NTRU. Xiaoyu Shen and his team in 2009 [3] enhanced NTRU by changing forms of random polynomial 'f' and coefficient of polynomial integer 'p' and using low hamming weight products to improve efficiency of NTRU for mobile java systems. Their programs written in Java ME and used device emulator in the WTK (Sun Java Wireless Toolkit) 2.5.2. They considered NTRU-251 and RSA-1024 have same security level. They had compared key generation time, encryption time and decryption time between NTRU-251 and RSA-1024 using equivalent security key strength. Sameer Hasan Al-bakri and M.L. Mat Kiah in 2010 [4] used hybrid NTRU and AES-Rijndael for peer-to-peer

SMS security. They implemented in J2ME using mobile information device application (MIDlet). They performed test on Symbian OS of Nokia N70, Nokia N73, Nokia N93 and Nokia 5800 Express mobile phones with data size 1block (20Byte) with key-size of NTRU - 251.

3 NTRU Cryptosystem

The NTRU public key cryptosystem was developed in 1996 at Brown University by three mathematicians J. Hoffstein, J.Pipher and J.H. Silverman. NTRU encryption algorithm is a lattice-based alternative to RSA and ECC and is based on the shortest vector problem in a lattice. NTRU can be used in mobile devices and other mobile applications because of its features of easy generation of keys, high speed and low memory use [3]. This is based on shortest vector problem in a lattice and operations based on objects in a truncated polynomial ring $R = z[X]/(X^N - 1)$.

All polynomials in the ring have integer coefficients and degree at most N-1:

$$a = a_0 + a_1 x + a_2 x^2 + \ldots\ldots\ldots + a_{N-2} x^{N-2} + a_{N-1} x^{N-1} = \sum_{i=0}^{N-1} a_i x^i$$

it can represented as vector: $a = (a_0, a_1, a_{2,\ldots\ldots\ldots}, a_{N-2,} a_{N-1})$

$$b = b_0 + b_1 x + b_2 x^2 + \ldots\ldots\ldots + b_{N-2} x^{N-2} + b_{N-1} x^{N-1} = \sum_{i=0}^{N-1} b_i x^i$$

it can represented as vector: $b = (b_0, b_1, b_{2,\ldots\ldots\ldots}, b_{N-2,} b_{N-1})$

$$a + b = \sum_{i=0}^{N-1} a_i x^i + \sum_{i=0}^{N-1} b_i x^i = \sum_{i=0}^{N-1} (a_i + b_i) x^i$$

$$a * b = \sum_{i=0}^{N-1} a_i x^i * \sum_{i=0}^{N-1} b_i x^i = \sum_{k=0}^{N-1} [\sum_{i+j=k(\mathrm{mod}\, N)} a_i * b_j] x^k$$

Another operation we should know is modular arithmetic in which: and $a = b(\mathrm{mod}\, c)$ means a and b have the same reminder when they are divided by c.

When we do modular arithmetic to a polynomial in the ring with the integer modulus, it just means to divided each coefficient of the polynomial by the modulus and keep the reminders as the new coefficients.

NTRU has 3 integer parameters: *N*, *p*, *q*. *N* represents the degree of the polynomials at most N-1; *p* is smaller than *q*. *p* and *q* are small moduli used to reduce the coefficients of the polynomials. They do not have common divisor. We briefly describe the NTRU algorithm [3] as follows.

3.1 Key Generation

We have to choose two random polynomials *f* and *g* in the ring with the restriction that their coefficients are small, usually in {-1, 0, 1}. We import another symbol here: $L(d_1, d_2)$, which means a set of polynomials with d_1 coefficients are 1, d_2 coefficients

are -1 and the rest are 0. Usually we choose f from $L_f(d_f, d_{f-1})$ and g from $L_g(d_g, d_{g-1})$. Then we compute f_p (the inverse of f modulo p) and f_q (the inverse of f modulo q) with the property that $f * f_p = 1 \pmod p$ and $f * f_q = 1 \pmod q$.

If f doesn't have these inverses, another f should be chosen. The pair of polynomials f and f_p should be kept as the private key, and the public key h can be computed by $h = p * f_q * g \pmod q$

Both f and f_p are used for private key and h is used for public key.

Example: The parameters (N, p, q) have the values $N = 11$, $p = 3$ and $q = 32$ and therefore the polynomials f and g are of degree at most 10. The system parameters (N, p, q) are known to everybody. The polynomials are randomly chosen, so suppose they are represented by

$$f = -1 + x + x^2 - x^4 + x^6 + x^9 - x^{10} \text{ and}$$

$$g = -1 + x^2 + x^3 + x^5 - x^8 - x^{10}$$

Using the Euclidean algorithm the inverse of f modulo p and modulo q, respectively, is computed so

$$f_p = 1 + 2x + 2x^3 + 2x^4 + x^5 + 2x^7 + x^8 + 2x^9 \pmod 3 \text{ and}$$

$$f_q = 5 + 9x + 6x^2 + 16x^3 + 4x^4 + 15x^5 + 16x^6 + 22x^7 + 20x^8 + 18x^9 + 30x^{10} \pmod{32}$$

Which creates the public key h computing the product $h = p * f_q * g \pmod q$

$$= 8 + 25x + 22x^2 + 20x^3 + 12x^4 + 24x^5 + 15x^6 + 19x^7 + 12x^8 + 19x^9 + 16x^{10} \pmod{32}$$

3.2 Encryption

The message to be sent can be put into a form of a polynomial $m \in L_m(d_m, d_m)$ whose degree is at most N-1. Then we randomly choose a blinding polynomial $r \in L_r(d_r, d_r)$ in the ring. So the encrypted message e should be computed by $e = r * h + m \pmod q$

Example:

Let $m = -1 + x^3 + x^4 - x^8 + x^9 + x^{10}$ *and* $r = -1 + x^2 + x^3 + x^4 - x^5 - x^7$

$$e = r * h + m \pmod q$$

$$= 14 + 11x + 26x^2 + 24x^3 + 14x^4 + 16x^5 + 30x^6 + 7x^7 + 25x^8 + 6x^9 + 19x^{10} \pmod{32}$$

3.3 Decryption

First, use a part of the private key f to compute polynomial $a = f * e \pmod q$, then $b = a \pmod p$, and then We use the other part of the private key f_p to compute polynomial $c = f_p * b \pmod p = -1 + x^3 + x^4 - x^8 + x^9 + x^{10} = m$.

If this procedure is successful, c will be the original message m. Actually, for appropriate parameter values, this probability is extremely high. The polynomial satisfies

$$a = f * e(\text{mod}\, q) = f * (r * h + m)(\text{mod}\, q) = f * (r * (p * f_q * g + m)(\text{mod}\, q)$$
$$= p * r * g + f * m(\text{mod}\, q) \quad [f * f_p = 1(\text{mod}\, q)]$$

The coefficients of *r, g, f, m* and the prime *p* are all much smaller than *q*, and for appropriate parameter values, the coefficients of *a* can be ensured lie in *[-q/2, q/2*], so after reduced modulo *q*, these coefficients are not changed. Then

$$b = a(\text{mod}\, p) = (p * r * g + f * m)(\text{mod}\, p) = f * m(\text{mod}\, p)$$
$$c = f_p * b(\text{mod}\, p) = f_p * (f * m)(\text{mod}\, p) = m(\text{mod}\, p)$$
$$[f * f_p = 1(\text{mod}\, p)]$$

so polynomial c is just the original message m.

Example:

$$a = f * e(\text{mod}\, q)$$
$$= 3 - 7x - 10x^2 - 11x^3 + 10x^4 + 7x^5 + 6x^6 + 7x^7 + 5x^8 - 9x^9 - 7x^{10}(\text{mod}\, 32)$$
$$b = a(\text{mod}\, p) = -x - x^2 + x^3 + x^4 + x^5 + x^7 - x^8 - x^{10}(\text{mod}\, 3)$$
$$c = f_p * b(\text{mod}\, p) = -1 + x^3 + x^4 - x^8 + x^9 + x^{10} = m \quad \text{(Proved)}$$

The below figure 4 is representing overall analysis of NTRU algorithm for SMS using between two mobile phones.

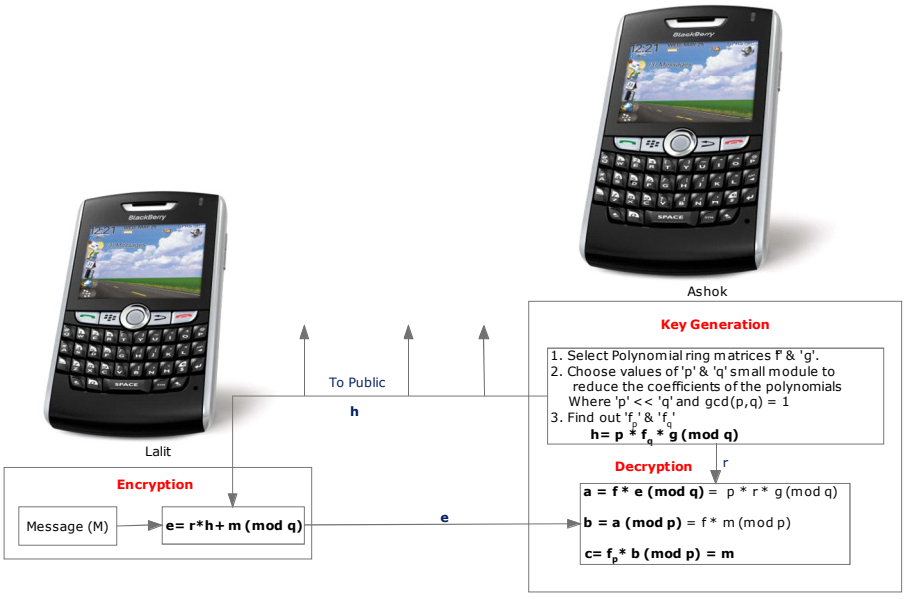


Fig. 3. Analysis of NTRU Crypto Algorithm applied for two mobile users

4 Theoretical Result Analysis and Comparison

RSA and ECC cryptosystems are considered as the most popular traditional public cryptography algorithms. In the literature, many authors presented many weaknesses on RSA and ECC. They stated that RSA is slow [5] Hastad stated in [6] that low exponent RSA is not secure if the same message is encrypted to several receivers. In practice, RSA has proved to be quite slow. Furthermore, RSA is not well suited for limited environments like mobile phones and smart cards without RSA co-processors [7]. RSA also requires longer keys in order to be secure compared to some other cryptosystems like ECC. ECC is faster than RSA [7], ECC-160 has 6× smaller key-size than RSA-1024 and can generate a signature 12 times faster than RSA and ECC is faster, it occupies less memory space than an equivalent RSA system, ECC generates asymmetry keys pair faster than RSA, ECC is more efficient than the ubiquitous RSA based schemes because ECC utilizes smaller key sizes for equivalent security, Security wise, ECC is stronger than RSA [8]. The NTRU crypto system is a new public key cryptography approved in 2009. The table 2 gives the total comparison between NTRU and RSA.

The company www.securityinnovation.com has built a cryptographic toolkit called NERI that is based on the NTRU algorithm. It provides data that compare the performance of NTRU with that of RSA and ECC on both servers and PDAs in Table 3. It implies that NTRU to have a performance advantage that ranges from 9:1 in the case of NTRU:ECC decryption on PDA to over 333:1 in the case of NTRU:RSA decryption on PDA.

In [4], they performed test on NTRU pair keys generation only for Nokia mobile phones like Nokia N70, N73, N93 and Nokia 5800 Xpress Music using Java emulator. The table 4 has shown its comparison. From the results, they noticed that NTRU algorithm performed very well on the mobile devices and there were no negative effects on the mobile devices' performance due to the small time required for the key generation. NTRU does not require high computing power, which makes it the best alternatives for mobile devices with providing either same or more security facility. Table 4 shows the proposed public key cryptography implementation in non-server architecture based on NTRU algorithm.

NTRU cryptosystem is gaining more popularity slowly because it's key size is very small, key generation, encryption speed, decryption speed are much faster and computation power requires very less, Operation speed is very fast, more efficient, consuming less space and more suitable for mobile devices shown in table 2. It is not free (as per our knowledge). It is standardized in IEEE 1363.1-2008 and X9.98-2010. Unlike RSA and ECC, NTRU is resistant to quantum computing based on crypto attacks. It is the smallest public key crypto available on market (8 kb). Some constraints are i) no support for NTRU in the leading browsers and ii) it is necessary required to implement NTRU at both ends of the SSL tunnel. www.securityinnovation.com provides SSL libraries and software development toolkits in C/C++ and Java. Unlike RSA and ECC, no successful attack has been recorded to break the security of this algorithm [4]. From the above, we hope that NTRU crypto algorithm will be more suitable and easy to implement in mobile devices for our proposed scheme.

Table 2. comparison among ntru, and rsa cryptosystems [3, 4]

Factors	NTRU	RSA
Key size	Very small (1/4 of RSA of same size)	slow
Key generation	200 times faster than RSA	slow
Encryption/sec	1113 times faster than 2048 – RSA	slow
Decryption/sec*	1132 ms for NTRU – 251 (More than 30 times faster)	35102 ms for RSA – 1024
Computation power	Too less than compared to both in mobile and smart cards.	Much more compared to NTRU
Speed	1300 times faster than 2048 – RSA and 117 times faster than ECC NIST – 224	Quite slow
Efficiency	Fastest	slow
Applicable to mobile device	Forefront on mobile environment	Not well suited without RSA coprocessors
Memory Space	Least than RSA	More compared both ECC and NTRU
Security	Strongest	Not secure if message is encrypted to several receivers. Needs longer key size.

Table 3. RSA, ECC and NTRU performance on servers (800 MHz Pentium III) and PDAs (Palm) [4, 7 & 8]

Key Size	Server		PDA	
	Encryption (blocks/sec)	Decryption (blocks/sec)	Encryption (blocks/sec)	Decrypt on (blocks/sec)
1024-bit RSA	1280	110	0.5	0.036
163-bit ECC	458	702	0.4	1.3
N=251 NTRU	22727	10869	21	12

5 NTRUSign Scheme

In this section, we briefly describe NTRUSign digital signature scheme. For NTRU encryption scheme, please refer above NTRU Cryptosystems in section 3.

In some steps, NTRUSign uses the quotient ring $R_q = Z_q[x] / (x^N - 1)$, where the coefficients are reduced modulo q, where q is typically a power of 2, for example 128.

The multiplicative group of units in R_q is denoted by R_q^*. The inverse polynomial of a a ε R_q^* is denoted by a^{-1}. If a polynomial '*a*' has all coefficients chosen from the set {0, 1}, we call this a binary polynomial.

Table 4. NTRU pair keys generation operation test [4]

	Nokia N70	**Nokia N73**	**Nokia N93**	**Nokia 5800 Xpress music**
generation	2G	3G	3G	5G
Operating System	Symbian OS v8.1a	Symbian OS v9.1	Symbian OS v9.1	Symbian OS v9.4
CPU	ARM9	Dual ARM 9	Dual ARM 11	ARM 11
Clock rate	220MHz	220MHz	332 MHz	434 MHz
Internal memory	22MB	42MB	50MB	81MB
External Memory	MMC type	2 GB Mini SD	2 GB Mini SD	16GB Mini SD
pair keys generation operation	142 ms	77 ms	53 ms	29 ms

The security of NTRUSign scheme is based on the approximately closest vector problem in a certain lattice, called NTRU lattice. In this scheme, the signer can sign a message by demonstrating the ability to solve the approximately closest vector problem reasonably well for the point generated from a hashed message in a given space.

The basic idea is as follows: The signer's private key is a short basis for an NTRU lattice and his public key is a much longer basis for the same lattice. The signature on a digital document is a vector in the lattice with two properties:

- The signature is attached to the document being signed.
- The signature demonstrates an ability to solve a general closest vector problem in the lattice.

NTRUSign digital signature scheme works as follows [9]:

5.1 System Parameters

- *N*: a (prime) dimension.
- *q*: a natural number used as a modulus.
- d_f, d_g: are nonnegative integers in the interval [0, N] used as a key size parameters; .
- NormBound: a bound parameter *of verification.*

5.2 Key Generation

A signer creates his public key h and the corresponding private key $\{(f, g), (F, G)\}$ as follows:

- Choose binary polynomials f and g with d_f 1's and d_g 1's, respectively.
- Compute the public key $h \equiv f^{-1} * g (\mathrm{mod}\, q)$.
- Compute small polynomials (F, G) satisfying $f * G - g * F = q$.

5.3 Signing Step

A signer generates his signature s on the digital document D as follows:

- Obtain the polynomials (m_1, m_2) mod q for the document D by using the public hash function.
- Write $G * m_1 - F * m_2 = A + q * B$; and $- g * m_1 + f * m_2 = a + q * b$;

where A and a have coefficients between – q/2 and q/2.

- Compute polynomials s and t as
  $s \equiv f * B + F * b \ (mod\ q)$,
  $t \equiv g * B + G * b \ (mod\ q)$.
  Here, a vector $(s,t) \in L_h^{NT}$ is very close to $m = (m_1, m_2)$.
- The polynomial s is the signature on the digital document D for the public key h.

5.4 Verification Step

For a given signature s and document D, a verifier should do the following:

- Hash the document D to recreate (m_1, m_2) mod q.
- Using the signature s and public key h, compute the corresponding polynomial $t \equiv s * h \ (mod\ q)$,
- Which becomes exactly the same as the polynomial $g * B + G * b \ (mod\ q)$. (Note that (s, t) is a point in the NTRU lattice L_h^{NT})
- Compute the distance from (s, t) to (m_1, m_2) and verify that this value is smaller than the NormBound parameter. In other words, check that $\| s - m_1 \|^2 + \| t - m_2 \|^2 \leq \text{NormBound}^2$, where the norm ($\| \cdot \|$) is a centered norm.
- NTRUSign algorithm uses the centered norm concept instead of Euclidean norm in verification step to measure the size of an element $a \in R$.

6 Discussion and Future Work

Now-a-days SMS is more popular for different applications in our daily real life. Errorless data transmission with secured is important in wireless environment. In this paper we have discussed about NTRU Cryptosystem and NTRUSign Scheme. From the table 2 - 4, we concluded that NTRU cryptosystem is much faster and providing stronger security than other traditional (example RSA and ECC in both server and PDA) cryptosystems. We are expecting that it will be more efficient scheme and will provide better result for our proposed scheme to implement for SMS's security at any mobile devices. So it may improve the current security level, fastest speed and provide reliable message at receiver end with respect to key generation, encryption decryption, CPU power consumption and memory size with smaller key size.

Our future work is to implement NTRU crypto algorithm and NTRUSign scheme for any mobile device to provide security of SMS and compare it with traditional cryptosystems with respect to all performance parameters like key generation time, encryption time & decryption, CPU power consumption, memory space and security strength.

Acknowledgment. This research work is supported by Ministry of Human Resource Development (MHRD), Government of India. The authors would like to thank the anonymous reviewers for their insightful comments.

References

1. Hossain, A., Jahan, S., Hussain, M.M., Amin, M.R., Shah Newaz, S.H.: A Proposal For Enhancing The Security System of Short Message Service In GSM. In: 2nd International Conference on Anti-Counterfeiting, Security and Identification, ASID 2008 (2008), doi:10.1109/IWASID.2008.4688386
2. Challa, N., Pradhan, J.: Performance Analysis of Public key Cryptographic Systems RSA and NTRU. International Journal of Computer Science and Network Security 7(8), 87–96 (2007)
3. Shen, X., Du, Z., Chen, R.: Research on NTRU Algorithm for Mobile Java Security. In: International Conference Scalable Computing and Communications; Eighth International Conference on Embedded Computing (SCALCOM-EMBEDDEDCOM 2009), pp. 366–369 (2009)
4. Al-Bakri, S.H., Mat Kiah, M.L., Zaidan, A.A., Zaidan, B.B., Alam, G.M.: Securing peer-to-peer mobile communications using public key cryptography: New security strategy. International Journal of the Physical Sciences 6(4), 930–938 (2011)
5. RSA description and algorithm, http://www.oocities.org/hmaxf_urlcr/rsa.htm
6. Kurosawa, K., Okada, K., Tsujii, S.: Low exponent attack against elliptic curve RSA, http://www.citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.44.2453
7. Karu, P., Loikkanen, J.: Practical Comparison of Fast Public-key Cryptosystems, http://www.tml.tkk.fi/Opinnot/Tik-110.501/2000/papers.html

8. Gupta, V., Gupta, S., Chang, S.: Performance Analysis of Elliptic Curve Cryptography for SSL, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.149.3368
9. Min, S., Yamamoto, G., Kim, K.: Finding Malleability in NTRUSign, http://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&sqi=2&ved=0CEwQFjAA&url=http%3A%2F%2Fwww.autoidlabs.org%2Fuploads%2Fmedia%2FAUTOIDLABS-WP-HARDWARE-033.pdf&ei=IS3aT4ZLieutB7O-kZUB&usg=AFQjCNE_5aqVHyvfFisC7GwheselLkgsfA