NTRU Cryptosystem and Its Analysis

Overview

1. Introduction to NTRU Cryptosystem
2. A Brief History
3. How the NTRU Cryptosystem works? Examples
4. Why the Decryption Works?
5. The Advantages of NTRU
6. Implementation and Comparative Analysis
7. Reference

NTRU keys and parameters

N - the polynomials in the ring R have degree N-1. (Non-secret)

q - the large modulus to which each coefficient is reduced. (Non-secret)

p - the small modulus to which each coefficient is reduced. (Non-secret)

f - a polynomial that is the private key.

g - a polynomial that is used to generate the public key h from f (Secret but discarded after initial use)

h - the public key, also a polynomial

r - the random “blinding” polynomial (Secret but discarded after initial use)

d - coefficient

1. What is NTRU Cryptosystem?

NTRU – Nth Degree Truncated Polynomial Ring Units (or R=Z[X]/(X^N-1))

NTRU is the first public key cryptosystem not based on factorization or discrete logarithmic problems.

NTRU is a lattice-based alternative to RSA and ECC and is based on the shortest vector problem in a lattice.

2. A Brief History

NTRU was founded in 1996 by 3 mathematicians: Jeffrey Hoffstein, Joseph H. Silverman, Jill Pipher

Later (at the end of 1996), these 3 mathematicians + Daniel Lieman founded the NTRU Cryptosys-tems, Inc, Boston, USA.

The mathematicians were considered on speeding up the process.

In 2009, NTRU Cryptosystem has been approved for standardization by the Institute of Electrical and Electronics Engineers (IEEE) [Hoffstein J., Lieman D., Pipher J., Silverman J. "NTRU: A Public Key Cryptosystem", NTRU Cryptosystems, Inc. (www.ntru.com).]

3. How the NTRU Cryptosystem works? Examples

Operations are based on objects in a truncated polynomial ring R=Z[X]/(X^N-1), polynomial degree at most N-1:

$$a_0 + a_1x + a_2x^2 + \cdots + a_{N-1}x^{(N-1)}$$

Key Generation

1st step. User B randomly chooses 2 small polynomials f and g in the R (the ring of truncated polynomials).

Notes:

- The values of these polynomials should kept in a secret.

- A chosen polynomial must have an inverse

2nd step. The inverse of f modulo q and the inverse of f modulo p will be computed

Properties: f*fq^-1 = 1 (modulo q)

and f*fp^-1 = 1 (modulo p)

3rd step. Product of polynomials will be computed: h = p * ((Fq)*g)mod q.

Private key of B: the pair of polynomials f and fp

Public key of B: the polynomial h.

Example of a key generation

Public parameters (*N, p, q, d*) = (7, 3, 41, 2).

*1. Bob chooses: **f** (x) = x^6–x^4+x^3+x^2–1 -Private and **g**(x) = x^6+x^4–x^2–x.*

*2. **F**q(x) = **f** (x)^ –1 (mod q) = 8x^6 + 26x^5 + 31x^4 + 21x^3 + 40x^2 + 2x + 37 (mod 41).*

***F**p(x) = **f** (x)^ –1 (mod p) = x^6 + 2x^5 + x^3 + x^2 + x + 1 (mod 3) - Private key of Bob*

*3. **h**(x) = p * (Fq)*g(mod q) = 20x^6 + 40x^5 + 2x^4 + 38x^3 + 8x^2 + 26x + 30 (mod 41) - Public key*

NTRU Encryption

User A has a message to transmit:

1st step. Puts the message in the form of polynomial m whose coefficients is chosen modulo p between -p/2 and p/2 (centered lift)

2nd step. Randomly chooses another small polynomial r (to obscure the message).

3rd step. Computes the encrypted message:

$$e = r*h + m \text{ (modulo } q)$$

Example of NTRU Encryption

Alice decides to send Bob the message:

$\mathbf{m}(x) = -x5 + x3 + x2 - x + 1$ using the ephemeral key $\mathbf{r}(x) = x\text{^}6 - x\text{^}5 + x - 1$.

e= r*h + m (modulo q)

$e(x) \equiv 31x\text{^}6+19x\text{^}5+4x\text{^}4+2x\text{^}3+40x\text{^}2 +3x+25 \pmod{41}$

$(N, p, q, d) = (7, 3, 41, 2)$

NTRU Decryption

B receives a message e from A and would like to decrypt it.

1st step. Using his private polynomial f he computes a polynomial a= f*e(mod q). B needs to choose coefficients of a that lie in an interval of length q.

2nd step. He computes the polynomial b=a(mod p). B reduces each of the coefficients of a modulo p.

- 3rd step. B uses the other private polynomial fp to compute c=fp*b(modulo p), which is the original message of A.

Example of NTRU Decryption

1. a= f*e(mod q) B computes a ≡ $x^6 + 10x^5 + 33x^4 + 40x^3 + 40x^2 + x + 40$ (mod *41*).

2. B then centerlifts modulo *q* to obtain

b = a(mod p) = $x^6 + 10x^5 – 8x^4 – x^3 – x^2 + x – 1$ (mod 3).

3. B reduces a(*x*) modulo *p* and computes

c = F*p*(*x*) **b*(*x*) ≡ $2x^5 + x^3 + x^2 + 2x + 1$ (mod *3*).

Centerlifting modulo *p* retrieves A's plain text m(*x*) = $–x5 + x3 + x2 – x + 1$.

(*N*, *p*, *q*, *d*) = (7, 3, 41, 2)

4. Why the Decryption Works?

If the NTRU parameters (N, p, q, d – public parameters) are chosen to satisfy

$$q > (6d + 1)\, p$$

this inequality ensures that decryption never fails.

Example (N, p, q, d) = (7, 3, 41, 2).

According to the example: 41 = q > (6d + 1)p = 39

According to the latest research the following parameters are considered secure [Wikipedia - the free encyclopedia "NTRU Cryptosystems Inc."]:

Parameters	N	q	p
Moderate Security	167	128	3
Standard Security	251	128	3
High Security	347	128	3
Highest Security	503	256	3

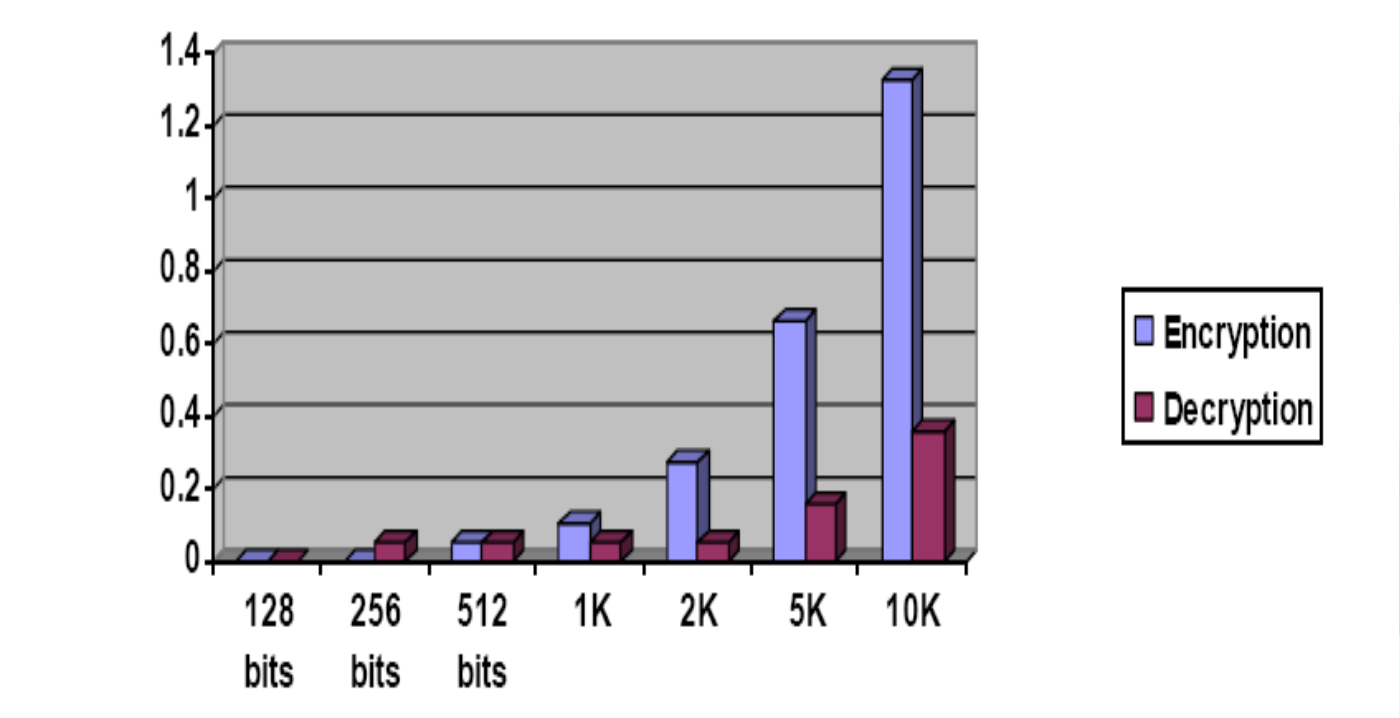
5. The advantages of NTRU

- more efficient encryption and decryption, in both hardware and software implementations;

- much faster key generation allowing the use of “disposable" keys (because keys are computationally “cheap" to create).

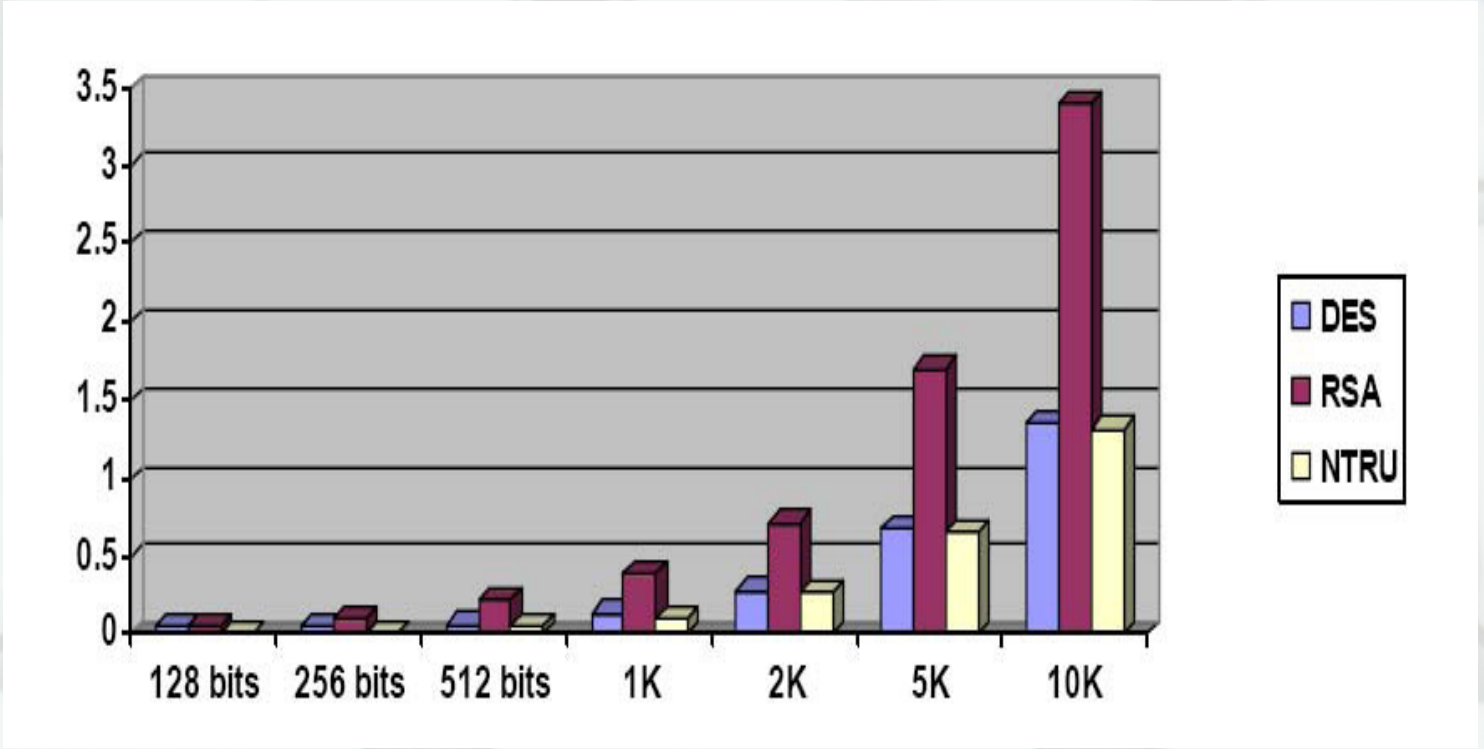
- low memory use allows it to use in applications such as mobile devices and Smart-cards.

6. Implementation and Comparative Analysis



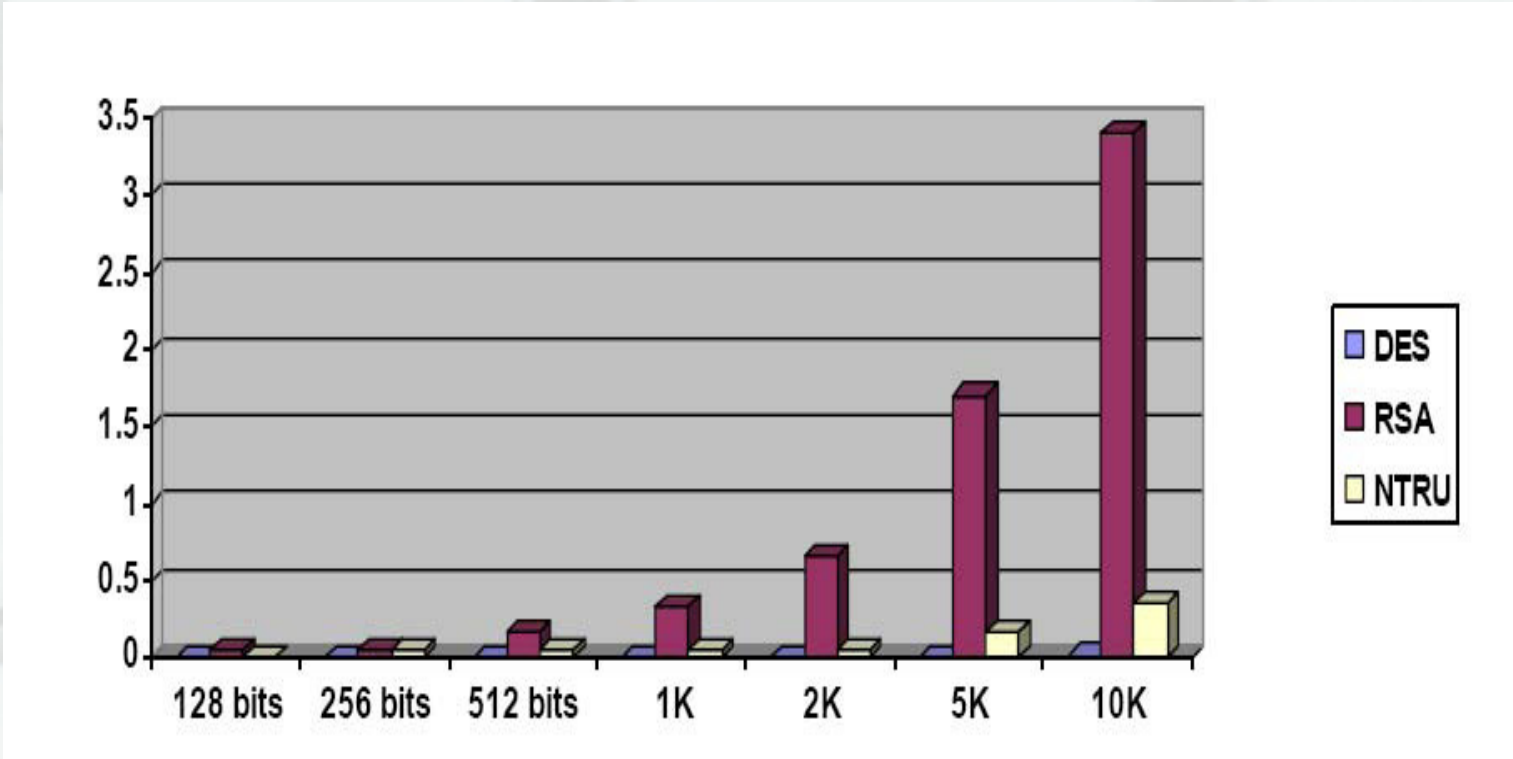
Performance of Encryption and Decryption timings of NTRU

Encryption Analysis DES, RSA, NTRU



Performance analysis on encryption for DES, RSA and NTRU methods

Decryption Analysis DES, RSA, NTRU



Performance analysis on decryption for DES, RSA and NTRU methods

Performance Analysis and Comparison of Symmetric and Asymmetric Key Cryptosystems

Method	DES	RSA	NTRU
Approach	Symmetric	Asymmetric	Asymmetric
Encryption	Faster	Slow	Fastest
Decryption	Fastest	Slow	Faster
Key Distribution	Difficult	Easy	Easy
Complexity	O(Log N)	O(N^3)	O(NLogN)
Security	Moderate	Highest	High

Conclusions

According to a paper by Calla Parasitism and Jayapura Prada:

- If an application is required with the highest decryption priority DES is more suitable

- An asymmetric key cryptographic system provides high security in all ways. Encryption, decryption and complexity are high in NTRU

- The RSA provides the highest security to the business application.

References

1. Hoffstein J., Lieman D., Pipher J., Silverman J. “NTRU: A Public Key Cryptosystem”, NTRU Cryptosystems, Inc. (www.ntru.com).

2. Parasitism C, Prada J. “Evaluation of Performance Characteristics of Cryptosystem Using Text Files”, Journal of Theoretical and Applied Information Technology, Jatit, 2008

3. Hoffstein J., Pipher J., Silverman J. An Introduction to Mathematical Cryptography, New York, 2008 http://www.springerlink.com/content/978-0-387-77993-5#section=229331&page=4&locus=51

4. Hoffstein J., Pipher J., Silverman J. “ NTRU – A ring based public key cryptosystem”

5. Wikipedia - the free encyclopedia “NTRU Cryptosystems Inc.”

Quiz

1. When and who founded the NTRU Cryptosystem?
2. What are the private keys of a user B?
3. Why r-key used in encryption?
4. What are the advantages of the NTRU cryptosystem and where it can be used?
5. What is the time complexity of the NTRU Cryptosystem?