

Capabilities and Performance of a JCE Implementation of NTRU Public Key Cryptosystem

Project Specification

ECE 646, Fall 2005
Professor Kris Gaj
George Mason University

Authors:

Krishnapriya Kadati (kkadati@gmu.com)
KarthikaDeepthi Bhimavarapu (kbhimava@gmu.edu)
George Koodarappally (gkoodara@gmu.edu)

Introduction:

NTRU is an emerging public key cryptosystem that uses polynomial algebra and arithmetic modulo a small integer p and a large integer q [7]. In the scope of this project we will compare the capabilities and performance of a pure Java JCE implementation, a JNI based C/C++ native implementation and a pure Visual C++ implementation..

Implementation Environment:

Java and C/C++ will be used for the implementation. The JCE Implementation will use the Java 2 Standard Edition 5.0 JDK API. Visual Studio .NET 2003 will be used for developing the C/C++ native implementation. All code will be developed and executed on a Pentium IV workstation running Windows XP/2000.

Additional Software Needed:

All Java/C/C++ source code will be developed from scratch based on published literature on NTRU. M.I.R.A.C.L library may be used to test correctness of developed mathematical operations.

Input/Output Specification:

The NTRU cryptosystem uses three integers (N, p, q) as operational parameters that determine the security of the system. A message m is a polynomial of degree $N-1$ whose coefficients are reduced modulo p . The polynomials are represented by their coefficients.

Functionality:

The NTRU functionality that will be implemented will include Key Generation, Encryption, Decryption as in NTRUEncrypt and Authentication as in NTRUSign.

Key Generation in NTRU involves the generation of a private/public key pair which is in the form of polynomials of degree $N-1$. The degree of the polynomials used determines the strength of the Cryptosystem. The higher the degree the higher the security.

The messages are encrypted by the sender using the public key of the receiver and decrypted by the receiver using their private key.

Test Plan:

The functionality testing will cover Key Generation, Encryption, Decryption and Authent functionality. The performance testing will compare the performance of the different implementations using four typical operational Input Parameters that cover Moderate, Standard, High and Highest Security levels.

The input test vector for functionality testing will be ($N: 11, q: 32, p: 3$)

The input test vectors for performance testing the different security scenarios will be as follows:

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

The Input Message will be in the form of a polynomial of degree $N-1$ with the coefficients reduced modulo p above.

Plan of Experiments:

Develop and Test pure Java Implementation of NTRUEncrypt and NTRUSign
Develop and Test Visual C++ Implementation of NTRUEncrypt and NTRUSign
Develop and Test JNI C/C++ Implementation of NTRUEncrypt and NTRUSign

Time Schedule:

September 12 Select initial choice of project topics

September 19	Select final choice of project topic
September 25	Submit first version of project specification
October 01	Submit final project specification
October 17	First progress report
October 24	Develop Java Implementation of NTRUEncrypt using JCE
October 31	Develop Visual C++ Implementation of NTRUEncrypt
November 07	Develop JNI C/C++ Implementation of NTRUEncrypt
November 14	Second progress report
November 21	Develop Java Implementation of NTRUSign using JCE
November 28	Develop Visual C++ Implementation of NTRUSign
December 05	Develop JNI C/C++ Implementation of NTRUSign
December 05	Final progress report with draft of final viewgraph presentation
December 12	Submit final project report
December 19	Final oral presentation

List of Possible Specification Changes:

The JNI C/C++ Implementation of NTRUSign is conditional on the timely completion of the Java and C++ implementations of NTRUSign.

List of Literature:

- [1] Java Cryptography Extension (JCE) (<http://java.sun.com/products/jce/>), Sun Microsystems, Inc.2004
- [2] D. Hook, *Beginning Cryptography with Java*. Indianapolis, IN: Wiley Publishing, Inc. 2005
- [3] Java Cryptography Extension (JCE) for the Java 2 Standard Edition 5.0 Development Kit (JDK 5.0), API Specification, Sun Microsystems, Inc.2004
- [4] C. Whelan, A. Dufy, A. Burnett, T. Dowling, "A Java API for Polynomial Arithmetic", Proceedings of the 2nd International conference on Principles and practice of programming in Java, Kilkenny City, Ireland, June 16 – 18, 2003, pp 139 – 144
- [5] NTRU CryptoLab Home (Online), (<http://www.ntru.com/cryptolab/index.htm>) , NTRU Cryptosystems, Inc.
- [6] NTRU CryptoLab Articles (Online), (<http://www.ntru.com/cryptolab/articles.htm>) , NTRU Cryptosystems, Inc.
- [7] Jeffrey Hoffstein, Jill Pipher, Joseph H. Silverman, "NTRU: A Ring Based Public Key Cryptosystem" (PDF), (<http://www.ntru.com/cryptolab/pdf/ANTS97.pdf>) , Algorithmic Number Theory (ANTS III), Portland, OR, June 1998, J.P. Buhler (ed.), Lecture Notes in Computer Science 1423, Springer-Verlag, Berlin, 1998, 267-288
- [8] J. Hoffstein, J. Silverman, "Optimizations for NTRU" (PDF), (http://www.ntru.com/cryptolab/pdf/TECH_ARTICLE_OPT.pdf) , Public-Key Cryptography and Computational Number Theory (Warsaw, September 11-15, 2000
- [9] J. Hoffstein, J. Silverman, "Random Small Hamming Weight Products With Applications to Cryptography" (PDF), (

- http://www.ntru.com/cryptolab/pdf/TECH_ARTICLE_RAND.pdf).
- [10] D. Bailey, D. Coffin, A. Elbirt, J. Silverman, A. Woodbury, “NTRU in Constrained Devices” (PDF), (<http://www.ntru.com/cryptolab/pdf/ntruches2001.pdf>), Proc. Cryptographic Hardware and Embedded Systems, Paris, France, 2001
 - [11] NTRU CryptoLab Tutorials (Online), (<http://www.ntru.com/cryptolab/tutorials.htm>) , NTRU Cryptosystems, Inc.
 - [12] Algebra Tutorial (Online), (http://www.ntru.com/cryptolab/tutorial_algebra.htm)
 - [13] The NTRU Public Key Cryptosystem (Online), (http://www.ntru.com/cryptolab/tutorial_pkcs.htm)
 - [14] The NTRU Public Key Cryptosystem: Enhancements I (Online), (http://www.ntru.com/cryptolab/tutorial_advanced.htm).
 - [15] The NTRU Public Key Cryptosystem: Enhancements II (Online), (http://www.ntru.com/cryptolab/tutorial_hamming.htm)
 - [16] NTRU CryptoLab NTRU Algorithms, NTRUSign (Online), (http://www.ntru.com/cryptolab/intro_ntrusign.htm), NTRU Cryptosystems, Inc.
 - [17] J. Hoffstein, N. Howgrave-Graham, J. Pipher, J. H. Silverman, W. Whyte, “NTRUSign: Digital Signatures Using NTRU Lattice” (PDF), (http://www.ntru.com/cryptolab/pdf/NTRUSign_RSA.pdf), CT-RSA Proceedings, 2003
 - [18] Colleen M. O'Rourke, "Efficient NTRU Implementations" (PDF), (http://www.crypto.wpi.edu/Publications/Documents/ms_corourke.pdf), MS Thesis, Worcester Polytechnic Institute, May 2002
 - [19] Rodney D'Souza, “The NTRU Cryptosystem: Implementation and Comparative Analysis” (PDF), (http://ece.gmu.edu/courses/ECE543/project/reports_2001/dsouza.pdf), ECE 646 Semester Project, George Mason University, Fall 2001.
 - [20] A. J. Menezes, P. C. Van Oorschot, S. A. Vanstone, *Handbook of Applied Cryptography*, (online), (<http://www.cacr.math.uwaterloo.ca/hac/>), CRC Press, 1996, Chapters 4 and 8

Anything Else: