IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. VIII (Mar-Apr. 2014), PP 148-151
www.iosrjournals.org

Text Encryption using Lattice-Based Cryptography

Vishnu Kumar
(Department of IT, DIT, Dehradun, India)

Abstract: *Lattice-based cryptography provides a much stronger notion of security, in that the average-case of certain problems in lattice-based cryptography is equivalent to the worst-case of those problems. There are strong indications that these problems will remain secure under the assumption of the availability of quantum computers, unlike both the integer-factorization and discrete logarithm problems as relied upon in many conventional cryptosystems. In this paper, the author will explore various methods to improve the practicality of lattice-based cryptosystems (GGH Encryption scheme) and to optimize the algorithm that make up these cryptosystems for modern computer processors.*

Keywords: *Asymmetric key Cryptosystem, Cryptography, Lattice Cryptography, Security*

I. Introduction

The Lattice-based cryptographic constructions hold a great promise for post-quantum cryptography. Many of them are quite efficient, and some even compete with the best known alternatives; they are typically quite simple to implement; and of course, they are all believed to be secure against quantum computers.

Lattice problems are typically quite hard. The best known algorithms either run in exponential time, or provide quite bad approximation ratios. The field of lattice-based cryptography has been developed based on the assumption that lattice problems are hard. There are currently no known quantum algorithms for solving lattice problems that perform significantly better than the best known classical (i.e., non-quantum) algorithms. This is despite the fact that lattice problems seem like a natural candidate to attempt to solve using quantum algorithms: because they are believed not to be NP-hard for typical approximation factors, because of their periodic structure, and because the Fourier transform, which is usually exploited in quantum algorithms, is tightly related to the notion of lattice duality.

In terms of security, lattice-based cryptographic constructions can be divided into two types. The first includes practical proposals, which are typically very efficient, but often lack a supporting proof of security. The second type admits strong provable security guarantees based on the worst-case hardness of lattice problems, but only a few of them are sufficiently efficient to be used in practice. We will discuss enhanced practical implementation of the latter type.

The strong security guarantees given by constructions of the latter type, namely that of worst-case hardness. What this means is that breaking the cryptographic construction (even with some small non-negligible probability) is provably at least as hard as solving several lattice problems (approximately, within polynomial factors) in the worst case. In other words, breaking the cryptographic construction implies an efficient algorithm for solving any instance of some underlying lattice problem. In most cases, the underlying problem is that of approximating lattice problems such as SVP to within polynomial factors, which as mentioned above, is conjectured to be a hard problem [1].

Such a strong security guarantee is one of the distinguishing features of lattice-based cryptography. Virtually all other cryptographic constructions are based on average-case hardness. For instance, breaking a cryptosystem based on factoring might imply the ability to factor some numbers chosen according to a certain distribution, but not the ability to factor all numbers. Attempts to solve lattice problems by quantum algorithms have been made since Shor's discovery of the quantum factoring algorithm in the mid-1990s, but have so far met with little success if any at all.

II. Ggh Encryption Scheme

The **Goldreich–Goldwasser–Halevi (GGH)** lattice-based cryptosystem is an asymmetric cryptosystem based on lattices. There is also a GGH signature scheme.

In 1996, Goldreich, Goldwasser and Halevi [2] proposed an efficient way to build a cryptosystem that uses lattice theory, inspired by McEliece cryptosystem [3] and based on Bounded Distance Decoding. Their practical proposition of a cryptosystem was attacked and broken by Nguyen in 1999 [4]. However, the general idea is still viable, as can be seen by the many variants of the basic GGH cryptosystem that have been proposed since.

The Goldreich–Goldwasser–Halevi (GGH) cryptosystem makes use of the fact that the closest vector problem can be a hard problem.It uses a trapdoor one-way function that is relying on the difficulty of lattice reduction. The idea included in this trapdoor function is that, given any basis for a lattice, it is easy to generate a vector which is close to a lattice point, for example taking a lattice point and adding a small error vector. But to return from this erroneous vector to the original lattice point a special basis is needed.

Key Generation

i. Create a good basis R.
ii. Transform this good basis R into a bad basis Q through a unimodular transformation.
iii. Publish bad basis Q as public basis and keep good basis R as private basis.

Encryption

i. Choose any lattice vector w using the public basis Q and add some small plaintext vector p to it.
ii. Send this new vector c = w + p as the cipher text.

Decryption

i. Using the private basis, compute the closest lattice vector w to the cipher text c.
ii. Subtract this lattice vector w from the cipher text to give the plaintext p = c - w.

It is important to note that while Nguyen broke the original GGH cryptosystem in 1999 due to a limited parameter set, the basic premise is still viable [5].

III. Proposed Work

GGH encryption scheme introduces some random error vector to the cipher text so that it can secure the cipher text. This error is reduced at the receiver end using Babai Roundoff algorithm. If the error is small as compared to the lattice point it can be removed but if the error is large then it can give unexpected results. So, in the proposed encryption scheme we have reduced the error introduction part and added RSA at that point to secure the cipher text and maintain integrity of the message. Also in the key generation step we have we have turned good basis R into a bad basis Q by performing transpose and multiplication on the good basis R.

Key Generation

i. Create a good basis R.
ii. Transform this good basis R into a bad basis Q through transpose and multiplication operation on R.
iii. Publish bad basis Q as public basis and keep good basis R as private basis.
iv. Use RSA and send the public key with bad basis Q as a composite key.

Encryption

i. Use public basis Q and add some small plaintext vector p to it.
ii. Compute new vector C’ = Q + p as the intermediate cipher text.
iii. Perform RSA encryption on C’ and compute final cipher text C.

Decryption

i. Perform RSA decryption and compute C’ from cipher text C.
ii. Using the private basis, compute its inverse and inverse of R transpose to generate vector w.
iii. Subtract this lattice vector from the cipher text to give the plaintext p = C- w.

IV. Result

4.1 Key Generation

Public key and Private key are generated by taking a random lattice and reducing it to reduced row echelon form. Implementation of RSA is done in the source code to maintain secrecy from user. (Refer Fig.1)

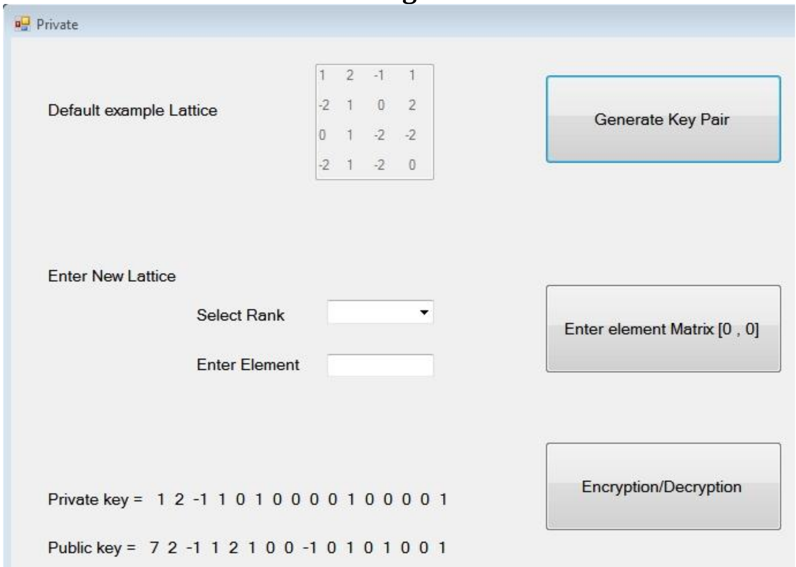
4.2 Encryption

Encryption is done by entering the plain text and inserting key to the module. Cipher text is exported to a file ready to send via unsecured communication channel. (Refer Fig.2)

4.3 Decryption

File containing cipher text is accessed by the module and using user’s private key the plaintext is obtained. (Refer Fig.3)

V. Figures

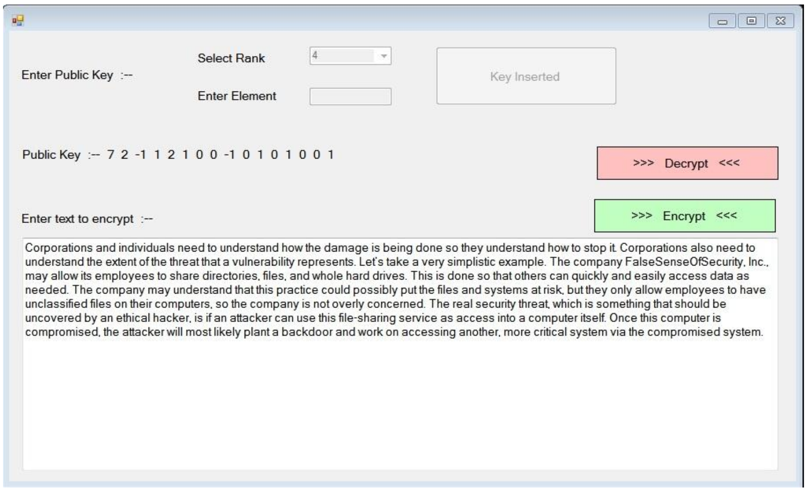


1	2	-1	1
-2	1	0	2
0	1	-2	-2
-2	1	-2	0

Private key = 1 2 -1 1 0 1 0 0 0 0 1 0 0 0 0 1

Public key = 7 2 -1 1 2 1 0 0 -1 0 1 0 1 0 0 1

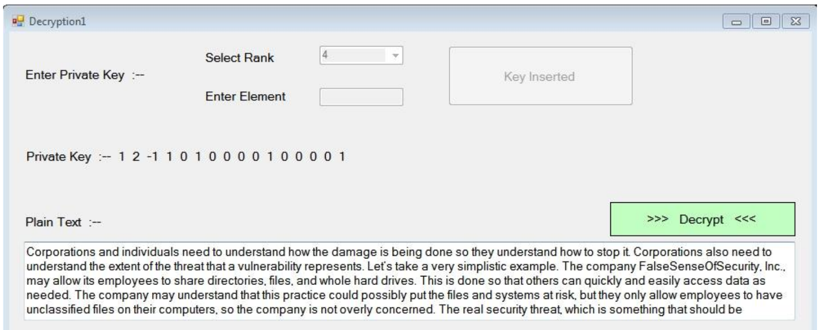
Fig. 1 key Generation Module



Public Key :-- 7 2 -1 1 2 1 0 0 -1 0 1 0 1 0 0 1

Corporations and individuals need to understand how the damage is being done so they understand how to stop it. Corporations also need to understand the extent of the threat that a vulnerability represents. Let's take a very simplistic example. The company FalseSenseOfSecurity, Inc., may allow its employees to share directories, files, and whole hard drives. This is done so that others can quickly and easily access data as needed. The company may understand that this practice could possibly put the files and systems at risk, but they only allow employees to have unclassified files on their computers, so the company is not overly concerned. The real security threat, which is something that should be uncovered by an ethical hacker, is if an attacker can use this file-sharing service as access into a computer itself. Once this computer is compromised, the attacker will most likely plant a backdoor and work on accessing another, more critical system via the compromised system.

Fig. 2 Encryption Module



Private Key :-- 1 2 -1 1 0 1 0 0 0 0 1 0 0 0 0 1

Corporations and individuals need to understand how the damage is being done so they understand how to stop it. Corporations also need to understand the extent of the threat that a vulnerability represents. Let's take a very simplistic example. The company FalseSenseOfSecurity, Inc., may allow its employees to share directories, files, and whole hard drives. This is done so that others can quickly and easily access data as needed. The company may understand that this practice could possibly put the files and systems at risk, but they only allow employees to have unclassified files on their computers, so the company is not overly concerned. The real security threat, which is something that should be

Fig. 3 Decryption Module

VI. Conclusion

The proposed system works fine for a large plain text. The advantages of this system over previous GGH encryption scheme is that if a single administrator is managing a large number of different keys then it reduces overhead of saving unimodular matrix along with the Private key. The second advantage is to eliminate the error due to Babai roundoff algorithm. Although this scheme works well but there are limitations of this system also. Introduction of RSA increases the complexity of algorithm but it also decreases the speed of encryption. The system could be implemented in the environment where secrecy is main concern rather than speed i.e. in faster computing environment.

References

Journal Papers:

[1]. Daniele Micciancio and Oded Regev, Lattice-Based Cryptography, Springer Berlin Heidelberg, DOI 10.1007/978-3-540-88702-7_5, 2009, 147-191.

[2]. O. Goldreich, S. Goldwasser, and S. Halevi, Public-key cryptosystems from lattice reduction problem, Electronic Colloquium on Computational Complexity (ECCC), 3(56), 1996.

[3]. R. J. McEliece, A public-key cryptosystem based on algebraic coding theor, Deep Space Network Progress Report, 44:114-116, January 1978.

[4]. P. Q. Nguyen, Cryptanalysis of the Goldreich-Goldwasser-Halevi cryptosystem from crypto '97, In Advances in Cryptology - Crypto 1999, Lecture Notes in Computer Science 1666, Springer-Verlag, 1999, 288-304.

Theses:

[5]. Micheal Rose, Lattice-Based Cryptography: A Practical Implementation, M.Sc. Thesis, School of Computer Science and Software Engineering, University of Wollongong, New South Wales, Australia, 2011.