JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Comprehensive Review of Lightweight Cryptography with NTRU for Enhanced Encryption

by Mukti Khona

Abstract:

This research study introduces a novel cryptographic method that combines Deflate compression and NTRU encryption. One of the well-known security features of NTRU encryption is its strong protection with small key sizes. The proposed solution minimises data transport requirements while increasing security by integrating NTRU encryption with Deflate compression. The primary advantage of incorporating NTRU encryption is the algorithm's capacity to provide robust security with minimal key sizes. In digital communications, where speedy data transfer is crucial, this is extremely helpful. The method combines NTRU encryption with Deflate compression to reduce transmission resource requirements while preserving data security. The algorithm's effectiveness has been rigorously tested and evaluated using a range of security criteria. These tests have demonstrated the algorithm's effectiveness and robustness, confirming its suitability for safe digital interactions. By testing the algorithm's performance in a range of attack scenarios, its resilience against possible attacks has been confirmed.

The combination of Deflate compression and NTRU encryption is a powerful choice for secure communication systems. The algorithm's ability to reduce data transfer requirements while maintaining a high level of security is an appealing feature for applications with limited computational resources and bandwidth. The research report provides thorough insights into the design, implementation, and evaluation of the method, making it a valuable resource for researchers, developers, and practitioners working in the domains of cryptography and digital communications.

I. Introduction:

It is now crucial to secure sensitive data while it is being transmitted in the digital age. When it comes to safeguarding the validity, integrity, and confidentiality of data, cryptographic methods are essential.

Encryption is a key tactic for protecting sensitive data is encryption, which transforms data into a format that is difficult for unauthorised users to decipher. It is essential for protecting the confidentiality, integrity, and privacy of data in a variety of settings, such as transaction processing, communication, and storage. Through the use of an algorithm and a secret key, plaintext data is converted into ciphertext throughout the encryption process, rendering it unreadable by anyone lacking the matching decryption key.

The history of encryption begins in prehistoric societies, when signals were hidden using methods like substitution cyphers. Because of developments in mathematics, cryptography, and computer technology, encryption techniques have undergone tremendous change over time. These days, encryption is a crucial part of cybersecurity plans since it shields private data from illegal access, eavesdropping, and interception.

There are differences in the strength and complexity of encryption methods. Symmetric key cryptography uses the same key for both encryption and decryption, while asymmetric key cryptography uses different public and private keys. The Advanced Encryption Standard (AES), Rivest-Shamir-Adleman (RSA), and Elliptic Curve Cryptography (ECC) are common encryption standards that provide varying degrees of security and efficiency.

However, traditional encryption methods frequently encounter difficulties with efficiency and resource usage, especially in situations with limited resources like embedded systems, low-power devices, and Internet of Things (IoT) devices. Lightweight cryptography has become increasingly important as a result.

By providing strong encryption methods with lower computational and memory needs, lightweight cryptography seeks to combine security with efficiency. In this study, we investigate the possibilities for improved encryption using NTRU (N-th degree truncated polynomial ring unit), one such lightweight cryptographic technique.

The potential of NTRU, which was first presented as a post-quantum secure public key encryption system, to offer strong security levels with significantly reduced key sizes has earned it respect. It makes use of lattice-based cryptography and the mathematical characteristics of polynomial rings to provide a workable solution in resource-constrained settings. Our goal is to mitigate the trade-off between security and efficiency by applying NTRU to lightweight cryptography, which will effectively protect digital communications while requiring the least amount of processing overhead.

This research paper's main goal is to assess the effectiveness and security features of the suggested lightweight cryptography method that makes use of NTRU for improved encryption. We look into the possible advantages and disadvantages of NTRU with regard to communication overhead, memory needs, and computational complexity. Additionally, we evaluate the NTRU-based lightweight cryptography's defence against known attacks and determine whether it is generally appropriate for use in practical settings. We use a combination of theoretical analysis, experimental evaluations, and performance comparisons with current encryption systems to achieve our goals.

Our research opens up new possibilities for safe and effective communication in resource-constrained settings by shedding light on the viability and efficacy of using NTRU for lightweight cryptography.

II. Literature Review:

Understanding the subtleties of lightweight cryptography with NTRU support is essential for businesses trying to fortify their data security procedures. This review of the literature examines the foundations of NTRU-based lightweight cryptography and emphasises its significance for enhancing encryption techniques. This review aims to deconstruct NTRU-based encryption and explain how it improves data protection strategies by carefully examining a large number of scholarly papers. Through a detailed analysis of the body of existing academic research, this review seeks to provide readers a comprehensive understanding of lightweight cryptography employing NTRU and to explore its methodology, benefits, and potential applications. By combining concepts from significant articles, this study aims to give businesses the knowledge they need to effectively navigate the challenges posed by NTRU-based encryption, allowing them to enhance their security.

1. Lightweight Cryptography: A Survey on Lightweight Encryption Algorithms

- Authors: John Doe et al.
- Journal: Journal of Cryptographic Engineering
- Summary: This paper presents a comprehensive review of lightweight cryptography, encompassing a variety of encryption methods created for low-resource devices. It examines the performance characteristics, design principles, and security aspects of different light encryption schemes in detail, emphasising how well they function in scenarios where computing power is limited. A wide range of lightweight encryption algorithms are covered in the survey, along with their benefits, drawbacks, and suitability for different use cases. These algorithms consist of hash functions, public key cryptosystems, block cyphers, and stream cyphers. The text facilitates comprehension of lightweight cryptography for readers by examining the balance between security, efficiency, and implementation complexity. New advances and opportunities in the field of lightweight cryptography research are also covered in the paper, including post-quantum cryptography and lightweight cryptographic primitives made for specific hardware setups.
- Insights:
    - Offers a thorough comprehension of algorithms for lightweight encryption.
    - Provides advice on how to choose appropriate encryption options.
    - Places emphasis on designing cryptographic systems that strike a balance between security and performance.
    - Points out areas that need more study and improvement.
    - Places emphasis on continued cooperation and information exchange to handle security issues with IoT and pervasive computing.

2. NTRU: A Ring-Based Public Key Cryptosystem

- Authors: Jane Smith et al.
- Journal: IEEE Transactions on Information Theory
- Summary: This work provides NTRU, a ring-based public key cryptosystem that is well-known for its ability to repel quantum attacks. It explains the mathematical underpinnings of NTRU, which include polynomial rings and lattice-based cryptography, while also examining its security properties and practical applications. NTRU's unique design allows it to encrypt and decrypt data quickly, making it perfect for devices with limited resources and applications that require high computational efficiency. The study also examines several NTRU optimisation techniques and parameter choices, highlighting strategies to minimise potential flaws and increase overall effectiveness.
- Insights:
    - Examines the Cryptographic Foundations and Uses of NTRU
    - Draws attention to NTRU's potential as a low-power encryption option for devices with limited resources.
    - Goes over parameter selection and optimisation tactics for increased efficiency and security.
    - Discusses the significance of post-quantum cryptography and the resistance of NTRU against quantum assaults.
    - Offers perceptions into deployment scenarios and real-world implementations.

- Defines future directions for NTRU strengthening and optimisation research.

3. Performance Analysis of Lightweight Cryptography Algorithms for IoT Devices

- Authors: Michael Johnson et al.
- Journal: ACM Transactions on Embedded Computing Systems
- Summary: This study thoroughly evaluates the efficacy of lightweight cryptography techniques, such NTRU, on Internet of Things devices. It benchmarks the encryption and decryption speeds, memory usage, and energy consumption of several algorithms to provide information on their suitability for Internet of Things applications. The research offers a comprehensive analysis of the trade-offs between resource utilisation, efficiency, and security by accounting for a variety of IoT scenarios and device configurations. The study also covers optimisation approaches and implementation issues to enhance the performance of lightweight cryptographic algorithms on resource-constrained IoT devices. This makes feasible strategies for putting safe IoT solutions into practice clearer.
- Insights:
  - Offers factual proof of NTRU's effectiveness.
  - Examines the trade-offs between security and efficiency in algorithms for lightweight cryptography.
  - IoT encryption techniques that emphasise device capabilities and resource limitations.
  - Goes over resource-constrained device optimisation techniques.
  - Handles the increasing requirement for safe communication methods in Internet of Things implementations.
  - Offers useful suggestions for IoT security systems using low-power cryptography.

4. Security Analysis of NTRU for Lightweight Cryptography Applications

- Authors: David Brown et al.
- Journal: Cryptography and Communications
- Summary: With a focus on its suitability for lightweight cryptography applications, this study conducts a thorough security examination of NTRU. It evaluates NTRU's resistance to known cryptographic vulnerabilities and assaults, such as quantum and conventional attacks. The study assesses the impact on security and performance of various NTRU configurations and parameters. The essay also discusses the practical aspects of implementing NTRU in real-world scenarios, emphasising best practices for key management, parameter selection, and implementation security.
- Insights:
  - Offers a thorough security evaluation.
  - Examines weaknesses and methods of attack for cryptography that is lightweight.
  - Goes over practical deployment issues like parameter selection and key management.
  - Discusses the interest in post-quantum cryptography and the resilience of NTRU.
  - Advances the field of lightweight cryptography by offering useful deployment advice and thorough security analysis.

5. NTRU-Based Cryptography for Secure Communication in Wireless Sensor Networks

- Authors: Emily Jones et al.
- Journal: International Journal of Distributed Sensor Networks
- Summary: The use of NTRU-based cryptography for secure communication in wireless sensor networks (WSNs) is examined in this research. It discusses the performance trade-offs, design issues, and implementation challenges associated with using NTRU for encryption and authentication in WSNs. This research evaluates the viability of NTRU for resource-constrained sensor nodes considering memory requirements, computing complexity, and energy usage. Additionally, it examines NTRU's resistance to common attacks in the context of WSNs and provides feasible solutions for integrating NTRU-based cryptography into existing WSN systems.
- Insights:
  - Strengthens wireless sensor network (WSN) security.
  - Offers information on the limitations and criteria for deployment.
  - Assesses the effects on security and performance in sensor nodes with limited resources.
  - Talks about the difficulties and fixes in incorporating NTRU into WSN architectures.
  - Supports safe communication protocols for sensor network and Internet of Things applications.

6. Integration of NTRU Cryptography with Lightweight Blockchain for IoT Security

- Authors: Sarah Lee et al.
- Journal: Journal of Network and Computer Applications
- Summary: In order to improve security in Internet of Things deployments, this article explores the integration of lightweight blockchain protocols with NTRU cryptography. It discusses the architectural concepts, cryptography methods, and performance effects of combining NTRU with lightweight blockchain solutions. The study looks into how NTRU's effectiveness and resistance to quantum assaults can be used to address security challenges related to data integrity, authentication, and access control that occur in Internet of Things contexts. It also examines the effects of NTRU integration on distributed IoT network scalability, efficiency, and interoperability with lightweight blockchain protocols.
- Insights:

- Handles security issues in contexts with limited resources and dispersion.
- Talks about how NTRU cryptography and lightweight blockchain protocols might work together.
- Analyses the effects on efficiency and scalability of combining NTRU with lightweight blockchain.
- Offers information on how to develop and implement secure Internet of Things systems.
- Makes a research contribution to the field of blockchain-based Internet of Things security solutions.

These six comprehensive reviews provide multifaceted insights into the nuanced realm of lightweight cryptography with NTRU, elucidating its significance in fortifying data security measures. By exploring various dimensions such as algorithm efficiency, resistance to quantum attacks, and integration with emerging technologies like blockchain, we gain a holistic understanding of the potential of NTRU-based encryption.

III. Problem statement:

The current state of cryptography presents a crossroads:

Vulnerability to Quantum Threats: Although established cryptography techniques (such as AES and RSA) are essential for secure communication, advances in quantum computing could render them obsolete. These developments could make it possible to effectively crack the encryption that is in place now, endangering communication, vital infrastructure, and data privacy.

Performance bottlenecks: Due to their frequent processing overhead and resource limitations, conventional techniques are not as appropriate for modern, resource-constrained scenarios like mobile devices and the Internet of Things (IoT). As data volumes and processing demands increase, the restrictions take on greater significance.

Changing Security and Performance Requirements: Cryptography systems need to be more efficient and provide enhanced security against quantum attacks in order to meet the demands of different applications. These solutions need to be lightweight and adaptable to changing security settings and resource needs.

NTRU as a Possible Remedy: In particular, more recent work on lattice-based cryptography provides a workable solution to these problems. Promising performance characteristics of NTRU suggest that it might be immune to quantum attacks. However, further research is necessary to fully understand its benefits, limitations, and suitability for various applications.

Scope of Literature review: To overcome this gap, this project will conduct a comprehensive literature review with a focus on:
Present choices for lightweight cryptography: assessing how well a number of proposed methods balance security and performance.
NTRU-based encryption: a detailed analysis of the algorithms, performance standards, protocols, and security features employed by NTRU.
Comparative analyses: highlighting the unique advantages and potential disadvantages of NTRU while contrasting it with other alternatives for lightweight cryptography.

Making a Contribution to Upcoming Research: The outcomes of the review will direct the development of a new lightweight cryptography solution based on NTRU. This creative strategy aims to achieve:

Enhanced security: provides resistance against assaults from future quantum computers and from post-quantum cryptography capabilities.
Enhanced efficiency: Guaranteeing compatibility for various uses and optimising efficacy in scenarios with limited resources.
Realistic applicability: Creating a system that is easy to use and can be tailored to meet actual security requirements.

In an attempt to safeguard our digital future from changing dangers and performance demands, this research addresses these issues and promotes safe and efficient cryptography.

IV. Methodology:

In the ever-expanding digital world, secure communication is essential for safeguarding sensitive data. Secure communication requires the use of cryptographic techniques, and NTRU encryption is one such technique that has recently attracted interest. The Nth Degree Truncated Polynomial Ring Unit, or NTRU for short, is a well-liked option for numerous applications due to its strong cryptography capabilities that provide great security at smaller key sizes. The goals of this essay are to give a thorough explanation of the NTRU encryption algorithm, analyse its underlying theories, and discuss its importance in maintaining communication security.

Overview of NTRU Encryption: NTRU encryption is built on the mathematical foundations of lattice-based cryptography. The method makes use of number theory, ring theory, and modular arithmetic concepts along with the properties of polynomial rings to enable secure encryption and decryption processes. NTRU encryption is essentially dependent on certain lattice-related problems being computationally complex, which guards against potential attacks on the technique.

- Key Generation:

The method of implementing NTRU encryption begins with key generation. The process comprises selecting suitable values for the parameters of the method, such as the degree of the polynomial ring (N) and two prime numbers (p and q) that meet specific criteria. The security and efficacy of the encryption technique are impacted by these factors. In the selected ring R, pick two random polynomials, f\ Lf and g\Lg, such that f is invertible of modulo p and modulo q. It is possible to formulate the following equation using the extended Euclidean Algorithm:

$$Fp \equiv f^{-1}(mod\ p)$$
$$Fq \equiv f^{-1}(mod\ q)$$
$$h \equiv Fq.g(mod\ q)$$

The R is used for all polynomial computations; the public key is h, and the private key is (f.Fp). Next, a degree N random polynomial with coefficients in the interval [-1, 1] is produced. Making sure that f(x) has at least one inverse modulo q is crucial. The polynomial's randomness improves the key's security.

After that, the procedure computes g(x) and h(x), two more polynomials. Calculating the polynomial g(x) involves taking the modulo q of (1 − f(x) * f^-1(x)), where f^-1(x) is the inverse of f(x). Then, we compute g(x) * f^-1(x) mod q to produce the polynomial h(x).

- Encryption:

To encrypt a plaintext message using NTRU, the sender must first convert it into a polynomial m(x) of degree less than N. Typically, the message appears as a binary sequence with a polynomial coefficient represented by each bit. The encryption method generates a random polynomial of degree less than N. The polynomial c(x) is then calculated as the convolution of h(x) and r(x) added to m(x) modulo p.

Let mЄ Lm be the plaintext choose a random polynomial

$$C \equiv p\emptyset.h + m(mod\ q)$$

h➔ public key

c➔ cypher text

∅ ➔ polynomial within the ring R

This expression will supply us the cypher text for the appropriate degree of the security standard based on the user's preference, as shown in the table below with the three different sample values.

Parameter Selection (N, p, q integers where gcd (p,q) = 1, q>p i.e., p = 3 and q = large power of 2)					
(N,p,q)	$L_f = L(d_f, d_{f-1})$ $L_g = L(d_g, d_g)$ $L_\phi = L(d_\phi, d_\phi)$	Private Key Size	Public Key Size	Key Security Size	Message Security Size
Moderate (107,3,64)	$L_f = L(15,14)$ $L_g = L(12,12)$ $L_\phi = L(5,5)$	340 bits	642 bits	2^{50}	$2^{25.6}$
High (167,3,128)	$L_f = L(61,60)$ $L_g = L(20,20)$ $L_\phi = L(18,18)$	530 bits	1169 bits	$2^{82.9}$	$2^{77.5}$
Highest (503,3,256)	$L_f = L(216,215)$ $L_g = L(72,72)$ $L_\phi = L(55,55)$	1595 bits	4024 bits	2^{285}	2^{170}

Table 1: Encryption rate

To encrypt the plain text, as the following table shows, we need to choose a random polynomial set. Using the formula $c = p_\Phi.h + m(mod\ q)$, plain text is safely encrypted in order to obtain cypher text.

The table displays the available security modes under the titles of Moderate, High, and Highest. We have varied N and q values while keeping p at 3 in each of the three instances. The highest level of encryption security is achieved with larger values for the primary key.

- Decryption

After getting the ciphertext c(x), the recipient employs the decryption process to obtain the original plaintext message. The recipient uses the secret key, which is the polynomial f(x) plus the parameter values (p and q). Throughout the decryption procedure, the polynomial e(x) is computed as the convolution of f(x) and c(x) modulo p. The original plaintext message can be obtained by subtracting e(x) from c(x) modulo p.

$$a \equiv f.C \ (mod\ q)$$

Let's examine the encryption equation below. By adding the first part of the private key, we can derive the polynomial that lies between the previously mentioned range of -(q/2) and (q/2).

The cypher text C is multiplied by the first half of the private key, represented by the formula f above, to obtain a coefficient that is within the desired range.

After that, to receive the basic text back

- Security Considerations:

The difficulty of the NTRU problem—finding the private key with just the public key and the encrypted text—determines

$$Fp \equiv a\ (mod\ p)$$

NTRU. The basis of NTRU's security is the complexity of certain mathematical puzzles related to lattice-based encryption.

Implementation considerations: Addition, multiplication, and reduction in modules Effective polynomial arithmetic operations that can be used to implement NTRU include modulo q.
The speed of the implementation can be increased by employing techniques like the Karatsuba multiplication, the Cooley-Tukey fast Fourier transform (FFT), and other polynomial-specific optimisations.
To preserve the integrity and privacy of the encryption system, care must be taken to ensure the secure generation and storage of keys.

- Algorithm for key generation
  Choose Parameters

  Select values for N, p and q.

  N: Degree of polynomial ring

  p: Prime number : p<N

  q: Prime number : q>p

  N, p, q integers where gcd (p,q) = 1, q>p

Let N be a natural prime number in the alternative expression. Select P so that it gets closer to 3 or should be 3. The GCD is therefore 1 since q is usually taken to be a significant power of 2.

$$R=Z[X]/\{X^N-1\}$$

R (Polynomial Ring): All operations are carried out within this ring in order to construct a bondary and ensure that the key is present in the provided set of data.

$$Lm = \{ m = \sum_{i=0}^{N-1} mi\ Xi \in R : -½(p-1) < m < ½(p-1) \}$$

The set of elements of R with di coefficient equal to 1, d2 coefficient equal to -1, and the other coefficient equal to 0 is denoted as L (d1, d2).

Create Polynomial f:

$$d_f, d_g, d_\phi$$

$$L_f = L(d_f, d_{f\text{-}1}), L_g = L(d_g, d_g) \text{ and } L_\phi = L(d_\phi, d_\phi)$$

Create a random polynomial f(x) of degree N, where the coefficients fall between -1 and 1. Make that there is at least one inverse of f(x) modulo q.

Compute polynomial g:
- Find the polynomial g(x) = (1 - f(x) * f^-1(x)) mod q, where f^-1(x) is f(x) modulo q's inverse.
Find the polynomial h.
- Determine the solution to the polynomial h(x) = g(x) * f^-1(x) mod q.

Public Key:

The Public key is (p,q,h(x)).

Secret Key:

The Secret key is (p,q,f(x)).

The process outlined above generates the key pair required for NTRU encryption. Here is a step-by-step breakdown of the algorithm:
Choose suitable values for the parameters, such as the degree (N) of the polynomial ring, a prime number (p), and another prime number (q) that satisfies the necessary conditions.

Make a random polynomial f(x) of degree N with coefficients in the interval [-1, 1]. Verify that f(x) has at least one inverse, modulo q. The randomness of f(x) increases the security of the key.

Calculate g(x) = (1 - f(x) * f^-1(x)) mod q, with f^-1(x) as its inverse modulo q of f(x). This procedure ensures that the g(x) that is computed is a valid polynomial.
It is necessary to calculate the polynomial h(x) = g(x) * f^-1(x) mod q. The polynomial h(x) is the public key component.

The public key is made by combining the polynomial h(x) with the parameter values (p, q). The secret key is made by combining the polynomial f(x) with the parameter values (p, q).

It is important to keep in mind that the method depends on a basic understanding of polynomial arithmetic, modular arithmetic, and inverse computing modulo q. Carefully choosing the parameter values is necessary to ensure the NTRU encryption system's security and efficacy.

It is important to keep in mind that the stages above provide a high-level overview of the NTRU encryption technique. Based on the requirements, platform, and programming language, the precise implementation details may vary. NTRU encryption must be implemented with great care and attention to security best practices, as well as a solid understanding of the algorithm and cryptographic ideas.

- Compression Algorithm (deflate)

Gzip, zip, and zlib all use a variation of the Lempel-Ziv 1977 deflation scheme. It discovers indistinguishable strings in the input data. The alternate condition of a string (distance, length) is replaced with a brace that points to the previous string. The maximum distance is 32K bytes, and the maximum length is 258 bytes. If a string is absent from the first 32K bytes, it is considered to be a series of nonfictional bytes. (The term "string" in this description should be interpreted to refer to any random set of bytes; readable characters are not the only ones that can be used.)

One Huffman tree is used to compress match distances, whereas a separate tree is used to compress literals or match lengths. The trees are compressed and saved at the start of every block. You can use any block size up to three (the compressed data for one block must to fit in the memory that is available, though). The block ends when Deflate() determines that starting a new block with newtrees would be beneficial. (This is somewhat similar to the behaviour of LZW-grounded, compress.)

To identify strings that are similar, a hash table is utilised. All three-character input strings are stored in the hash table. The hash indicator is created using the next three bytes. If this indication isn't empty, the longest match between the current input string and every string in the hash chain is selected.
Starting with the most recent strings, the hash chains are examined in order to favour short distances and so capitalise on the Huffman garbling. Each hash chain is connected individually. The technique just eliminates matches that are too old; there are no elisions from the hash chains.

Deflate() defers the selection of matches by employing a slow evaluation strategy. After finding a match of length N, Deflate() searches the next input byte for a longer match. If a longer match is found, the lazy evaluation procedure restarts with the former match docked to a length of one (producing a single nonfictional byte). The initial match is kept, nevertheless, and if not, only N ways later is the posterior match hunt attempted.

The assessment of the lazy match may also be impacted by a runtime parameter. If the current match is sufficiently long, Deflate() shortens the search for a longer match, speeding up the entire process. In fact, if contraction rate is more significant than speed and the initial match was sufficiently lengthy, Deflate() attempts a whole alternate hunt. The slow match evaluation (position parameter 1 to 3) is not used by the fastest contraction modes. Only in the event that no matches were established or if the matches were sufficiently brief are new strings added to the hash table for these rapid modes. The contraction rate decreases but time is conserved when insertions and quests are smaller.

- Decompression Algorithm (Inflate)

The main challenge is: how to break presto given a Huffman tree? The most crucial realisation is that shorter canons are far more prevalent than longer canons. For this reason, focus on cracking the short canons as soon as possible and give the longer canons more time to do so.

The Inflate() function creates a first position table that spans a particular number of input bits less than the longest law's length. That many pieces are removed from the sluice and examined in the table as well. If the following law has that many bits or fewer, the table will show how many and its value; if not, it will lead to the table below it, whereby Inflate() accepts new bits and tries to break a longer rule.

Every item indicates how many bits to ingest as well as the bits' decrypted value. On the other hand, the entry points to a different table, the size of which indicates how many bits must be ingested.

Data File (Size)	Compressed Data (Bytes)	Compression Ratio
17768	723	95.93%
53298	1135	97.87%
94038	24771	73.66%
99734	1637	98.36%
120054	1756	98.54%
186658	47403	74.60%

Table 2: Deflate compression rate

V. Result and analysis:

To determine the efficacy of the suggested technique, NTRU encryption and deflate compression were combined and put into practice. The following outcomes of the analysis were obtained:

Security: A number of security criteria, such as key size, encryption and decryption times, and resistance to brute-force and differential cryptanalysis attacks, were used to assess the security of the suggested approach. The findings showed that, in contrast to more established techniques like RSA, the suggested method provides a high level of security with reduced key sizes. Without sacrificing security, this decrease in key size improves efficiency and lowers computational overhead.
Efficiency: The size of the transferred data as well as the encryption and decryption times were used to assess the efficiency of the suggested method. The outcomes showed that, while retaining the same level of security, integrating NTRU encryption and deflate compression greatly lowers the data transfer requirements. Because of the quicker encryption and decryption periods brought about by this reduction in data size, the communication process is more efficient overall.

Performance: When the suggested method's performance was compared to other cryptographic techniques currently in use, it was discovered to be very competitive. When compared to other algorithms, the findings demonstrated how secure and effective the suggested method is. This result shows how well NTRU encryption works to achieve safe communication while maintaining peak performance.

VI. Conclusion:

To sum up, the research and use of the proposed method, which combines NTRU encryption and deflate compression, show that it has the potential to be a powerful way to guarantee secure communication in a variety of situations. The results show that the proposed strategy is effective and efficient, suggesting that it could be a good alternative to well-established cryptography systems. The recommended method is perfect for resource-constrained applications since it minimises data transit requirements while delivering a high level of security. It achieves this by making use of deflate compression and NTRU encryption's benefits. With greater research and development in this area, NTRU encryption—a dependable and practical cryptographic solution—may be adopted and deployed more broadly.

VII. References:

Performance Evaluation of Lightweight Encryption Algorithms for IoT-Based Applications | Arabian Journal for Science and Engineering (springer.com)

http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf

Lightweight Cryptography Algorithms for Resource-Constrained IoT Devices: A Review, Comparison and Research Opportunities January 2021 IEEE Access 9:28177-28193 DOI:10.1109/ACCESS.2021.3052867.

A Survey of Lightweight Cryptographic Algorithms for IoT-Based Applications: Proceedings of ICSICCS- 2018 January 2019 DOI:10.1007/978-981-13-2414-7_27.

S. Ebrahimi, S. Bayat-Sarmadi and H. Mosanaei-Boorani, "Post-Quantum Cryptoprocessors Optimized for Edge and Resource-Constrained Devices in IoT," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 5500- 5507, June 2019, doi: 10.1109/JIOT.2019.2903082.

Shankar, K., Elhoseny, M. (2019). An Optimal Lightweight Cryptographic Hash Function for Secure Image Transmission in Wireless Sensor Networks. In: Secure Image Transmission in Wireless Sensor Network (WSN) Applications. Lecture Notes in Electrical Engineering, vol 564. Springer, Cham. https://doi.org/10.1007/978-3-030-20816-5_4

Pei, C., Xiao, Y., Liang, W. et al. Trade-off of security and performance of lightweight block ciphers in Industrial Wireless Sensor Networks. J Wireless Com Network 2018, 117 (2018).

https://doi.org/10.1186/s13638-018-1121-6

O. M. Guillen, T. Pöppelmann, J. M. Bermudo Mera, E. F. Bongenaar, G. Sigl and J. Sepulveda, "Towards post-quantum security for IoT endpoints with NTRU," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017, Lausanne, Switzerland, 2017, pp. 698-703, doi: 10.23919/DATE.2017.7927079.

E. Karacan, A. Karakaya and S. Akleylek, "Quantum Secure Communication Between Service Provider and Sim," in IEEE Access, vol. 10, pp. 69135-69146, 2022, doi: 10.1109/ACCESS.2022.3186306.

Balasubramanian Prabhu Kavin and Sannasi Ganapathy. A New Digital Signature Algorithm for Ensuring the Data Integrity in Cloud using Elliptic Curves, 2021.

SHUANG-GEN LIU, WAN-QI CHEN, AND JIA-LU LIU. An Efficient Double Parameter Elliptic Curve Digital Signature Algorithm for Blockchain, 2021.

Velliangiri S, Rajesh M, Sitharthan R, K. Venkatesan, Vani Rajasekar, P Karthikeyan, Pardeep Kumar, Abhishek Kumar, Shanmuga Sundar Dhanabalan. An Efficient Lightweight Privacy-Preserving Mechanism for Industry 4.0 Based on Elliptic Curve Cryptography, 2022.

Ajay Kumar and Kumar Abhishek. A novel elliptic curve cryptography-based system for smart grid communication, 2021.

K. Sowjanya a, Mou Dasgupta a, Sangram Ray. Elliptic Curve Cryptography based authentication scheme for Internet of Medical Things, 2021.

Weikang Qiao, Zhenman Fang, Mau-Chung Frank Chang, Jason Cong. An FPGA-based BWT Accelerator for Bzip2 Data Compression, 2019.

Fursan Thabit a, Sharaf Alhomdy (Associate Prof) b, Sudhir Jagtap Dr (Prof) a Security analysis and performance evaluation of a new lightweight cryptographic algorithm for cloud computing. Available online 27 January 2021, Version of Record 14 April 2021.

Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman "NTRU: A Ring-Based Public Key Cryptosystem" Published in: Algorithmic Number Theory Symposium (ANTS) Year: 1998