mathematics

Article

An NTRU-like Message Recoverable Signature Algorithm

Tingle Shen, Li Miao, Bin Hua and Shuai Li *

School of Information Engineering, Ningxia University, Yinchuan 750021, China; shentl@stu.nxu.edu.cn (T.S.); limiao_smile@nxu.edu.cn (L.M.); huabin@nxu.edu.cn (B.H.)
* Correspondence: lis@nxu.edu.cn

Abstract: An important feature of Nyberg-Rueppel type digital signature algorithms is message recovery, this signature algorithm can recover the original information from the signature directly by the verifier in the verification phase after signing the message. However, this algorithm is currently vulnerable to quantum attacks and its security cannot be guaranteed. Number Theory Research Unit (NTRU) is an efficient public-key cryptosystem and is considered to be one of the best quantum-resistant encryption schemes. This paper proposes an NTRU-like message recoverable signature algorithm to meet the key agreement requirements in the post-quantum world. This algorithm, designed for the Internet of Things (IoT), constructs a secure system using the Group-Based Message Recoverable Signature Algorithm (NR-GTRU), by integrating a Group-Based NTRU-Like Public-Key Cryptosystem (GTRU) with an efficient Nyberg-Rueppel type of NTRU digital signature algorithm (NR-NTRU). This signature algorithm, resistant to quantum algorithm attacks, offers higher security at the cost of a slight efficiency reduction compared to traditional NTRU signature algorithms, and features Nyberg-Rueppel message recovery, making it well-suited for IoT applications.

Keywords: group; IoT; message recovery; NTRU; Nyberg-Rueppel

MSC: 20G45; 68P25; 94A60

check for updates

Citation: Shen, T.; Miao, L.; Hua, B.; Li, S. An NTRU-like Message Recoverable Signature Algorithm. *Mathematics* **2024**, *12*, 2051. https://doi.org/10.3390/math12132051

Academic Editor: Andrea Scozzari

Received: 22 May 2024
Revised: 28 June 2024
Accepted: 28 June 2024
Published: 30 June 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the IoT environment, secure communication between devices can be achieved through public key cryptographic security mechanisms. With the growing demand of IoT devices, when subjected to security threats (e.g., trespass attack, device scanning attack, active eavesdropping, spoofing attack, firmware attack, man-in-the-middle attack, Distributed denial of service attack (DDoS attack, etc.) in the smart device, it may still threaten the data security, disrupt the daily life and industrial operations, and affect the confidentiality, authentication, and integrity of the data. Therefore, security mechanisms based on public key cryptography face additional challenges in IoT environments [1,2]. The rapid development of quantum computing has triggered a series of studies showing the existence of quantum algorithms that can fully solve encryption algorithms such as RSA [3] in polynomial time. For example, Shor [4] proposed a quantum algorithm that solves large integer factorization and the Elliptic Curve Discrete Logarithm Problem (ECDLP) in polynomial time in a quantum computer. Among the existing encryption schemes, Stergiou [5] et al. proposed an efficient security model for IoT and cloud computing convergence, which is based on RSA. Similarly, Jose [6] et al. proposed an efficient IoT encryption method based on Elliptic Curve Cipher (ECC) [7]. However, due to the encryption and decryption complexity and security issues of these two schemes, they are no longer applicable to the IoT. While theoretical estimates suggest that quantum computers would need at least thousands of qubits to break existing codes, global tech giants are certainly working towards this goal [8]. IBM is currently leading the way in quantum computing hardware with its 127-qubit processor [9] and plans to surpass 1000 quantum bits by 2023 [10]. The likelihood of having scalable quantum computers in the next decade is predicted to be very high [11]. Therefore, there is an urgent need for quantum-resistant encryption and signature schemes.

Lattice cryptography are an important branch of post-quantum cryptography. Computing the Shortest Vector Problem (SVP) in a lattice is an important and difficult problem, which is the security cornerstone of contemporary lattice cryptography. Current practical and secure encryption/decryption lattice systems and digital signature lattice cryptosystems are basically designed based on the above problem. On the one hand, the lattice cryptography has the above difficult problem as the security foundation of the theoretical statute; on the other hand, the lattice cryptography has moderate consumption of space and time resources. Further, based on the lattice cryptography, it is possible to design attribute encryption, homomorphic encryption and other advanced cryptographic application algorithms. Therefore, lattice cryptography are widely recognised as the most promising branch of post-quantum cryptography. Among the lattice cryptography, the NTRU public key encryption system has been widely studied by scholars because of the simplicity of the algorithm, fast computation speed, and small storage space occupied.

Most lattice-based signature schemes have large signature sizes, which makes them unsuitable for resource-constrained IoT environments. Traditional digital signature schemes usually need to bind the message and the signature to make it easier for the verifier to verify the message, but this also incurs additional bandwidth costs, especially when the message and signature sizes are relatively large. The concept of message recovery was born based on the idea of reducing the consumption of bandwidth. In message recovery, the message is embedded in the signature. The sender sends the signature to the receiver and once the receiver receives the signature, he can perform signature verification and recover the original message from the signature [12]. In 1993, Nyberg and Ruppel modified the Digital Signature Algorithm (DSA) to support message recovery. It was the first signature scheme to support message recovery [13] and reduces the amount of information to be transmitted, and thus can save the transmission bandwidth dramatically.

In 1998, Hoffstein, Pipher, and Silverman [14] designed a fast public key cryptosystem based on the finite computation of constraint polynomials over a polynomial ring, the NTRU cryptosystem. The NTRU cryptosystem is 1.5 times faster than the ECC in hardware implementation [15]. Compared to the software implementation of RSA, NTRU is 200 times faster in key generation, nearly 3 times faster in encryption, and about 30 times faster in decryption [16]. The security of the NTRU cryptosystem is considered to be comparable to that of RSA and ECC when using the recommended parameters [17]. NTRU is not only a fast public-key cryptosystem, but also quantum-resistant.

In 2003, Hoffstein et al. also combined the GGH scheme and the NTRU lattice and proposed the NTRU signature scheme [18]. This scheme has shorter public and private keys and is more efficient, but it leaks trapdoor information during the Signature and does not give rigorous security proof.

In 2008, Gentry et al. designed a one-way function, also known as preimage sampleable functions, based on the hard problem on the lattice (Short Integer Solution (SIS) problem) and proposed a digital signature scheme based on it, the GPV08 digital signature [19]. This scheme is provably secure under the random oracle model and is resistant to adaptive chosen-message attacks. However, the GPV08 digital signature scheme is inefficient, mainly due to the inefficiency of its Preimage Gaussian sampling algorithm.

In 2008 and 2010, the NTRU encryption algorithm was standardised by the IEEE [20] and ASC X9 [21].

In 2012, Lyubashevsky proposed a lattice-based digital signature scheme based on SIS problem without using trapdoor matrices [22].

In 2013, Tian [23] et al. first introduced the concept of message recovery to lattice-based cryptography.

In 2016, NIST initiated the Post-Quantum Cryptography Project to start a worldwide call for post-quantum cryptographic algorithms with a view to laying the groundwork for subsequent standardisation [24]. After three rounds of selection, NIST announced on 5 July

2022 the final algorithms selected for standardization as well as those that require further discussion in the fourth round. The NTRU public-key cryptosystem involves only simple polynomial modular multiplication and addition/subtraction operations in the computation process, and its encryption and decryption speeds and time efficiencies outperform existing public-key cryptosystems, such as RSA, ElGamal [25], etc. under the same security level. In NIST's post-quantum cryptography scheme collection project, schemes based on the NTRU lattice occupy an important position. For example, the Falcon digital signature [26] is constructed based on the NTRU lattice and is one of the pre-standardization signatures in the standard collection of NIST post-quantum cryptography schemes.

In 2017, Faguo Wu [27] et al. present a new identity-based proxy signature scheme over an NTRU lattice with message recovery (IB-PSSMR), which is more efficient than the other existing identity-based proxy signature schemes in terms of the size of the signature and the cost of energy.

In 2019, Shuai Li [28] et al. proposed a group-based NTRU-like public key cryptosystem, GTRU, using non-Abelian poly-$\mathbb{Z}$ groups and NTRU encryption algorithms. This scheme constructs a high-performance GTRU for IoT. In the security analysis, it is demonstrated that the IoT GTRU proposed by this scheme is more secure against lattice-based attacks than the NTRU security and its efficiency exceeds that of traditional cryptographic algorithms.

To enhance the security of the Nyberg-Rueppel digital signature algorithm for IoT applications, this paper proposes a message recoverable signature algorithm based on NTRU-like. This scheme is based on the GTRU cryptosystem proposed by Shuai Li et al. [28] and exploits the message recovery property of the Nyberg-Rueppel digital signature algorithm. It has the advantage of resisting the attack of quantum algorithms, and it is a suitable digital signature scheme for IoT applications with higher security compared to the traditional NTRU public key cryptosystem and Nyberg-Rueppel digital signature algorithm.

The remainder of this paper is organised as follows. Section 2 provides an overview of the research in this article. Section 3 presents the mathematical background knowledge and theorems. Section 4 constructs the NR-GTRU signature algorithm. Section 5 compares the performance with the NR-NTRU algorithm, Section 6 analyses the security properties, Section 7 describes the advantages and challenges faced by NR-GTRU in IoT. Finally, the conclusions are presented in Section 8. Appendix A gives an example of this signature algorithm and Appendix B lists all the abbreviations and annotations that appear in this paper.

2. Research Content

Due to security reasons, traditional public key cryptosystems will be replaced by post-quantum cryptographic schemes, of which lattice-based cryptosystems are one of the best alternatives.

By studying and analyzing NTRU and digital signature algorithms in related literature, it is concluded that the security of NTRU mainly stems from the difficulty of finding the shortest vector in a given lattice, i.e., the Shortest Vector Problem (SVP): given a lattice L, find a non-zero vector v satisfying that for any non-zero vector $u \in L$, $\|v\| \leq \|u\|$.

To ensure the security of the Internet of Things, this paper focuses on the Nyberg-Rueppel digital signature scheme and the group-based NTRU-like public key cryptosystem (GTRU) and proposes a new group-based message recoverable signature scheme. This scheme uses the GTRU cryptosystem, which has message recovery properties, is suitable for the IoT, and is resistant to lattice-based attacks, making it a secure digital signature scheme.

The research focus of this paper includes the following aspects:

1. Design and improve a group-based NTRU-like signature algorithm based on the GTRU public key cryptosystem studied by Shuai Li [28] et al. It is also ensured that this signature algorithm can statute to the SVP, guaranteeing that it can withstand lattice-based attacks.

2. The signature scheme proposed by Nyberg and Rueppel [13] is adapted to a group-based NTRU-like signature algorithm to make it message-recovery friendly and resistant to some well-known forgery attacks and other related lattice attacks.
3. The efficiency and security of the signature scheme are verified and analyzed.

3. Mathematical Background

In this section, the NTRU public-key cryptosystem and poly-$\mathbb{Z}$ group are introduced.

3.1. NTRU

We denote by $\mathbb{Z}$ the set of all integers, $\mathbb{Z}$ being the infinite cyclic group under the ordinary addition. For a positive integer n, we use $\mathbb{Z}^n$ to denote the n-ary cartesian product of $\mathbb{Z}$. As the n-ary direct product of $\mathbb{Z}$, $\mathbb{Z}^n$ is the group under vector addition. For the positive integer m, denoted $\mathbb{Z}_m = \mathbb{Z}/m\mathbb{Z}$, the operation mod m defines the natural homomorphism from $\mathbb{Z}^n$ to $\mathbb{Z}_m^n$.

NTRU [29] has three parameters N, p, q, where N is used for modulo operations with polynomial coefficients less than N, and N is generally set to a prime number for better security. On the other hand, p and q must be mutually prime, and they are used to minimise the polynomial coefficients. Note that p must be less than q. NTRU also has four subsets $\mathcal{L}_f$, $\mathcal{L}_g$, $\mathcal{L}_r$, and $\mathcal{L}_m$ of $\mathbb{Z}^n$, where $\mathbb{Z}^n$ is endowed with the structure of a ring under the addition operation

$$(a_1, \ldots, a_n) + (b_1, \ldots, b_n) = (a_1 + b_1, \ldots, a_n + b_n), \tag{1}$$

and the multiplication operation, i.e., the cyclic convolution product

$$(a_1, \ldots, a_n) * (b_1, \ldots, b_n) = (c_1, \ldots, c_n), \tag{2}$$

where

$$c_k = \sum_{i+j=k+1 \ mod \ n} a_i b_j.$$

The key generation, encryption and decryption operations of NTRU are as follows.

3.1.1. Key Generation

1. First choose two random polynomials $f \in \mathcal{L}_f$, $g \in \mathcal{L}_g$ and have f_p, f_q to satisfy:

$$f * f_p = e_1 \ mod \ p,$$

$$f * f_q = e_1 \ mod \ q,$$

In some cases, f has no corresponding f_p, f_q, in which case another polynomial $f \in \mathcal{L}_f$ must be re-selected to compute f_p, f_q.

2. Compute the public key h using the chosen f, g:

$$h = f_q * pg \ mod \ q$$

3. The public key is (n, p, q, h) and the private key is (f, f_p).

3.1.2. Encrypt

1. Select the plaintext to be encrypted $m \in \mathcal{L}_m$, after randomly selecting $r \in \mathcal{L}_r$.
2. Calculate c:

$$c = h * r + m \ mod \ q$$

3.1.3. Decrypt

1. To recover the plaintext m, you need to multiply c by the private key f, and then take the modulus of q after calculating the result:

$$a = f * c \; mod \; q$$

2. Finally, using a and f_p, the plaintext m is computed:

$$m = f_p * a \; mod \; p$$

The NTRU can be decrypted correctly because $a \; mod \; m = b \; mod \; m$ and $0 \leq a, b \leq m$, which means that for $a, b, b \in \mathbb{Z}$, $a = b$.

3.2. $\mathbb{Z}^n$

Shuai Li [28] et al. construct high-performance GTRU for IoT by the use of the non-abelian poly-$\mathbb{Z}$ group $\mathbb{Z}^{n-3} \times \mathcal{H}$, where $\mathcal{H}$ is the discrete Heisenberg group. The signature algorithm constructed in this paper uses the GTRU encryption algorithm as the basis, and the following is a brief summary of the Propositions made in GTRU.

The circular convolution product of two vectors $\boldsymbol{h}$ and $\boldsymbol{r}$ in $\mathbb{Z}^n$ can be expressed as the image $\boldsymbol{H}(\boldsymbol{r})$ under the $\mathbb{Z}^n$ self-homomorphism $\boldsymbol{H}$, where $\boldsymbol{H}$ is determined by $\boldsymbol{h}$.

Moreover, the result of a modulo m operation on a vector in $\mathbb{Z}^n$ can be expressed as the image of that vector in the natural homomorphism. The following proposition shows that the result of choosing the coefficients of an n-vector on $\mathbb{Z}_m$ can be expressed as the image of this n-vector on $\mathbb{Z}_m^n$ under the function from $\mathbb{Z}_m^n$ to $\mathbb{Z}^n$ in the interval from $-m/2$ to $m/2$ [28].

Theorem 1 ([28])**.** *Let G be a group, N be a normal subgroup of G, and $\digamma_N$ be the natural homomorphism from G to G/N, i.e., $\digamma_N(x) = xN$, T_N be a transversal to N in G, i.e., $G = \bigcup_{t \in T_N} tN$. For every $x \in G$, denote the unique element in $xN \cap T_N$ by xT_N, then the function*

$$\rho T_n \; : \; G/N \to G, \rho T_N(xN) = xT_N$$

is well-defined.

Further, for every $t \in T_N$, $\rho T_N \circ \digamma_N(t) = t$.

Theorem 2 ([28])**.** *Let G be a group, N be a normal subgroup of G, $E(G)$ be the endomorphism monoid of G, $E(G/N)$ be the endomorphism monoid of G/N. Denote $E(G)_N = \{f \in E(G) : f(N) \subseteq N\}$, then the function*

$$\overline{\digamma}_N \; : \; E(G)_N \to E(G/N), \overline{\digamma}_N(f) = \overline{f}$$

is a monoid homomorphism, where $\overline{f} \in E(G/N)$ is defined by $\overline{f}(xN) = f(x)N$.

Furthermore, let $\digamma_N$ be the natural homomorphism from G to G/N, i.e., $\digamma_N(x) = xN$, then for every $f \in E(G)_N$, $\digamma_N \circ f = \overline{\digamma}_N(f) \circ \digamma_N$.

3.3. Poly-$\mathbb{Z}$ Group

Let X and Y betwo groups with group operations $\cdot$ and $*$ respectively, and $\phi : Y \to Aut(X)$ be a group homomorphism. The semi-direct product $X \rtimes_\phi Y$ of X and Y with respect to ϕ. As a set, $X \rtimes_\phi Y$ is equal to $X \times Y$, but the group operation of $X \rtimes_\phi Y$ is:

$$(x_1, y_1) \star (x_2, y_2) = (x_1 \cdot \phi(y_1)(x_2), y_1 * y_2)$$

A group G is called polycyclic if it has a subnormal series with cyclic quotients [30,31]. The Hirsch length of a polycyclic group G is the number of factors in the subnormal series of G. Specially, a polycyclic group G is called poly-$\mathbb{Z}$ if G has a subnormal series with infinite cyclic quotients.

The Hirsch dimension of a polycyclic group G corresponds to the number of factors within its subnormal series. Specifically, a polycyclic group G is called poly-$\mathbb{Z}$ when it features a subnormal series composed of quotients that are infinitely cyclic [28]. A poly-$\mathbb{Z}$ group can be created by successive infinite cyclic extensions [32]. Namely, up to isomorphism, a poly-$\mathbb{Z}$ group with group operation $\star$ is equal to

$$\mathbb{Z}^{\{\phi_i:1\leq i<n\}} = ((\mathbb{Z} \rtimes_{\phi_1} \mathbb{Z}) \rtimes_{\phi_2}) \rtimes_{\phi_n - 1} \mathbb{Z}.$$

As a set, $\mathbb{Z}^{\{\phi_i:1\leq i<n\}}$ is equal to $\mathbb{Z}^n$. In other words, any element in $\mathbb{Z}^{\{\phi_i:1\leq i<n\}}$ can be written as an n-ary integer vector. Specifically, the identity element of $\mathbb{Z}^{\{\phi_i:1\leq i<n\}}$ is denoted by $\mathbf{0}$ = (0, . . . , 0). We use $\boldsymbol{e_i}$ to denote the element in $\mathbb{Z}^{\{\phi_i:1\leq i<n\}}$ with a 1 in the i-th entry and 0's elsewhere for $1 \leq i \leq n$. For $\boldsymbol{a} \in \mathbb{Z}^{\{\phi_i:1\leq i<n\}}$, let $\boldsymbol{a}^{-1}$ denote the inverse of $\boldsymbol{a}$. For $k \in \mathbb{Z}$ and $\boldsymbol{a} \in \mathbb{Z}^{\{\phi_i:1\leq i<n\}}$, the k-th power of $\boldsymbol{a}$ is given by

$$\boldsymbol{a}^k = \begin{cases} \underbrace{\boldsymbol{a} \star \star \boldsymbol{a}}_{k\ times} & \mathrm{k} > 0 \\ \mathbf{0} & \mathrm{k} = 0 \\ (\boldsymbol{a}^{-1})^{-k} & \mathrm{k} < 0 \end{cases}$$

For $\boldsymbol{x} \in \mathbb{Z}^{\{phi_i:1\leq i<n\}}$, $f \in E(\mathbb{Z}^{\{\phi_i:1\leq i<n\}})$, we have

$$f(\boldsymbol{x}) = f(\boldsymbol{e}_1^{x_1} \star \star \boldsymbol{e}_n^{x_n}) = f(\boldsymbol{e}_1)^{x_1} \star \star f(\boldsymbol{e}_n)^{x_n},$$

thus f is entirely determined by $f(\boldsymbol{e}_i)$, $1 \leq i \leq n$. In other words, $f \in E(\mathbb{Z}^{\{\phi_i:1\leq i<n\}})$ can be written as an $n \times n$ integer matrix [28].

Let n be a positive integer at least 3, and $G_n = \mathbb{Z}^{n-1} \rtimes_\phi \mathbb{Z}$, where $\phi : \mathbb{Z} \to Aut(\mathbb{Z}^{n-1})$ is delineated by

$$\phi(a) = \begin{pmatrix} 1 & 0 & \cdots & a \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}, a \in \mathbb{Z}.$$

G_n is a special poly-$\mathbb{Z}$ group. The group operation of G_n is given by

$$x \star y = (x_1 + y_1 + x_n y_{n-1}, x_2 + y_2, \ldots, x_n + y_n), \tag{3}$$

and $G_n \simeq \mathbb{Z}^{n-3} \times$, where $\mathcal{H}$ is the discrete Heisenberg group [28,33].

Theorem 3 ([1])**.** *For $f \in E(G_n)$ with $f(\boldsymbol{e}_1) \neq 0$, there must have*

$$\begin{cases} f_{11} = f_{(n-1)(n-1)} f_{nn} - f_{(n-1)(n-1)} f_{n(n-1)}, & \\ f_{1k} = 0 & 2 \leq k \leq n, \\ f_{k(n-1)} = f_{kn} = 0 & 2 \leq k \leq n-2 \end{cases}$$

Subsequently, the endomorphism f of G_n with $f(\boldsymbol{e}_1) \neq 0$ can be written as the matrix with the form in Equation (4).

$$\left(\begin{array}{c|cccc|cc}
f_{11} & 0 & 0 & \cdots & 0 & 0 & 0 \\
ine f_{21} & f_{22} & f_{23} & \cdots & f_{2(n-2)} & 0 & 0 \\
f_{31} & f_{32} & f_{33} & \cdots & f_{3(n-2)} & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
f_{(n-2)1} & f_{(n-2)2} & f_{(n-2)3} & \cdots & f_{(n-2)(n-2)} & 0 & 0 \\
ine f_{(n-1)1} & f_{(n-1)2} & f_{(n-1)3} & \cdots & f_{(n-1)(n-2)} & f_{(n-1)(n-1)} & f_{(n-1)n} \\
f_{n1} & f_{n2} & f_{n3} & \cdots & f_{n(n-2)} & f_{n(n-1)} & f_{nn}
\end{array}\right) \tag{4}$$

Denote $E_1(G_n) = \{f \in G_n \ : \ f(\boldsymbol{e}_1) \neq 0\}$. Give the image of an element in G_n under an element in $E_1(G_n)$, and the operation of two elements in $E_1(G_n)$ in the following.

For $\boldsymbol{x} \in G_n$ and $f \in E(G_n)$ with $f(\boldsymbol{e}_1) \neq \boldsymbol{0}$, if $\boldsymbol{y} = f(\boldsymbol{x})$, then

$$y_i = \delta_i + \sum_{k=1}^{n} x_k f_{ki}, \tag{5}$$

where $\delta_i = 0$ for $2 \leq i \leq n$ and

$$\begin{aligned}
\delta_1 = {} & \frac{x_{n-1}(x_{n-1}-1)}{2} f_{(n-1)(n-1)} f_{(n-1)n} \\
& + \frac{x_n(x_n-1)}{2} f_{n(n-1)} f_{nn} \\
& + x_{n-1} x_n f_{(n-1)n} f_{n(n-1)}
\end{aligned}$$

For $f \in E(G_n)$ with $f(\boldsymbol{e}_1) \neq \boldsymbol{0}$, $g \in E(G_n)$ with $g(\boldsymbol{e}_1) \neq \boldsymbol{0}$, if $h = f \circ g$, then

$$h_{ij} = \delta_{ij} + \sum_{k=1}^{n} g_{ik} f_{kj}, \tag{6}$$

where $\delta_{ij} = 0$ for $i \neq n-1, n$. $j \neq 1$, and

$$\begin{aligned}
\delta_{i1} = {} & \frac{g_{i(n-1)}(g_{i(n-1)}-1)}{2} f_{(n-1)(n-1)} f(n-1)n \\
& + \frac{g_{in}(g_{in}-1)}{2} f_{n(n-1)} f_{nn} \\
& + g_i(n-1) g_{in} f_{(n-1)n} f_{n(n-1)}
\end{aligned}$$

For $i = n-1, n$; particularly, if $h = id$, i.e., g is the inverse of f, we have Equation (7), and $g_{11} = f_{11}^{-1}$ and $g_{i1} = f_{11}^{-1}(\delta_i 1 + \sum_{i=2}^{n} g_{ik} f_{k1})$ for $2 \leq i \leq n$ [28].

$$\begin{pmatrix} g_{22} & \cdots & g_{2n} \\ \vdots & \ddots & \vdots \\ g_{n2} & \cdots & g_{nn} \end{pmatrix} = \begin{pmatrix} f_{22} & \cdots & f2n \\ \vdots & \ddots & \vdots \\ f_{n2} & \cdots & f_{nn} \end{pmatrix}^{-1} \tag{7}$$

Theorem 4 ([28])**.** *Let the function $\digamma_m$ from the poly-$\mathbb{Z}$ group G_n to the set $\mathbb{Z}_m^n$ is defined by*

$$\digamma_m((x_1, \ldots, x_n)) \to (x_1 \ mod \ m, \ldots, x_n \ mod \ m), \tag{8}$$

then $\digamma_m$ is a group homomorphism.

Theorem 5 ([28])**.** *For $f \in E_1(G_n)$, if m is odd, then $f(ker \ \digamma_m) \subseteq ker \ \digamma_m$.*

4. An NTRU-like Message Recoverable Signature Algorithm

In this section, this paper uses GTRU [28] to extend the NTRU encryption algorithm to general groups and constructs the poly-$\mathbb{Z}$ group [14] by an infinite cyclic of successive extensions.

The signature of NR-GRTU is shown in Figure 1, where the signer uses the private key to sign the message M and sends the signature information to the verifier, who uses the public key to verify the correctness of the signature and recover the original message.

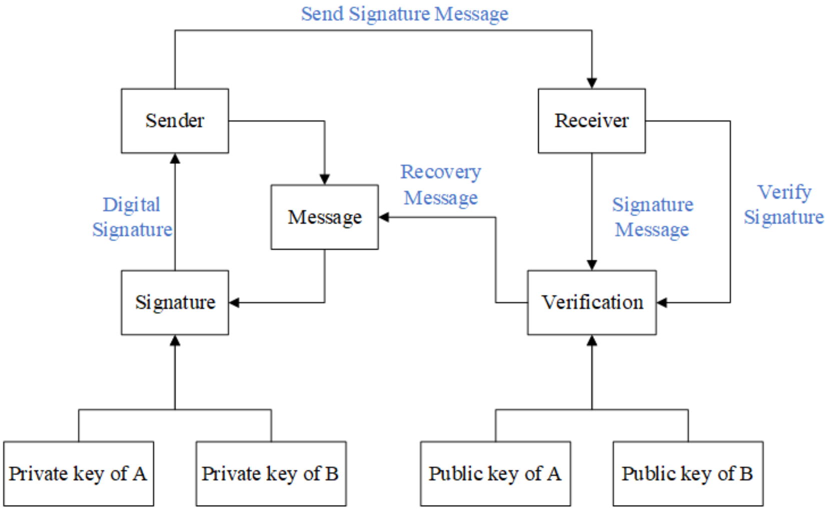

Figure 1. NR-GTRU Signature Process.

Given a group G, a normal subgroup N of G, and a transversal T_N to N in G, let $\digamma_N$, ρT_N and $\overline{\digamma}_N$ be as depicted in Theorems 1 and 2, i.e.,

$$\begin{aligned} \digamma_N &: G \to G/N, \digamma_P(g) = gP, \\ \rho T_N &: G/N \to G, \rho T_N(gN) = gT_N \in gN \cap T_N, \\ \overline{\digamma}_N &: E(G)_N \to E(G/N), \overline{\digamma}_N(f)(gN) = f(g)N. \end{aligned}$$

An NTRU-Like Message Recoverable Signature Algorithm proposed in this paper is shown below.

4.1. Parameters

N: Number, coefficients of the polynomial
p: Small modulus to reduce the coefficient
q: Large modulus to reduce the coefficient
G: Group
P: A normal subgroup of the group G.
Q: A normal subgroup of the group G.
$E(G)$: The endomorphism group of G
$\mathcal{L}_f, \mathcal{L}_g, \mathcal{L}_u, \mathcal{L}_v$: Subgroups of $E(G)$
$\mathcal{L}_m, \mathcal{L}_r, \mathcal{L}_k$: Subgroups of G

4.2. Key Generation

1. Choose $f \in \mathcal{L}_f$ and $g \in \mathcal{L}_g$ and there exist f_P, f_Q, and g_Q satisfying the

$$\overline{\digamma}_P(f \circ f_P) \circ \digamma_P = \digamma_P,$$

$$\overline{\digamma}_Q(f \circ f_Q) \circ \digamma_Q = \digamma_Q,$$

$$\overline{\digamma}_Q(g \circ g_Q) \circ \digamma_Q = \digamma_Q.$$

In some cases, f, g has no corresponding f_P, f_Q, g_Q, in which case the other polynomials $f \in \mathcal{L}_f$ and $g \in \mathcal{L}_g$ must be re-selected.

2. Choose $U \in \mathcal{L}_u$ and $v \in \mathcal{L}_v$, let $u = E \star pU$ (where E is the identity matrix), and satisfy

$$\overline{F}_Q(u \circ u_Q) \circ F_Q = F_Q$$

3. Calculates h and l

$$h = \overline{F}_Q(g \circ f_Q),$$

$$l = \overline{F}_Q(u_Q \circ v).$$

4. The private key of A is (f, g) and the public key of A is h; The private key of B is u and the public key of B is l.

4.3. Signature

1. A randomly selected $m \in \mathcal{L}_m, r \in \mathcal{L}_r$.
2. Based on Equation (3), A uses the public key l of B to calculate c

$$c = F_Q(m) \star pl \circ F_Q(r).$$

3. A calculates

$$H(h, c) = (s_p, t_p).$$

4. A randomly selected $k \in \mathcal{L}_k$.
5. A calculates s_0 and t_0

$$s_0 = s_p \star k,$$

$$t_0 = h \circ F_Q(s_0).$$

6. A calculates

$$a = g_P \circ F_P(t_p) \circ F_P(-t_0).$$

7. A calculates s and t

$$s = s_0 \star f \circ a,$$

$$t = t_0 \star g \circ a.$$

8. A sends $(c, (s, t))$ to B.

4.4. Verification

1. B calculates $t = h \circ F_Q(s)$, verifies equality, and rejects otherwise.
2. B calculates

$$H(h, c) = (s_p, t_p).$$

3. B verifies that $(s, t) = (s_p, t_p)\ mod\ p$, otherwise reject.
4. B restores message m

$$z = u \circ F_Q(c),$$

$$z = m\ (mod\ p).$$

Theorem 6. *H (h, c) is based on polynomial groups for a special kind of computation, and in the aforementioned mathematics, this paper explains how to construct this signature scheme using a non-abelian poly-$\mathbb{Z}$ group, whose $f \in E(\mathbb{Z}^{\{\phi_i:1 \leq i < n\}})$ can be represented as a matrix of $n \times n$. Through this property, this paper is designed as $H(h, c) = (s_p, t_p)$, where s_p, t_p represent the symbolic and numerical matrices of the $h \star c$ result, respectively.*

Proof of Verification 1.

$$\begin{aligned} t &= h \circ \digamma_Q(s) \\ &= h \circ \digamma_Q(s_0 \star f \circ a) \\ &= h \circ \digamma_Q(s_0) \star h \circ \digamma_Q(f \circ a) \\ &= t_0 \star h \circ \digamma_Q(f \circ a) \\ &= t_0 \star \overline{\digamma}_Q(g \circ f_Q) \circ \digamma_Q(f \circ a), \end{aligned}$$

among them

$$\begin{aligned} \overline{\digamma}_Q(g \circ f_Q) \circ \digamma_Q(f \circ a) &= \overline{\digamma}_Q(g \circ f_Q) \circ \overline{\digamma}_Q(f \circ a) \circ \digamma_Q \\ &= \overline{\digamma}_Q(g \circ f_Q \circ f \circ a) \circ \digamma_Q \\ &= \overline{\digamma}_Q(g \circ a) \circ \digamma_Q \\ &= g \circ a, \end{aligned}$$

so it can be inferred

$$\begin{aligned} t &= t_0 \star \overline{\digamma}_Q(g \circ f_Q) \circ \digamma_Q(f \circ a) \\ &= t_0 \star g \circ a \end{aligned}$$

□

Proof of Verification 4.

$$\begin{aligned} z &= u \circ \digamma_Q(c) \\ &= u \circ \digamma_Q(\digamma_Q(m) \star pl \circ \digamma_Q(r)) \\ &= \underbrace{u}_{E \star pU} \circ \digamma_Q(m) \star u \circ \digamma_Q(pl \circ \digamma_Q(r)), \end{aligned}$$

thus taking the model p for z gives:

$$z = m \ (mod\ p).$$

□

An example of NR-GTRU signature verification is given in Appendix A.

5. Performance Analysis

In this section, this paper discusses the efficiency analysis of the NR-GTRU signature algorithm and the NR-NTRU [34] signature algorithm.

5.1. Parameter

In NR-NTRU and NR-GTRU, it is mostly recommended that $p = 3$ and a prime number q is chosen to ensure security, and the value of q is guaranteed to be sufficiently large, and here in this paper, $q = 1009$ is chosen.

Since the operation of NR-GTRU is based on polynomial groups while NR-NTRU is based on normal polynomials, if the same N is chosen, the computational complexity of NR-GTRU will be N times higher than that of NR-NTRU. Therefore, for performance analysis, this paper sets NR-GTRU to take $N = 17$ and NR-NTRU to take $N = 293$ as a way to ensure that their arithmetic storage is at the same level.

5.2. Key Generation

Compared to NR-NTRU, NR-GTRU requires more computation in key generation because it uses polynomial groups instead of a single polynomial. In addition, NR-GTRU imposes strict requirements on the format of the polynomial group. Therefore, the NR-GTRU proposed in this paper is slower in generating the key compared to NR-NTRU. However, considering that the key needs to be generated only once, a slightly slower key generation speed is acceptable.

5.3. Signature and Verification

Compared to NR-NTRU, the NR-GTRU signature algorithm does not make use of norm constraints to restrict its corresponding polynomial group. Due to the unique arithmetic algorithm and polynomial group of NR-GTRU, it necessarily requires more computations to obtain the corresponding results.

5.4. Analysis

According to the proposed parameters, we implement the comparison of NR-GTRU and NR-NTRU in the following environment: Intel (R) Core (TM) i7-7700 CPU @3.60 GHz, 8GB RAM, Windows 10 operating system. For NR-GTRU and NR-NTRU, we performed Key Generation, Signature, and Verification 1000 times, respectively, and listed the average time for one signature verification in Table 1.

Table 1 demonstrates the efficiency comparison between NR-GTRU and NR-NTRU. From the table, it can be seen that in the key generation phase, NR-GTRU is much slower compared to NR-NTRU. And in the signature verification phase, the time taken by NR-NTRU is almost 1.1 times the time taken by NR-NTRU. In the verification signature phase, NR-NTRU is also slightly slower than NR-NTRU than NR-NTRU.

The performance analysis of NR-GTRU, NR-NTRU and NTRU is given in Table 2.

Table 1. This is a comparison between NR-GTRU and NR-NTRU.

	NR-GTRU	**NR-NTRU**
Parameter N	17	293
Parameter p	3	3
Parameter q	1009	1009
key (bit)	2^{12}	2^{12}
Key Generation (ms)	58.3568	37.9395
Signature (ms)	10.3836	9.4576
Verification (ms)	7.5134	6.4936

Table 2. Comparison among signature algorithms.NR-NTRU [34], NTRUsign [18] and our scheme.

	Message Recovery	**Key Size**	**Operation Speed**	**Algorithms**
NTRUsign [18]	No	Medium	Fast	NTRU
NR-NTRU [34]	No	Medium	Slower than NTRU	NTRU
NR-GTRU	Yes	Large	Slower than NR-NTRU	GTRU

6. Security Analysis

In this section, the paper continues with a discussion of NR-GTRU security.

In contrast to traditional signature verification methods, NR-GTRU sends an encrypted form of the message rather than a summary of the message. When an attacker intercepts the output of $(c,(s,t))$, they get the ciphertext (i.e., the output of c). In this case, the attacker may attack the proposed signature algorithm as an encryption algorithm only [34].

6.1. Brute Force Attacks

An attacker can try to perform a brute force attack on the NR-GTRU digital signature algorithm, which involves testing all possible private key values until the correct one is found. However, in the case of NR-GTRU, since it uses a large key space, it is usually impractical to find the right key through multiple attempts.

6.2. Lattice-Based Attacks

The security of the message is related to the difficulty of finding the shortest vector, while the security of the key is related to the difficulty of the shortest vector problem. Lattice basis reduction algorithms, especially the LLL (Lenstra-Lenstra-Lovász) algorithm [35], are the main means of attacking NTRU cryptosystems.

To explore the security of GTRUs constructed from poly-$\mathbb{Z}$ groups against lattice-based attacks, Shuai Li et al. [28] generalised the shortest vector problem on NTRU lattices. In their work, the shortest vector problem was generalised to poly-$\mathbb{Z}$ groups. In the poly-$\mathbb{Z}$ group G, the shortest vector problem is defined as follows: given a matrix f of $n \times n$ and a norm N, where f equates to a one-to-one endomorphism of the group G, a non-zero vector v must be found in the lattice $L = \{f(x) : x \in G\}$ such that $N(v) = \underset{x \in L/\{0\}}{min} N(x)$.

It can be shown that the shortest vector problem for the ordinary additive group $\mathbb{Z}^n$ is actually the original shortest vector problem. Therefore, the shortest vector problem for the poly-$\mathbb{Z}$ group serves as a generalisation of the original shortest vector problem and is at least as hard as the original problem.

6.3. Forgery and Key Recovery Attacks

Forgery attacks are the most conventional attacks on signature algorithms. For lattice-based signature algorithms, if an attacker attempts a forgery attack, they have to solve the problem of finding the approximate closest vector in the corresponding lattice [36,37], which is a known hard problem. Furthermore, the study by Hoffstein et al. [37] analyses this type of attack based on lattice bases and explores its security. Therefore, considering these factors, the signature algorithm proposed in this topic is also able to defend against this type of attack.

7. Advantages and Challenges

NR-GTRU has multiple advantages in terms of its applicability in IoT environments and can effectively ensure its security.

1. *Against quantum attacks:* The traditional signature algorithm is primarily implemented by using algorithms such as RSA, DSA and finite field cryptography. However, all these schemes are practically inadequate against quantum attacks [38]. The NR-GTRU signature algorithm is implemented on top of the GTRU encryption algorithm, and the security of GTRU is associated with the difficult problem of SVP for lattice cryptography, so it is reasonable to believe that the NR-GTRU signature algorithm can be well protected against quantum attacks.
2. *Faster execution time:* Although NR-GTRU loses some of its efficiency when compared to the traditional NTRU signature algorithm, it is still more efficient than the traditional RSA signature algorithm and the Elliptic Curve Digital Signature Algorithm (ECDSA).

However, there are still some issues that need to be effectively addressed in practical applications:

1. *Computation cost:* Due to the large number of matrix operations used in NR-GTRU and the resource constrained IoT devices, which usually have low processing power, limited memory and low computational power, this definitely increases the computational cost in IoT devices.
2. *Appropriate parameters selection:* NR-GTRU has certain parameter requirements and it is a difficult task to select the appropriate parameters and mathematical functions for a particular application.
3. *Communication cost:* NR-GTRU is a lattice-based signature algorithm that is more efficient and secure than traditional encryption and signature algorithms, but this also results in more communication bits to be processed.

4. *Side channel attacks:* Digital signature algorithms based on lattice cryptography theory can operate securely in a quantum environment, but in practice, they still face the threat of side-channel attacks such as energy analysis attacks and timing attacks.

8. Conclusions

With the widespread adoption of IoT in smart infrastructures, security and privacy have become major challenges in the last few years. Traditional signature algorithms cannot effectively address the security challenges in IoT environments as they are not resistant to quantum attacks.The paper presents an algorithm that constructs a lightweight high-performance Group-Based Message Recoverable Signature Algorithm (NR-GTRU) signature algorithm for IoT applications. It uses a combination of Group-Based NTRU-Like Public-Key Cryptosystem (GTRU) and efficient Nyberg-Rueppel type of NTRU digital signature algorithm (NR-GTRU). The developed algorithm is a quantum attack-resistant algorithm and performs well as compared to traditional algorithms, except the efficiency. However, the practical application of NR-GTRU in IoT still has certain drawbacks that need to be addressed, such as computation cost, communication cost, and side channel attacks are all issues that need to be considered, and in the future, we will continue to improve the usability of the scheme and investigate the specific application scenarios of the scheme.

Author Contributions: Conceptualization, T.S. and S.L.; methodology, T.S.; validation, L.M. and B.H.; writing—original draft preparation, T.S.; writing—review and editing, L.M., B.H. and S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Key R&D Program of Ningxia (No. 2021BEB04004).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to thank the editor and the reviewers for their valuable comments and suggestions that improved the quality of this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

In this appendix, this paper gives an example of NR-GTRU signature verification. At this first, we choose $n = 7, p = 3, q = 1009$ for parameters.

Appendix A.1. Key Generation

1. Choose $f \in \mathcal{L}_f$ and $g \in \mathcal{L}_g$ and there exist f_P, f_Q, and g_Q

$$f = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & -1 & -1 \\ -1 & 0 & 1 & -1 & 1 & 0 & 1 \end{pmatrix},$$

$$g = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 1 & 0 & 0 \\ -1 & -1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & -1 & -1 & 0 & -1 & 1 \\ 1 & 1 & 0 & 0 & -1 & 1 & 0 \end{pmatrix}.$$

2. Choose $U \in \mathcal{L}_u$ and $v \in \mathcal{L}_v$ and let $u = E \star pU$

$$u = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & -2 & 0 & 0 & 0 & 0 & 0 \\ -3 & 3 & 1 & -3 & -3 & 0 & 0 \\ 3 & 3 & 3 & -2 & 0 & 0 & 0 \\ 3 & 3 & -3 & -3 & -2 & 0 & 0 \\ -3 & 3 & 3 & 3 & 3 & -2 & 0 \\ -3 & 0 & 3 & -3 & -3 & 0 & -2 \end{pmatrix},$$

$$v = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 1 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 1 & 1 & 0 & 0 \\ -1 & -1 & 1 & -1 & -1 & -1 & 0 \\ -1 & 1 & 1 & -1 & 0 & 1 & 1 \end{pmatrix}.$$

3. Calculate h and l

$$h = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -5 & -1 & 2 & 5 & -4 & 0 & 0 \\ -1 & -1 & 0 & 1 & -1 & 0 & 0 \\ -3 & -1 & 0 & 3 & -3 & 0 & 0 \\ 4 & 2 & -1 & -4 & 3 & 0 & 0 \\ 7 & 5 & -1 & -10 & 10 & 0 & -1 \\ -4 & -1 & 1 & 6 & -6 & 1 & 0 \end{pmatrix},$$

$$l = \begin{pmatrix} 252 & 0 & 0 & 0 & 0 & 0 & 0 \\ 359 & 130 & -325 & 260 & 488 & 0 & 0 \\ 315 & 392 & 32 & -227 & 456 & 0 & 0 \\ 63 & 113 & -32 & 227 & -456 & 0 & 0 \\ 240 & -179 & 195 & -358 & -293 & 0 & 0 \\ 76 & 214 & -276 & 423 & 163 & -504 & 0 \\ 217 & -334 & 456 & -162 & 325 & 504 & 504 \end{pmatrix}.$$

4. The private key of A is (f, g) and the public key of A is h; The private key of B is u and the public key of B is l.

Appendix A.2. Signature

1. A randomly selected $m \in \mathcal{L}_m, r \in \mathcal{L}_r$

$$m = \begin{pmatrix} 1 & 0 & -1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

$$r = \begin{pmatrix} 0 & 0 & 0 & 1 & -1 & 0 & -1 \end{pmatrix}$$

2. A uses the public key l of B to calculate c

$$c = \begin{pmatrix} 261 & -141 & -32 & 224 & -454 & -502 & -503 \end{pmatrix}.$$

3. A calculates

$$H(h, c) = (s_p, t_p).$$

4. A calculates

$$s_p = \begin{pmatrix} -1 & 1 & 1 & -1 & -1 & -1 & 1 \end{pmatrix},$$
$$t_p = \begin{pmatrix} 221 & 61 & 171 & 283 & 413 & 503 & 502 \end{pmatrix}.$$

5. A randomly selected $k \in \mathcal{L}_k$

$$k = \begin{pmatrix} 0 & -3 & 0 & 3 & -3 & 3 & 0 \end{pmatrix}.$$

6. A calculates s_0 and t_0

$$s_0 = \begin{pmatrix} 2 & -2 & 1 & 2 & -4 & 2 & 1 \end{pmatrix},$$

$$t_0 = \begin{pmatrix} -3 & 0 & -1 & -1 & 3 & 1 & -2 \end{pmatrix}.$$

7. A calculates a

$$a = \begin{pmatrix} 1 & -1 & 1 & 1 & -1 & 0 & 1 \end{pmatrix}.$$

8. A calculates s and t

$$s = \begin{pmatrix} 0 & 0 & 3 & 1 & 0 & 2 & 2 \end{pmatrix}.$$

$$t = \begin{pmatrix} -4 & 4 & 0 & -2 & 2 & 2 & -2 \end{pmatrix}$$

9. A sends $(c, (s, t))$ to B.

Appendix A.3. Verification

1. B calculates $t = h \circ F_Q(s)$, verifies equality, and rejects otherwise.

$$t1 = h \circ F_Q(s) = \begin{pmatrix} -4 & 4 & 0 & -2 & 2 & 2 & -2 \end{pmatrix}.$$

2. B calculates:

$$H(h, c) = (s_p, t_p).$$

3. B verifies that $(s, t) = (s_p, t_p)\ mod\ p$, otherwise reject.
4. B restores message m

$$z = u \circ F_Q(c) = \begin{pmatrix} 16 & 9 & -4 & 4 & -2 & -5 & -3 \end{pmatrix},$$

$$z = m\ (mod\ p).$$

Appendix B

All abbreviations and annotations appearing in this paper are shown in the Table A1.

Table A1. Related abbreviations and their annotations.

Abbreviation	Annotation
DDoS	Distributed Denial of Service Attack
DSA	Digital Signature Algorithm
ECC	Elliptic Curve Cipher
ECDLP	Elliptic Curve Discrete Logarithm Problem
ECDSA	Elliptic Curve Digital Signature Algorithm
GTRU	A Group-Based NTRU-Like Public-Key Cryptosystem
IoT	Internet of Things
NR	Nyberg-Rueppel
NR-GTRU	An NTRU-Like Message Recoverable Signature Algorithm
NR-NTRU	An Efficient Nyberg-Rueppel Type of NTRU Digital Signature Algorithm
NTRU	Number Theory Research Unit
SIS	Short Integer Solution problem
SVP	Shortest Vector Problem

References

1. Ling, Z.; Luo, J.; Xu, Y.; Gao, C.; Wu, K.; Fu, X. Security vulnerabilities of internet of things: A case study of the smart plug system. *IEEE Internet Things J.* **2017**, *4*, 1899–1909. [CrossRef]
2. Das, A.K.; Zeadally, S.; He, D. Taxonomy and analysis of security protocols for internet of things. *Future Gener. Comput. Syst.* **2018**, *89*, 110–125. [CrossRef]
3. Rivest, R.L.; Shamir, A.; Adleman, L. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM* **1978**, *21*, 120–126. [CrossRef]
4. Shor, P.W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **1999**, *41*, 303–332. [CrossRef]
5. Stergiou, C.; Psannis, K.E.; Kim, B.-G.; Gupta, B. Secure integration of IoT and cloud computing. *Future Gener. Comput. Syst.* **2018**, *78*, 964–975. [CrossRef]
6. Jose, D.V.; Vijyalakshmi, A. An overview of security in internet of things. *Procedia Comput. Sci.* **2018**, *143*, 744–748. [CrossRef]
7. Koblitz, N. Elliptic curve cryptosystems. *Math. Comput.* **1987**, *48*, 203–209. [CrossRef]
8. Liu, T.; Ramachandran, G.; Jurdak, R. Post-Quantum Cryptography for Internet of Things: A Survey on Performance and Optimization. *arXiv* **2024**, arXiv:2401.17538.
9. IBM. Ibm Unveils Breakthrough 127-Qubit Quantum Processor. Nov 2021. Available online: https://newsroom.ibm.com/2021-11-16-IBM-Unveils-Breakthrough-127-Qubit-Quantum-Processor (accessed on 24 March 2023).
10. IBM. Ibm's Roadmap for Scaling Quantum Technology. 2020. Available online: https://www.ibm.com/quantum/roadmap (accessed on 24 March 2023).
11. Mosca, M. Cybersecurity in an era with quantum computers: Will we be ready? *IEEE Secur. Priv.* **2018**, *16*, 38–41. [CrossRef]
12. Lu, X.; Wen, Q.; Yin, W.; Liang, K.; Jin, Z.; Panaousis, E.; Chen, J. Quantum-resistant identity-based signature with message recovery and proxy delegation. *Symmetry* **2019**, *11*, 272. [CrossRef]
13. Nyberg, K.; Rueppel, R.A. A new signature scheme based on the DSA giving message recovery. In Proceedings of the 1st ACM Conference on Computer and Communications Security, Fairfax, VA, USA, 3–5 November 1993; pp. 58–61.
14. Hoffstein, J.; Pipher, J.; Silverman, J.H. NTRU: A ring-based public key cryptosystem. In Proceedings of the International Algorithmic Number Theory Symposium, Portland, OR, USA, 21–25 June 1998; pp. 267–288.
15. Kamal, A.A.; Youssef, A.M. An FPGA implementation of the NTRUEncrypt cryptosystem. In Proceedings of the 2009 International Conference on Microelectronics-ICM, Marrakech, Morocco, 19–22 December 2009; pp. 209–212.
16. Shen, X.; Du, Z.; Chen, R. Research on NTRU algorithm for mobile java security. In Proceedings of the 2009 International Conference on Scalable Computing and Communications; Eighth International Conference on Embedded Computing, Dalian, China, 25–27 September 2009; pp. 366–369.
17. Howgrave-Graham, N.; Silverman, J.H.; Whyte, W. Choosing parameter sets for NTRUEncrypt with NAEP and SVES-3. In Proceedings of the Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, 14–18 February 2005; pp. 118–135.
18. Hoffstein, J.; Howgrave-Graham, N.; Pipher, J.; Silverman, J.H.; Whyte, W. NTRUSIGN: Digital signatures using the NTRU lattice. In Proceedings of the Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, 13–17 April 2003; pp. 122–140.
19. Gentry, C.; Peikert, C.; Vaikuntanathan, V. Trapdoors for hard lattices and new cryptographic constructions. In Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing, Victoria, BC, Canada, 17–20 May 2008; pp. 197–206.
20. *IEEE Std 1363.1-2008*; IEEE Standard Specification for Public Key Cryptographic Techniques Based on Hard Problems over Lattices. IEEE: New York, NY, USA, 2009; pp. 1–81. [CrossRef]
21. *ANSI X9.98-2010*; Lattice-Based Polynomial Public Key Establishment Algorithm for the Financial Services Industry. Accredited Standards Committee, Inc.: Annapolis, MD, USA, 2010.
22. Lyubashevsky, V. Lattice signatures without trapdoors. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, 15–19 April 2012; pp. 738–755.
23. Tian, M.; Huang, L. Lattice-based message recovery signature schemes. *Int. J. Electron. Secur. Digit. Forensics* **2013**, *5*, 257–269. [CrossRef]
24. NIST. *Post-Quantum Cryptography: Proposed Requirements and Evaluation Criteria*; NIST: Gaithersburg, MD, USA, 2016.
25. ElGamal, T. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inf. Theory* **1985**, *31*, 469–472. [CrossRef]
26. Fouque, P.-A.; Hoffstein, J.; Kirchner, P.; Lyubashevsky, V.; Pornin, T.; Prest, T.; Ricosset, T.; Seiler, G.; Whyte, W.; Zhang, Z. Falcon: Fast-Fourier lattice-based compact signatures over NTRU. *Submiss. Nist'S-Post-Quantum Cryptogr. Stand. Process.* **2018**, *36*, 1–75.
27. Wu, F.; Yao, W.; Zhang, X.; Zheng, Z. An Efficient Lattice-Based Proxy Signature with Message Recovery. In Proceedings of the International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage, Guangzhou, China, 12–15 December 2017; Volume 10656, pp. 321–331.
28. Shuai, L.; Xu, H.; Miao, L.; Zhou, X. A Group-based NTRU-like Public-key Cryptosystem for IoT. *IEEE Access* **2019**, *7*, 75732–75740. [CrossRef]
29. Singh, S.; Padhye, S. Generalisations of NTRU cryptosystem. *Secur. Commun. Netw.* **2016**, *9*, 6315–6334. [CrossRef]
30. Segal, D. *Polycyclic Groups*; Cambridge University Press: Cambridge, UK, 2005.
31. Gebhardt, V. Efficient collection in infinite polycyclic groups. *J. Symb. Comput.* **2002**, *34*, 213–228. [CrossRef]

32. Cavallo, B. *Algorithmic Properties of Poly-Z Groups and Secret Sharing Using Non-Commutative Groups*; City University of New York: New York, NY, USA, 2015.
33. Hall, B.C.; Hall, B.C. *Lie Groups, Lie Algebras, and Representations*; Springer: Berlin/Heidelberg, Germany, 2013.
34. Elverdi, F.; Akleylek, S.; Kirlar, B.B. Efficient Nyberg-Rueppel type of NTRU digital signature algorithm. *Turk. J. Math.* **2022**, *46*, 59–70.
35. Lenstra, A.K.; Lenstra, H.W.; Lovász, L. Factoring polynomials with rational coefficients. *Math. Ann.* **1982**, *261*, 515–534. [CrossRef]
36. Schanck, J. Practical Lattice Cryptosystems: NTRUEncrypt and NTRUMLS. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
37. Hoffstein, J.; Pipher, J.; Schanck, J.M.; Silverman, J.H.; Whyte, W. Transcript secure signatures based on modular lattices. In Proceedings of the Post-Quantum Cryptography: 6th International Workshop, PQCrypto 2014, Waterloo, ON, Canada, 1–3 October 2014; Proceedings 6; Springer: Cham, Switzerland, 2014; pp. 142–159.
38. Perlner, R.A.; Cooper, D.A. Quantum resistant public key cryptography: A survey. In Proceedings of the 8th Symposium on Identity and Trust on the Internet, Gaithersburg, MD, USA, 14–16 April 2009; ACM: New York, NY, USA, 2009; pp. 85–93.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.