Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 7s (2024)

Quantum Resistant Cryptosystem-Based Security Protocol for 5G Network

Hitesh T Loriya[1], Divyesh R Keraliya[2], Rahul D Mehta[3]

[1]Electronics and Communication Engineering Department, L E College - Morbi, hitesh_loriya@yahoo.co.in
[2]Electronics and Communication Engineering Department, GEC - Rajkot, drkeraliya@gmail.com
[3]Electronics and Communication Engineering Department, GEC - Rajkot, rdmehta@hotmail.com

Article History:

Received: *03-05-2024*

Revised: *05-07-2024*

Accepted: *25-07-2024*

Abstract

Security is crucial for wireless communication networks, especially as quantum computing advances rapidly. It won't be long before quantum attacks become feasible, potentially crippling large wireless networks within minutes. Current methods for securing connections and transactions—such as keys, certificates, and data—could be compromised by quantum computers. One concern is the use of fake base stations with stronger signal strengths to lure users into connecting with them in wireless communication network. A quantum-powered attacker could easily break traditional encryption algorithms and launch various attacks almost instantly. An example of this is the "bidding down" attack, where an attacker convinces both the user and network entities that security features are not upheld, despite their actual presence. This type of attack exploits the security weaknesses of older mobile networks. This paper explores security advancements and challenges related to contemporary public key cryptography, including RSA algorithms based on factorization and discrete logarithm problems with Diffie-Hellman and Elliptic-Curve Cryptography (ECC). While these methods provide adequate protection today, quantum computers could potentially break these algorithms or weaken cryptographic keys and hashes within minutes. Quantum cryptography, or quantum-resistant cryptography, objectives to develop algorithms and protocols resilient to quantum computing threats. This paper examines security vulnerabilities in wireless communication networks, focusing on key confirmation and authentication mechanisms. We propose a robust authentication and key agreement protocol for 5G networks using quantum-resistant cryptography. Our proposed protocol, verified with a verification tool, enhances the security of the authentication and key agreement procedures in wireless communication networks.

Keywords: Security, Quantum attack, Cryptosystem, NTRU

1. Introduction

Cryptography involves the study of systems designed to securely transfer information among users, combining principles from engineering and mathematics. Its primary applications include ensuring authentication, confidentiality, key distribution and integrity. Two main types of cryptosystems used for secure information transfer are private key cryptosystems and public key cryptosystems. In the context of 5G networks, the 3GPP defines a security architecture [1] that employs cryptosystems to protect subscriber information exchanged between network entities. Quantum computing, which utilizes quantum bits or qubits, represents a significant departure from traditional computing. Unlike classical computers, which process information sequentially, quantum computers can handle multiple

pieces of information simultaneously, producing numerous possible solutions at once. There are five prominent public key cryptosystems: Diffie-Hellman, Elliptic Curve Cryptography (ECC), Elgamal Cryptographic System (ECS), RSA, and NTRU. Alese et al. [2] conducted a relative analysis of ECC, RSA, Elgamal, and Menezes-Vanstone Elliptic Curve Encryption algorithms, implemented in Java, evaluating their performance based on key generation, encryption, and decryption times. Giripunje et al. [3] discussed asymmetric key cryptography solutions, emphasizing the importance of secure authentication in mobile communications. This paper highlighted that ECC, requiring smaller key sizes and lower computational power compared to RSA, is more suitable for constrained devices such as mobile phones. Pallipamu et al. [4] performed a security examination of digital signature schemes, including DSA, and Elgamal, and RSA, assessing the mathematical complexity of key generation, signature verification, and overall security strengths of these algorithms.

2. Objectives

The security of information is a critical concern in wireless communication networks. Implementing public key algorithms on highly constrained devices, such as mobile phones and PDAs, necessitates the use of fast and efficient cryptosystems. It is essential to analyze various public key cryptosystems to identify the most efficient option. Among these, the NTRU public key cryptosystem stands out as the fastest, offering different levels of security with high performance even on limited resources. NTRU employs lattice-based cryptography for data encryption and decryption. Unlike many other public key cryptosystems, NTRU is resistant to quantum attacks using Shor's algorithm.

3. NTRU Cryptosystem

Here is a brief overview of the well-known RSA cryptosystem. The RSA cryptosystem, developed by Ron Rivest, Adi Shamir, and Leonard Adleman, is widely used and incorporated into web browsers from Microsoft and Netscape. The process for generating RSA public and private keys is as follows:

Key Generation Algorithm:

1. Select Two Large Prime Numbers p and q.
2. Calculate $n=p\times q$.
3. Compute $\phi(n)=(p-1)\times(q-1)$.
4. Select a public key exponent e such that $\gcd(\phi(n),e)=1$ and $1<e<\phi(n)$.
5. Determine the Private Key d such that $d\times e \bmod \phi(n)=1$
6. Public key is (n,e) and Private key is (n,d).

Encryption and Decryption:

Encryption: To encrypt a message M, compute $C = M^e \bmod n$

Decryption: To decrypt the ciphertext C, compute $M = C^d \bmod n = (M^e)^d \bmod n = (M)^{ed} \bmod n$ which retrieves the original message M. This process ensures secure communication by using the public key for encryption and the private key for decryption.

The NTRU Cryptosystem, developed by J H. Silverman, J Hoffstein, and J Pipher, is named after the N-th degree TRUncated polynomial ring [5]. NTRU's operation defines in the ring $\mathbb{Z}[x]/(x^n - 1)$ which is identified as the ring of convolution polynomials of rank "n", where "n" is a prime. Addition can be performed in O(n) and multiplication requires only $O(n^2)$ operations in the ring of convolution polynomials. NTRU utilizes three primary rings:

1. $R = \mathbb{Z}[x]/(x^n - 1)$
2. $R_p = \mathbb{Z}/p\mathbb{Z}[x]/(x^n - 1)$
3. $R_q = \mathbb{Z}/q\mathbb{Z}[x]/(x^n - 1)$

Here, p (specifying a ring $\mathbb{Z}/p\mathbb{Z}$) and q (specifying a ring $\mathbb{Z}/q\mathbb{Z}$) are positive integers used to reduce the coefficients of polynomials during encryption and decryption. Polynomials in these rings can be represented either by their coefficients or directly in polynomial form. Operations such as addition and multiplication in Rp or Rq are similar to those in R, with the result's coefficients reduced modulo p or q.

An element f (A polynomial in $\mathbb{Z}[x]/(x^n - 1)$.) of the rings R, R_p and R_q can be represented as polynomial or vector coefficients.

NTRU operate in the ring $R = \mathbb{Z}[x]/(x^n - 1)$. An element $f \in$ R represented as a polynomial,

$$f = \sum_{i=0}^{n-1} f_i x^i = [f_0, f_1, \ldots, f_{n-1}]$$

$$f * g = h$$

"g" is a polynomial in $\mathbb{Z}[x]/(q, x^n - 1)$ used with f_q (A polynomial in $\mathbb{Z}[x]/(q, x^n - 1)$). This polynomial is obtained by reducing the coefficients of f modulo q. to construct the public key

"h" is the public key, a polynomial in $\mathbb{Z}[x]/(q, x^n - 1)$.

$$h_k = \sum_{i=0}^{k} f_i g_{k-i} + \sum_{i=k+1}^{n-1} f_i g_{n+k-i} = \sum_{i+j \equiv k \bmod n} f_i g_i.$$

The variable "k" represents a security parameter that determines the level of protection against specific types of attacks, such as plaintext awareness. The public key (h) is computed by performing a cyclic convolution between f and g. The output is in the R. To minimize the coefficient, it is necessary to calculate the multiplication modulo q. This will ensure that the result is of the form $\mathbb{Z}$ [X]$/(q, x^n - 1)$. The computation of the product f * g necessitates n^2 multiplications. In NTRU, either f or g has coefficients that are predominantly 0's and ±1's, allowing for a very efficient computation of f * g. In addition, when N is a large amount, one can select n to have a high degree of divisibility by 2. In this scenario, the convolution product can be calculated using Fast Fourier Transforms, requiring O (N log N) operations. Operations involving generating functions (*G*) and hashing functions (*H*) must be defined on rings of polynomials. The *G* and *H* elements are utilized in the construction of a digital envelope within the NTRU protocol.

$p_p(n)$ = {Polynomials with coefficients modulo p and degree no greater than n−1}, and

$$[g]_p = \begin{cases} g \text{ with its cofficients reduced} \\ \text{modulo } p \text{ to the range } (-p/2, p/2). \end{cases}$$

G function and a H function are exactly defining as

$$G = p_p\,(N) \;\rightarrow\; p_p\,(N) \text{ and } H = p_p\,(N) \times\; p_p\,(N) \rightarrow\; p_p\,(K)$$

They should be computationally efficient, highly non-linear, and difficult to predict. The digital envelope in the NTRU public key cryptosystem relies on the selection of functions *G* and *H*, as well as an integer *k*. The likelihood of successfully forging a valid ciphertext is p^{-k}.

Key Creation:

Two small polynomials $f \in \mathcal{L}_f$ (Polynomials in $\mathbb{Z}[x]/(x^n - 1)$ whose coefficients satisfy d_f (Distribution of the coefficients of the polynomial f (Part of the private key)) and g (Part of public key) $\in \mathcal{L}_g$ (The set of polynomials in $\mathbb{Z}[x]/(x^n - 1)$ whose coefficients satisfy d_g (Distribution of the coefficients of the polynomial *g*)) are randomly generated to produce a pair of public and private keys. The polynomial f must fulfil the extra condition of having modular inverses both modulo q and modulo p. fp represents a polynomial in the quotient ring $\mathbb{Z}[x]/(p, x^n - 1)$, which is a crucial component of the private key. This polynomial is derived by decreasing the coefficients of f modulo p. For appropriate parameter selections, this statement holds true for the majority of f choices, and the computation of these inverses can be easily performed by modifying the Euclidean method. The inverses of f_q^{-1} and f_p^{-1} are denoted as mentioned above

$f_q^{-1} * f \equiv 1 \bmod q$ and $f_p^{-1} * f \equiv 1 \bmod p$.

f, g, f_q^{-1} and $f_p^{-1}$are kept secret and public key is calculated and issued as follow

$$h \equiv p f_q^{-1} * g \bmod q.$$

Polynomial of f is the private key and Polynomial of h is the public key

Encryption:

NTRU cryptosystem chooses plaintext *m* (polynomial in $\mathbb{Z}[x]/(p, x^n - 1)$). from the set

$$m \in p_p\,(n - k).$$

and a polynomial in $\mathbb{Z}[x]/(q, x^n - 1)$ (used with *h* to encode a message) which is random polynomial $r \in \mathcal{L}_r$ (Polynomials in $\mathbb{Z}[x]/(x^n - 1)$ whose coefficients satisfy d_r (Number of 1s and -1s used in a certain random polynomial *r*)) Cipher text (The encrypted message, a polynomial in $\mathbb{Z}[x]/(q, x^n - 1)$) is computed as follow

$$e \equiv r * h + \left[m + H\left(m, [r * h]_p\right) X^{n-k} + G\left([r * h]_p\right)\right]_p \pmod q.$$

Decryption:

The decryption process begins with multiplying (convolving) the received polynomial *e* by the private key *f*

$$a \equiv f * e \bmod q,$$

Choose the coefficients of a in the interval from $-q/2$ to $q/2$. Now treating a as a polynomial with integer coefficients, computes the temporary polynomial $t \in \mathbb{Z}[x]/(p, x^n - 1)$ by

$$t = f_p^{-1} * a \ (\bmod p),$$

Next, it calculates the two intermediate quantities

$$b \equiv e - t \ (\bmod p) \text{ and } \equiv t - G(b) \ (\bmod p),$$

And then writes *c* in the form

$c = c' + c"x^{n-k}$ with deg $(c') < n - k$ and deg(c") $< k$.

Finally, compares the quantities

$$c" \text{ and } H(c', b).$$

If the decrypted message matches the original, it is accepted c' as valid. Otherwise, the message is rejected as invalid.

NTRU decryption method is given below.

$a \equiv f * e$

$\equiv f * r * h + f * \left[m + H\left(m, [r * h]_p\right)X^{n-k} + G\left([r * h]_p\right)\right]_p \ (\bmod q)$

$\equiv f * pr * f_q^{-1} * g + f * \left[m + H\left(m, [r * h]_p\right)X^{n-k} + G\left([r * h]_p\right)\right]_p \bmod q,$

$\equiv pr * g + f * \left[m + H\left(m, [r * h]_p\right)X^{n-k} + G\left([r * h]_p\right)\right]_p \bmod q,$

By selecting a suitable parameter, it is ensured that all coefficients of the parameter lie between the range of (-q)/2 and (q)/2. This guarantees that the parameter remains unchanged even if its coefficients are lowered modulo q. This implies that by reducing the coefficients of the polynomial f * e modulo q within the range of (-q)/2 to q/2, it becomes feasible to precisely recover the polynomial.

$$a = pr * g + f * \left[m + H\left(m, [r * h]_p\right)X^{n-k} + G\left([r * h]_p\right)\right]_p \text{ in } R$$

Reducing *a* modulo *p* then gives the polynomial

$$f * \left[m + H\left(m, [r * h]_p\right)X^{n-k} + G\left([r * h]_p\right)\right]_p \text{ in } R$$

And then multiplying by f_p^{-1} produces

$$t = m + H\left(m, [r * h]_p\right)X^{n-k} + G\left([r * h]_p\right) \text{ in } \mathbb{Z}[X]/(p, X^n - 1)$$

Compute $b = e - t$ and recover $b = r * h$.

Therefore computation of *c* yields

$$c = m + H\left(m, [r * h]_p\right)X^{n-k}.$$

Accordingly, c' is the original message *m*, and $c"$ should match up with the hash

$H\big(m, [r * h]_p\big) = H(m, b)$, as noted above.

Parameter Selection:

With appropriately chosen parameters, the decryption process has a very high probability of successfully recovering the original message. However, occasional decryption failures can occur, often due to improper centering of the message. To address this, it is advisable to include a few check bits in each message block. If decryption fails due to improper centering, you can attempt to recover the message by adjusting the coefficients of $a \equiv f * e \bmod q$ within a slightly different range, for example, from $-q/2 + x$ to $q/2 + x$. where x is a small positive or negative value. If no such adjustment yields a valid decryption, a gap failure occurs, making the message difficult to decrypt. For well-chosen parameters, such failures are infrequent and can generally be considered negligible in practice.

(a) Sample spaces. The space of messages $\mathcal{L}_m$ consists of all polynomials modulo *p.*

$$\mathcal{L}_m = \Big\{ m \in R: m \text{ has coefficients lying between } -\frac{1}{2}(\mathrm{p}-1) \text{ and } \frac{1}{2}(\mathrm{p}-1) \text{ and has degree at most } N-k-1 \Big\}.$$

It is most suitable to take assume p is odd. Sets of the form are used to describe the other sample spaces.

$$\mathcal{L}(d_1, d_2) = \{ f \in R: f \text{ has } d_1 \text{cofficients equal } 1, d_2 \text{ cofficients equal } -1, \text{the rest } 0 \}.$$

With this notation, choose three positive integers $d_f, d_g, d_r$and set

$$\mathcal{L}_f = \mathcal{L}\big(d_f, d_f - 1\big), \quad \mathcal{L}_g = \mathcal{L}\big(d_g, d_g\big), and \quad \mathcal{L}_r = \mathcal{L}(d_r, d_r).$$

(Don't set $\mathcal{L}_f = \mathcal{L}\big(d_f, d_f\big)$ and a polynomial satisfying $f(1) = 0$ can never be invertible.) Notice that $f \in \mathcal{L}_f, g \in \mathcal{L}_g$and r $\in \mathcal{L}_r$ have L^2norms

$$|f|_2 = \sqrt{2d_f - 1 - n^{-1},} \qquad |g|_2 = \sqrt{2d_g,} \qquad |r|_2 = \sqrt{2d_r.}$$

Give values for d_f, d_g, d_r that allow decryption while maintaining various security levels.

(b) Define the *width* of an element $f \in R$

$$|f|_\infty = \max_{0 \le i \le N-1} \{f_i\} - \min_{0 \le i \le N-1} \{f_i\}.$$

This is a sort of L^∞ norm on *R.* Similarly, describe a *cantered* L^2norm on *R* by

$$|f|_2 = \left(\sum_{i=0}^{n-1} \big(f_i - \bar{f}\big)^2 \right)^{1/2}, where\ \bar{f} = \frac{1}{n} \sum_{i=0}^{n-1} f_i.$$

(Equivalently, $|\mathrm{f}|_2/\sqrt{\mathrm{n}}$ is the standard deviation of the coefficients of F.)

The following proposition was suggested by Don Coppersmith.

Proposition: For any $\epsilon > 0$ there are constants $\gamma_1, \gamma_2 > 0$, depending on ϵ and N, such that for randomly chosen polynomials $f, g \in R$, the probability is greater than $1 - \varepsilon$ that they satisfy

$$\gamma_1 |f|_2 |g|_2 \leq |f * g|_\infty \leq \gamma_2 |f|_2 |g|_2.$$

Of course, this proposition would be useless from a practical viewpoint if the ratio γ_2/γ_1were very large for small $\varepsilon's$. However; it turns out that even for moderately large values of *N* and very small values of ϵ, the constants γ_1, γ_2 are not at all extreme.

(c) A Decryption Criterion. Let

$$m' = \left[m + H\left(m, [r * h]_p\right) X^{n-k} + G\left([r * h]_p\right)\right]_p$$

Polynomial used $(e \equiv r * h + m' \bmod q.)$ for encryption. In order for the decryption process to operate, it is necessary that

$$|f * m' + pr * g|_\infty < q.$$

This will almost always be true if parameters are selected so that

$$|f * m'|_\infty \leq q/4 \text{ and } |pr * g|_\infty \leq q/4\,;$$

This proposes that take

$$|f|_2 |m|_2 \approx q/4\gamma_2 \text{ and } |r|_2 |g|_2 \approx q/4p\gamma_2$$

For a γ_2 matching to a small value for ϵ. For example, experimental evidence recommends that for N = 167 and N = 503, suitable values for γ_2 are 0.27 and 0.17 respectively.

NTRU is a lattice-based public-key cryptosystem recognized for its high performance and strong resistance to quantum computing attacks, making it a leading alternative to RSA and Elliptic Curve Cryptography (ECC). It is built on the "Approximate Close Lattice Vector Problem," a mathematical challenge that underpins its security. One of NTRU's defining features is its efficient polynomial multiplication, which is the most complex operation during encryption and decryption. This operation is notably faster compared to the corresponding processes in other asymmetric cryptosystems such as RSA, ElGamal, and ECC. Performance comparisons between the NTRU cryptosystem and RSA focusing on key-size, key-generation, encryption, and decryption—are illustrated in Figures 1, 2, and 3, respectively [6].

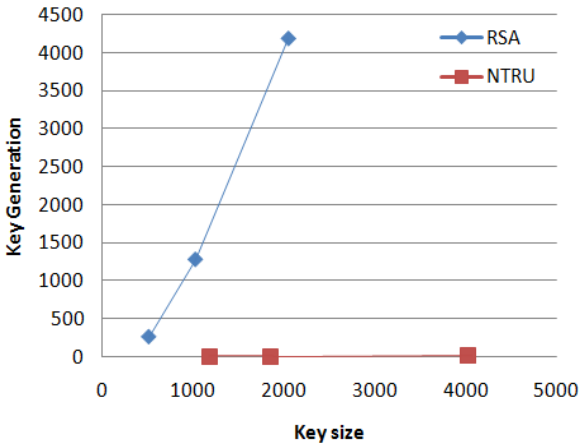

Figure-1 Key-size V/s Key-generation

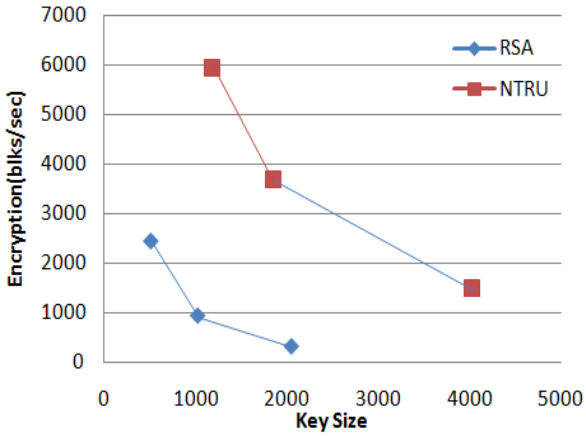

Figure-2 Key-size V/s Encryption (blocks/sec)

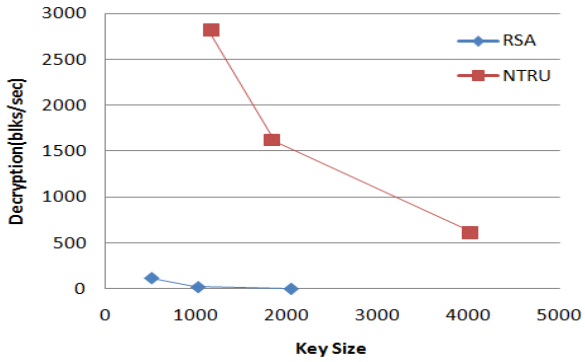

Figure-3 Key-size V/s Decryption (blocks/sec)

The performance analysis from the figures clearly indicates that NTRU outperforms RSA in key generation, encryption, and decryption. NTRU has been endorsed by two prominent standards organizations: the IEEE and the Financial-Services Industry Standards Committee-X9, specifically through the X9.98 Lattice-Based Polynomial Public-Key Establishment Algorithm. This endorsement highlights NTRU's role in securing communications within the financial sector. The X9.98 standard represents a significant advancement by enhancing the resilience of systems against both quantum and classical attacks, thereby strengthening the overall security of financial services systems.

4. Proposed AKA security protocol

The 3GPP committee has established the security architecture for the 5G network [2]. The proposed AKA protocol involves four main entities: UE (User-Equipment), SEAF (Subscription-Encryption and Authentication Function), AUSF (Authentication-Server Function), and UDM/ARPF (Unified Data Management/Authentication-Repository and Processing Function), as illustrated in Figure 4. The NTRU cryptosystem is employed to secure message exchanges between these entities.

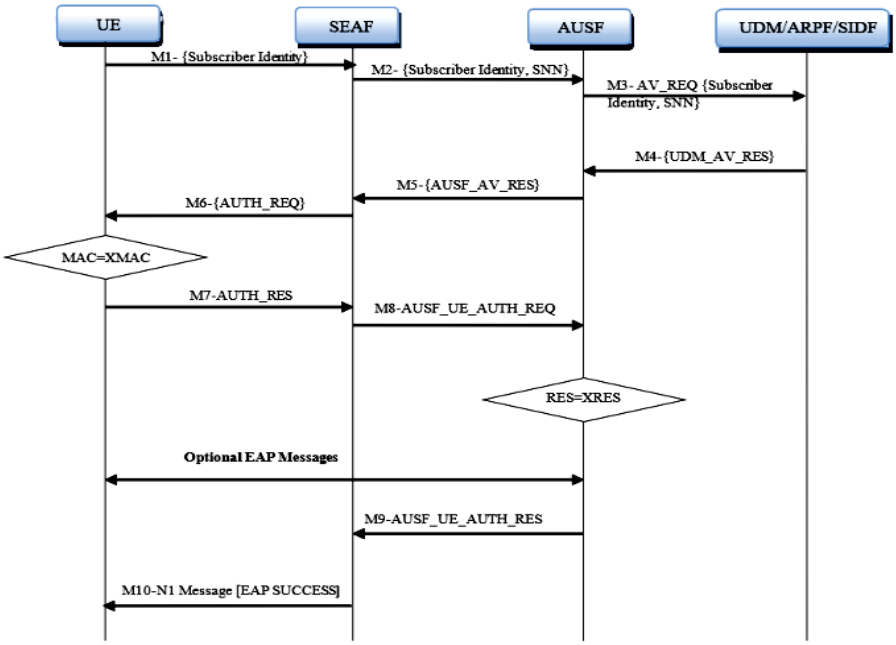

Figure 4 Proposed AKA security protocol.

In the protocol, the UE sends its subscriber identity in message-M1 to SEAF, encrypted with the NTRU public key of UDM/ARPF. SEAF then forwards a request containing the subscriber identity to AUSF and sends SNN in message-M2, also encrypted with UDM/ARPF's NTRU public key. AUSF passes the received message to UDM/ARPF as AV_REQ message-M3. UDM/ARPF decrypts the authentication request using its NTRU private key and generates an authentication response, UDM_AV_RES. This response, which contains validation information, is sent to AUSF in message-M4 (AVs), encoded with AUSF's NTRU public key. This allows AUSF to confirm the UE's identity. AUSF stores the validation vectors (AVs) and uses the NTRU public key of SEAF to send RAND and AUTN from the selected AV to SEAF in message-M5. SEAF then decrypts these messages using its NTRU private key and forwards RAND and AUTN to UE in message-M6, encrypted with UE's NTRU public key. UE verifies the MAC in AUTN by decoding it with its NTRU private key, XMAC. If the MACs match, UE calculates and returns the response RES to SEAF in message-M7. UE also computes CK and IK, similar to the calculations in UDM/ARPF. SEAF then sends the received message to AUSF as AUTH_UE_AUTH_REQ in message-M8. Upon receiving RES from UE, AUSF compares it with XRES from the AV. If they match, the authentication is deemed successful, and AUSF computes the session key K_{AUSF} from CK and IK to secure the wireless communication with UE. AUSF also derives K_{SEAF} from K_{AUSF} and sends it to SEAF in message-M9. UE independently calculates K_{AUSF} and K_{SEAF}. This process ensures that (i) UE and AUSF share the same session key K_{AUSF}, and (ii) UE and SEAF share the same session key K_{SEAF}. An optional EAP (Extensible Authentication Protocol) procedure follows message-M8, where SEAF sends an N1 Message (EAP Success) to UE with K_{gNB} in message-M10. The security of the protocol has been verified using ProVerif, a cryptographic verifier based on the Dolev-Yao model. ProVerif, which uses a modified form of pi-calculus, assesses secrecy and authentication properties. The absence of detected attacks indicates that the subscriber identity can be securely transferred across insecure channels between network entities. Even if an attacker controls the channel, the subscriber identity and other messages remain secure.

5. Discussion

This paper demonstrates that the NTRU cryptosystem is one of the fastest public key cryptosystems, offering various levels of security at high speeds. Unlike RSA and similar algorithms, which are vulnerable to quantum computing techniques capable of factoring integers and computing discrete logarithms in polynomial time, NTRU remains robust. The security of NTRU is based on the challenging problem of lattice reduction, making it resistant to quantum computing attacks. Approved for standardization by IEEE in 2009, NTRU offers strong security even on devices with constrained resources, where computing power bandwidth and storage are limited. It is known for its low CPU and battery consumption, reducing server utilization significantly. NTRU's efficiency surpasses that of other public key cryptosystems in both hardware and software implementations. The proposed AKA protocol, which leverages the NTRU cryptosystem, ensures the confidentiality of subscriber identities and other messages within wireless communication networks. We used the ProVerif tool to validate this approach. While the NTRU-based solution may introduce more delay compared to existing methods, it provides enhanced security. This highlights a trade-off between security and computational overhead, but ultimately, the improved security justifies the added computational cost.

Acknowledgement

We would like to convey our sincere thanks to Dr. K R Parmar and Dr. A. Kulshreshta for their guidance and support.

Conflicts of interest

"The authors declare that there is no conflict of interest regarding the publication of this paper."

References

[1] 3GPP, "Security Architecture and Procedures for the 5G System", 3GPP TS 33.501 version 15.2.0. Technical Report, The 3rd Generation Partnership Project.

[2] Alese, B. K., Philemon E. D., Falaki, S. O., "Comparative Analysis of Public-Key Encryption Schemes", International Journal of Engineering and Technology, Volume 2, No. 9, September 2012.

[3] Giripunje Lokesh, Nimbhorkar Sonali, "Comprehensive Security System for Mobile Network Using Elliptic Curve Cryptography over GF (p)", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 5, May 2013.

[4] Pallipamu Venkateswara Rao, K Thammi Reddy, P Suresh Varma "A Survey On Digital Signatures", International Journal of Advanced Research in Computer and Communication Engineering, Volume 3, Issue 6, June 2014

[5] Hoffstein, J., Lieman, D., Pipher, J. and Silverman, J.H., "NTRU: A Ring-Based Public Key Cryptosystem" Available online on www.ntru.org.

[6] Hoffstein, J., Lieman, D., Pipher, J. and Silverman, J.H., "NTRU: A Public Key Cryptosystem". http://grouper.ieee.org/ groups/1363/lattPK/submissions.html#NTRU1

[7] Hien Ba Nguyen, Thesis on "An Overview of the NTRU Cryptographic system", San Diego State University, 2014.

[8] Jover R P, Marojevic V, "Security and protocol exploit analysis of the 5G specifications", IEEE Access 2019, 7, 24956–24963.

[9] Ahmad I, Shahabuddin S, Kumar T, Okwuibe, J, Ylianttila M, "Security for 5G and beyond", IEEE Commun. Surv. Tutor. 2019, 21, 3682–3722.

[10] Hu X., Liu C., Liu S., Li J., Cheng X., "A vulnerability in 5G authentication protocols and its Countermeasure", IEICE Trans. Inf. Syst. 2020, 103, 1806–1809.

[11] Khan H., Martin K.M., "A survey of subscription privacy on the 5G radio interface-the past, present and future", J. Inf. Secure. Appl. 2020, 53, 102537.

[12] Blanchet, B.; Smyth, B.; Cheval, V.; Sylvestre, M. "ProVerif 2.05: Automatic Cryptographic Protocol Verifier, User Manual and Tutorial". Proverif User Manual, 2023. https://bblanche.gitlabpages.inria.fr/proverif/