A NOVEL LIGHTWEIGHT SECURE ROUTING BASED ON THE TPGF FOR WMSNs

Long Tran Huy[*], **Chinh Tran Thien**[*], **Hoai Trung Tran**[+]
[*] Posts and Telecommunications Institute of Technology, Vietnam
[+] University of Transport and Communications, HaNoi, VietNam

Abstract— Lightweight, secure routing protocol designed to ensure safety and security during routing on wireless networks with resource-constrained devices such as wireless sensor networks (WSNs) and IoT (Internet of Things). In this paper, we propose a geographically lightweight secure routing protocol (LS-TPGF) in a wireless multimedia sensor network (WMSN) using Cyclic Redundancy Check (CRC) and Elliptic Curve Cryptography (ECC) for node and message authentication. This protocol places great emphasis on verifying and determining the trusted origin of nodes and routing messages in the network. This ensures that only nodes and messages from trusted sources are accepted and participate in the routing process. By using lightweight authentication methods and efficient encryption algorithms, this routing protocol helps prevent spoofing attacks and ensures the security of information transmitted over the network. The algorithm's effectiveness has been confirmed through security analysis and simulation evaluation.

Keywords— CRC, Secu-TPGF, GSR, GSTP, MD5, SHA-3, MAC, Security, Routing, WMSN.

I. INTRODUCTION

Thanks to the features offered by new transmission technology, such as 4G and 5G, such as high speed and low latency, WSN continues to develop as one of the exciting and challenging research areas in the modern era. WMSN is a particular type of WSN using multimedia sensor nodes that can enhance the ability of WSN in the event description. However, the efficient transmission of multimedia streams in the WMSN is a challenge due to the sensor nodes' limited transmission bandwidth and energy resources. When designing a routing protocol, the following three requirements should be considered [1]: Multipath transmission, Hole Pass, and Shortest Path. Among them, Two-Phase geographic Greedy Forwarding (TPGF) [2] is one of the routing protocols first designed for WMSN, and it uses greedy geo-forwarding to discover one or more transmission paths through the WMSN holes are optimized for each node in the WMSN. Like most network protocols, TPGF is not designed to resist various attacks, so it is very vulnerable to external attacks such as data replay, identity theft, or internal attacks from the inside by extracting key and security information from the compromised node and then acting as a network protocol. These routing attacks can disrupt the entire network operation.

For a routing protocol, node authentication and routing messages are the important factors determining whether the designed routing protocol is secure. It plays an important role in ensuring the integrity and security of the network. They help prevent spoofing attacks and ensure that information is forwarded to the correct destination and on the optimal path. Node authentication is verifying the identity of a node (or device) in the network. When a node wants to join the network, authentication is performed to ensure the node is valid and has access to the network. At the same time, this also ensures that the message transmitted from the node is the owner, thereby verifying and protecting the integrity of information and transactions. It creates trust and accountability in validating and validating that node's activities and behavior in the network environment.

A routing message is a packet containing routing information in a network. When a node wants to transmit information to another node in the network, it uses a routing message to specify the optimal path and forward the information to the destination. Routing messages typically contain information about the destination address, source address, path evaluation parameters (such as latency or bandwidth), and other information needed to decide the best route.

Recent studies aimed at node authentication and routing messages have been proposed based on the routing mechanism of the original TPGF protocol, such as the SecuTPGF protocol proposed in [3] that used the message authentication code (MAC) to authenticate the origin and protect the information may change in the routing message. However, this incurs high computational costs. Or by using the MD5 hash in GSTP [4] and SHA-3 in the GSR [5] protocol to provide both node and message authentication, allowing it to secure the identity of node 1-hop and route through that 1-hop node. A comparison of MD5 and SHA-3 [6] with different parameters such as cost, message length, speed, and attacks described SHA-3 as more secure than MD5. MD5 is faster than SHA-3, but thanks to

Contact author: Long Tran Huy
Email: longth@ptit.edu.vn
Manuscript received: 6/2023, revised: 7/2023, accepted: 8/2023.

reduced circuitry, SHA-3 can work better on small devices like sensors with low computing power.

It can be said that MAC is considered more secure than MD5 because MD5 has discovered security holes [7]. Attacks using collisions on MD5 hashes have allowed hackers to generate two messages with the same hash value, which could lead to being fooled in checking the authenticity of the data. Meanwhile, MAC does not have the same security holes and is considered more secure. However, MAC can be slower than MD5 because it uses a secret key to generate the authentication code, while MD5 uses only a data hashing algorithm. MAC and SHA-3 are encryption tools used to protect data integrity. However, they have different purposes and applications. MAC is used to confirm the authenticity of data and ensure that it has not been modified in transit. The MAC is usually generated using a hash function such as SHA-3 and a secret key. MAC is a good choice for data validation and assurance that it has not been modified. However, if you just need to check the data's integrity and ensure it has not been changed, SHA-3 might be a better choice, as it provides a unique hash value for each data set.

Sensor nodes are devices with limited resources, such as limited battery capacity, low computing power, small storage capacity, and difficulty applying high-performance software and algorithms. Therefore, a secure and efficient routing protocol must be designed to prolong the network's life while preventing as many attacks as possible. Accordingly, lightweight cryptography studies aim to create compact implementation solutions without sacrificing security. It is a solution that offers a compromise between security and efficiency in the implementation of cryptographic algorithms. Therefore, more in-depth research is needed to keep up with and match the rapidly growing needs of WMSN applications.

In particular, CRC is a method of error checking widely used in most communication and data storage protocols. It has simple computation and fast processing speed, usually taking only a few microseconds to calculate the CRC value of a data block. The CRC uses polynomial division to generate an error check code, and if this does not match the error check value recalculated when the data is received, the data has failed in transit. Generally, CRC works faster than MAC because calculating MAC value can be more resource-intensive and takes more time than calculating CRC value. However, CRC can only detect random errors and cannot detect data tampering or MITM (man-in-the-middle) attacks. Therefore, if only CRC is used to verify the integrity of the data, the data can still be tampered with or stolen. To solve this problem, ECC is a lightweight encryption algorithm that encrypts and decrypts data. It uses a key pair (public key and private key) to encrypt and decrypt data and is considered one of the most efficient and secure encryption methods.

Therefore, the paper proposes a modified version of the TPGF protocol named LS-TPGF that uses both methods: using CRC for authentication and encrypting ECC to ensure message integrity and confidentiality.

The rest of this paper is organized as follows: Part II presents the work related to CRC and ECC. Part III presents lightweight, geographically secure WMSN routing utilizing CRC and ECC for node and message authentication. Part IV Simulation and evaluation. Part V concludes.

II. CRC AND ECC ALGORITHMS

A. Cryptographically Secure CRC

The CRC, devised by W. Wesley Peterson in 1961, originally identified inadvertent alterations in data transmitted across communication channels and played a role in energy conservation [8]. In contemporary contexts, CRC has found extensive applications within communication protocols, including Ethernet, Wi-Fi, Bluetooth, and numerous others.

Presently, diverse methodologies have emerged to enhance the security and efficiency of CRC, notably the utilization of products derived from smaller irreducible polynomials, which offer enhanced computational feasibility compared to singular polynomial approaches in generating CRC. As the researchers [9] outlined, a strategy involving reduced polynomials for hash function construction proves highly suitable for brief messages. The suggested approach advocates the employment of a sum of degree n, formed by the multiplication of k irreducible polynomials, as opposed to a solitary irreducible polynomial of degree n. As the quantity of irreducible polynomials of order n exhibits exponential growth concerning n, the computation of smaller irreducible polynomials is significantly streamlined. Determining irreducible polynomials can be achieved either through random polynomial selection accompanied by an irreducibility test (entailing a time complexity of $\Omega(n^3)$ bit operations [10]), or by maintaining a repository of such polynomials. The most popular CRC size is n = 32 and the number of irreducible polynomials of degree 32 is $134.215.680 \approx 2^{27}$. In contract, the number of irreducible polynomials of degree 16 is only 4080. Therefore, for many applications, a database of irreducible polynomials of degree 16 is acceptable while a database of irreducible polynomials of degree 32 is too large.

Moreover, as per Elena Dubrova's investigation [11], most link layers employ CRC exclusively to counteract inadvertent alterations during transmission. Safeguarding data integrity necessitates the incorporation of n-bits token authentication codes, such as HMAC keyed hash message authentication, KECCAK KMAC message authentication, or authentication token CBC-MAC cryptographic blockchain message. However, this approach enlarges the message by n bits and mandates a distinct encryption/decryption mechanism that is comparably more intricate than the CRC-based encoding/decoding process. For instance, research [12] demonstrated that KMAC128 engenders a storage requirement 45 times greater and consumes 28 times more power than a 128-bit CRC-based MAC algorithm [13].

The principal merit of cryptographically secure CRCs lies in their capacity to ensure robust data integrity, thereby enabling the identification of intentional and inadvertent data modifications during transmission. Cryptographically secure CRCs are meticulously structured to satisfy the

requisites of cryptographic hash functions, necessitating specific attributes such as collision resistance, resistance to first pre-image attacks (the prevention of deriving a new message with an identical hash value as a previously known one), and resistance to second pre-image attacks (the prevention of the capability to uncover an alternative message with an identical hash value as a previously known one).

Furthermore, the utility of cryptographically secure CRCs derives from their rapid computability, rendering them well-suited for systems demanding swift processing. Additionally, their efficacy is accentuated by their modest memory and computational demands, differentiating them favorably from certain alternative hash functions.

Beyond this, cryptographically secure CRCs find application across diverse domains, encompassing digital signature schemes, message authentication codes, secure communication protocols, as well as error detection and correction frameworks. Their exceptional precision in error detection bolsters their effectiveness as a tool.

The principal advantage inherent in cryptographically secure CRCs resides in their dual capability to uphold robust data integrity and facilitate efficient, concurrent error detection. This unique combination positions them as a compelling choice across applications where data security and operational efficiency are of paramount concern.

B. Security based on Elliptic curve cryptography (ECC)

ECC is a proficient and secure encryption technique, boasting noteworthy advantages over conventional methods such as RSA or DSA. ECC excels in key storage efficiency, yielding smaller key sizes than RSA or DSA. Its notable attributes encompass rapid encoding and decoding velocity, heightened security measures, and minimal power consumption. ECC's versatility is evidenced by its utility across multiple security applications, including encryption, digital signatures, and key exchange protocols.

Numerous research has harnessed the potential of ECC, offering implementations across a spectrum of public key cryptography tasks such as authentication, digital signatures, key agreement, and encryption. Dr. S. Vasundhara [14] has conducted an exhaustive evaluation of prevailing studies, conclusively demonstrating that using elliptic curves in cryptography surpasses the security and efficiency of alternative encryption techniques. Victor S. Miller introduced an encryption approach akin to the Diffie-Hellman key exchange protocol, boasting approximately 20% enhanced efficiency. Neal Koblitz demonstrated that establishing ECCs rooted in discrete logarithms presents greater resilience against breaches within finite groups than binary fields. This entails smaller block sizes, elevated speed, and augmented security. S. Maria Celestin and K. Muneeswaran achieved text encryption using ECC by converting messages into ASCII values and then mapping them to affine points on the elliptic curve through point addition involving the product of the ASCII value and the generator value. Sarvana, Suneetha, and Chandrasekhar devised a secure method for communicating with multiple parties through ECC authentication, incorporating supplementary parameters. Jarvinen. K and Skytta. J discussed the parallelization of ECC, curtailing point multiplication delay via a multiple-field multiplication technique that parallels Koblitz curves. Amara M. and Siad A elucidated the role of network security using ECC, highlighting its superiority over RSA and concluding ECC's superior suitability for encryption. Correspondingly, Vasundhara [14] introduced a novel approach for text encoding using ECC, segmenting ASCII values into groups determined by the ECC parameter's 'p-value, with radix exceeding the maximum ASCII value in the script. These groups are translated into large integers, paired, and employed as 'Pm' in ECC operations. This method obviates character-to-elliptic curve coordinate mapping and the need for shared lookup tables. Empirical results demonstrated swift encryption and decryption even with extensive input word counts, yielding more compact ciphertext than alternative methods. This, in turn, conserves bandwidth during transmission and eliminates customary mapping and lookup table requirements. ECC's smaller key sizes confer heightened security, surpassing the well-established RSA. The formidable challenge of solving the discrete logarithm problem in elliptic curves lends further robustness to ECC. The congruence of ECC's security with other cryptosystems and its diminutive key size renders it ideal for resource-constrained devices encompassing restricted power, storage, and processing capacities.

The algorithm can be summarized as follows:

Consider an elliptic curve expressed as $y^2 = x^3 + ax + b \bmod p$, where $0 \le x < p$. Constants a and b are non-negative integers less than prime number p, adhering to the condition:

$$4a^3 + 27b^2 \bmod p \neq 0 \quad (1)$$

Assume nodes A and B belong to the curve as mentioned above and introduce a generator G. The private keys of A and B are nA and nB, respectively. Their public keys are derived as follows:

$$Pa = nAG \text{ and } Pb = nBG \quad (2)$$

Should A aim to send message Pm to B, A will utilize B's public key for encryption, resulting in the ciphertext of the form:

$$Pc = \{kG, Pm + kPb\} \quad (3)$$

Where k represents a random integer, ensuring distinct ciphertext generation. Node B deciphers by subtracting nBkG coordinates from Pm + kPb.

$$Pm = \{Pm + kPb - nBkG\} \quad (4)$$

III. THE PROPOSED LS-TPGF PROTOCOL

A. Network Model and Assumptions

In the considered Wireless Multimedia Sensor Network (WMSN), the positions of sensor and source nodes are predetermined using GPS technology. Each sensor node possesses a transmission radius (TR) and connects with M neighboring sensor nodes within one hop. The sink node stands as a reliable entity with ample resources. Each node is assumed to withstand a particular duration before being compromised. Given the vulnerability of sensor nodes in WSNs, the absence of trust in these nodes is a prevalent

assumption, as adversaries can easily compromise them. Consequently, the proposed solution aims to enhance network efficiency in the presence of attacks.

The conceptualized WMSN can be depicted as a graph denoted by G(V, E), where $V=\{v_1, v_2,\ldots,v_n\}$ constitutes a finite collection of sensor nodes (vertices), and $E = \{e_1, e_2,\ldots,e_n\}$ represents a finite array of links (edges). $V_{source}=\{v_{s1}, v_{s2},\ldots,v_{sn}\}$ is a finite assemblage of randomly positioned source nodes in the network. Each sensor node manifests three states: available, unavailable, and dead. The link states can be either available or unavailable. Dead nodes, termed "holes," emerge when nodes exhaust their power or become overloaded due to occupied transmission lines, primarily about multimedia data transmission. These holes are categorized as follows: $V_{Static\_Hole}= \{v_{SH1}, v_{SH2},\ldots,v_{SHn}\}$, representing static hole nodes due to geographical conditions (e.g., rivers, swimming pools) or energy depletion. A path set of overloaded sensor nodes is denoted as $P_{nth} = \{v_{Pn1}, \ldots, v_{Pnm}\}$. This collection of paths culminates in a set of dynamic hole nodes, $V_{Dynatic\_Hole}= \{v_{DH1}, v_{DH2},\ldots,v_{DHn}\} = P_{1th} + \ldots + P_{nth}$, which revert to normalcy upon resolution of the respective path disruption. Correspondingly, links in the unavailable state are encompassed within $E_{hole} =\{e_{H1}, e_{H2},\ldots,e_{Hn}\}$. The array of malicious nodes is symbolized by $V_{Malicious}= \{v_{M1}, v_{M2},\ldots,v_{Mn}\}$. Thus, the secure nodes and corresponding links are represented by $V_{available} = V - V_{Dynamic\_Hole} - V_{Static\_Hole} - V_{Malicious}$ and $E_{available} = E - E_{Hole}$.

Mirroring the TPGF protocol [2], the inbuilt protocol addresses two main challenges. Initially, it identifies the complete set of secure paths, $P_{nth} = \{v_{Pn1}, \ldots, v_{Pnm}\}$, within the graph $G_{available}(V_{available}, E_{available})$ by disregarding hole and attack nodes. Subsequently, the optimization process aims to identify the path with the least number of nodes ($N_{optimized}$) from among the Pnth_optimized paths found, where $P_{nth\_optimized} = \{v_{OPn1}, \ldots, v_{OPnm}\}$ và ($P_{nth\_optimized} \subseteq P_{nth}$).

B. Attack Model

Within and beyond the network, adversary nodes interfere with the routing protocol. Consider nodes A and B that require communication via the wireless medium. The malicious nodes E and M, both inside and outside the network, launch various attacks such as spoofing, Sybil, Wormhole attacks, Flooding, and Selective Forwarding, as depicted in Figure 1. For instance, Node E, eavesdropping on private information from nodes A and B, might subsequently execute impersonation or substitution attacks, undermining message integrity.

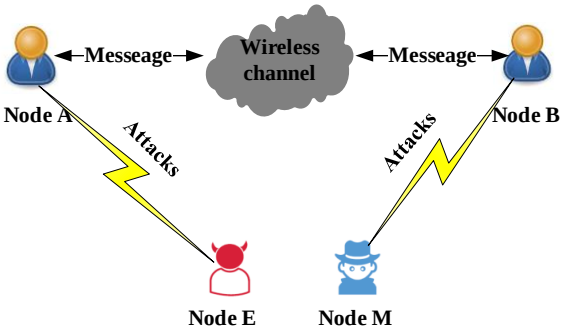

Figure 1. Attack model

C. Solution Model for Message Encryption and Authentication (CRC+ECC)

The proposed solution encompasses three phases: (i) network setup, (ii) identification of secure 1-hop nodes, and (iii) secure communication through these 1-hop nodes.

1. Network Setup

The WSN manager, authorized for authentication (base station), initiates network deployment and the initialization process using its infrastructure to minimize power consumption in other nodes. After the sensor network is deployed, the base station processes each sensor node's identity (ID) and computes the CRC hash of their IDs, storing the resulting hash as an attribute in the node (Figure 2).

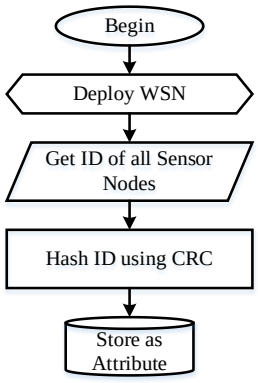

Figure 2. Flow graph of network setup

To implement CRC encryption, the message polynomial, M(x), is multiplied by x^n (where n is the degree of the generating polynomial, p(x)). The result undergoes modulo division by p(x), yielding the CRC check bits, r(x).

$$r(x) = M(x).x^n \; mod \; p(x) \quad (5)$$

These check bits are combined with the message, forming the CRC codeword:

$$M(x).x^n \oplus r(x) \quad (6)$$

Here, "⊕" signifies the XOR operation. Decoding involves dividing the received message modulo p(x) and comparing the remainder coefficients with the received CRC check bits. A mismatch indicates an error. The steps are as follows:

- Convert each node's ID into binary bits (e.g., 110101101).
- Employ the CRC-16-CCITT hash with the generating polynomial $p(x) = x^{16} + x^{12} + x^5 + 1$, represented as 10001000000100001.
- The generating polynomial comprises 17 bits.
- Append a 16-zero string to the transmitted bit stream, resulting in 11010110100000000000000000.
- Perform binary division, appending the division remainder to the message: 11010110101010111101110110.
- Authentication is achieved by dividing the received message by p(x). A result of 0 signifies an unchanged message, while a non-zero result indicates an incorrect received message.

Following the deployment phase, the source node in the network initiates the subsequent step of identifying secure 1-hop nodes.

2. Discovering Secure 1-Hop Nodes

This phase precludes rival nodes from participating in the WSN, exclusively allowing validating nodes to join in the initial stage. The CRC algorithm serves authentication purposes, while ECC encrypts outgoing messages (Figure 3). The process is outlined as follows:

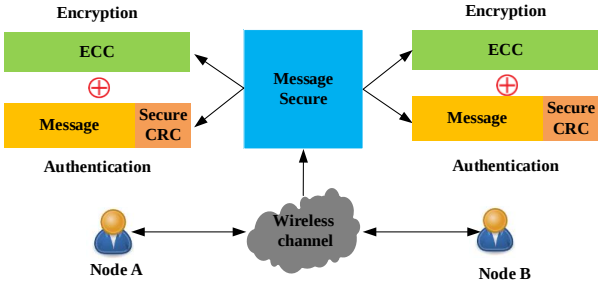

Figure 3. Proposed model

Each node broadcasts a message to discover its 1-hop nodes upon sensor node deployment. This message includes the node's Identity (ID) + CRC code and Location and Geographic Location (GL) information. Prior to transmission, the message is encrypted with ECC (as depicted in Figure 3). The node then awaits responses from neighboring nodes.

The CRC code is appended to the node's ID for authentication purposes. Subsequently, the entire routing message undergoes ECC encryption:

a. Encoding

Step 1: Select an elliptic curve using a 128-bit key and origin G.

Step 2: Retrieve the message for transmission.

Step 3: Convert the entire message into binary code.

Step 4: Choose a positive integer k (the signer's secret key) with $1 \le k \le n-1$. Using point multiplication, compute the point k * G (where G is the chosen origin) and kPb.

Step 5: Calculate Pm + kPb using point addition or doubling as required.

Step 6: Send the encrypted message Pc = {kG, Pm + kPb} to the recipient.

Each execution of the program generates distinct ciphertext, even for the same initial message, due to the randomness of k in the operations.

b. Decoding

The decryption process is as follows:

Step 1: Obtain the ciphertext.

Step 2: Separate the left part (kG) and the right part (Pm + kPb) of Pc.

Step 3: Multiply the left part by nB and subtract it from the right part to obtain Pm:

$$\{Pm + kPb\} - nBkG = Pm$$

Considering that Pb = nBG, the subtraction can be transformed into an addition by multiplying the y-coordinate by -1. This operation can be demonstrated through point addition.

c. Algorithm for Discovering Secure 1-Hop Nodes

The procedure for discovering secure 1-hop nodes unfolds as follows:

Step 1. Each node broadcasts a message to the network. For instance, Node A sends a message structured as follows:

$$a \to * : ECC\big(HELLO\,(ID_A + CRC_A, GL_A)\big) \qquad (7)$$

Where ID_A represents the Identity of node A, augmented with the CRC check code (as elaborated in section III.A), and GL_A signifies the location of node A.

Step 2. Neighbor node B decrypts the ECC message and verifies if node A's ID is in its storage directory. This verification entails dividing the received ID_A+CRC_A message by p(x) (explained in the preceding section). If the result is zero, it indicates a match with the value in the archive. Node B proceeds to transmit a message containing its ID and location to node A:

$$B \to A : ECC\big((ID_B + CRC_B, GL_B)\big) \qquad (8)$$

Step 3. Upon receiving this message, Node A acknowledges and stores it as a 1-hop neighbor. Every node in the network repeats this process to verify secure 1-hop neighbors, establish secure links, and augment their list of secured 1-hop neighbors.

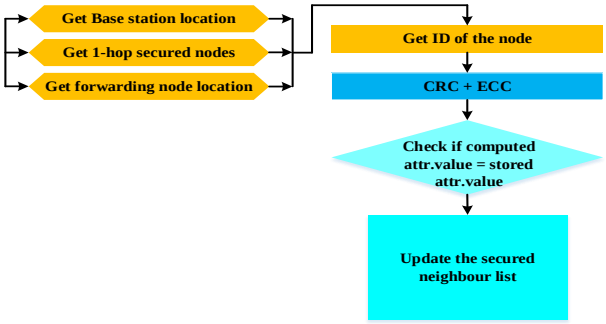

Figure 4. Flow graph of discovering secured 1-hop nodes

This iterative verification ensures that each network node recognizes secure 1-hop nodes, establishes secure links, and adds them to their list of secured 1-hop neighbors, as depicted in the corresponding figure.

3. Transmission through Secure 1-Hop Nodes

The source node initializes the routing process, which dispatches a request to the nearest secure 1-hop node among the identified secure 1-hop neighbors or the base station. Upon receiving the request, the forwarding node confirms if it possesses a secure 1-hop node for transmission. If such a node exists, it forwards the request to the subsequent relay node or the base station. In cases where multiple single-hop secure nodes are identified, the one closest to the base station is selected. In a 'blocking' situation, the node reverts to the previously secured 1-hop node, marks itself, and disregards its involvement.

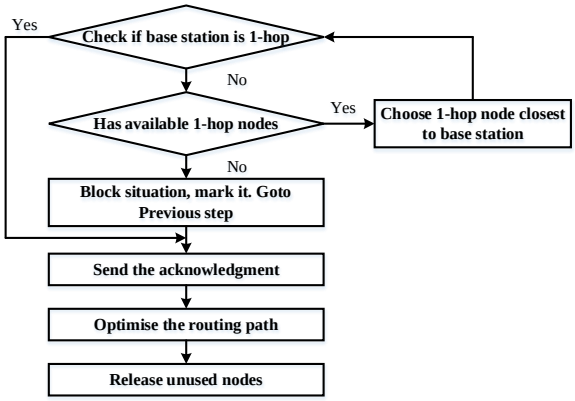

Figure 5. Flow graph of secured forwarding and transmission

Iterative steps of marking and returning, as illustrated in Figure 5, determine the subsequent secured 1-hop node for greedy forwarding. Each designated 1-hop secure node is assigned a numerical label associated with the path number.

An acknowledgment from the base station signifies the establishment of a routing path. This acknowledgment is relayed over 1-hop secure nodes with the same maximum node count and path number. Optimization is undertaken at each intermediate node during the backpropagation process along the designated path to eliminate path loops. Upon confirmation, the source node commences multimedia data transmission, concurrently executing a release instruction for all other 1-hop nodes not participating in the transmission.

IV. SIMULATION AND EVALUATION

To evaluate and analyze the proposed LS-TPGF protocol, the Nettopo emulator, specifically designed for the TPGF protocol, is employed [15], [16]. LS-TPGF, which builds upon the TPGF routing protocol, leverages CRC and ECC algorithms for security enhancement. Its performance is benchmarked against the previous SecuTPGF protocol, which employed user-defined security algorithms. Evaluation metrics encompass network characteristics like the number of routing paths and the average path length.

A. Performance Evaluation Parameters and Considerations

In evaluating network performance, while the lifetime parameter remains a primary concern in conventional Wireless Sensor Networks (WSNs), Wireless Multimedia Sensor Networks (WMSNs) often prioritize parameters like end-to-end delay and path length when assessing routing algorithms [3]. These parameters are elucidated as follows:

- End-to-End Delay (D_e2e): This signifies the time taken to transmit information from the source node to the sink node. The average latency for each hop is represented as D_{hope} + $D_{otherfactors}$. Mathematically, it can be expressed as:

$$D_{e2e} = k \times \left(D_{hope} + D_{otherfactors}\right) \quad (9)$$

Where k is the number of hops, D_{hop} denotes transmission delay, and $D_{otherfactor}$ accounts for delays related to other factors.

The average delay (Dhop + Dotherfactors) remains constant for each hop. Consequently, we observe that:

$$D_{e2e} \alpha k \quad (10)$$

Equation (10) establishes that terminal delay is proportional to the number of hops (k). As the number of hops decreases, the end-to-end latency reduces, implying a shorter time for information transmission.

- Path Length (P_Length): Path length is determined by summing up the weights attributed to each traversed link. Some routing protocols employ hop count to gauge the number of intermediate nodes that a packet must traverse between source and sink nodes:

$$P_{Length} = k(\text{ hop}) \quad (11)$$

Where k represents the number of hops.

- Routing Latency or End-to-End Latency: This term refers to the time needed to transmit information from the source node to the sink node. It can be calculated from equation (11).

In the simulation, the network size was maintained at 640 × 400. The average number of hops and paths were computed by varying the number of nodes (ranging from 100 to 1000) to achieve diverse outcomes. Simulation parameters are detailed in the table below:

Table 1. Simulation parameters

Parameter	Value
Network size	640 x 400 m
Number of sensor nodes	100 - 1000
Number of base station	1
Number of source nodes	1
Initial Energy of sensor nodes	10 J
Transmission radius	60 - 120 m
Expected lifetime	1 - 14 h

- Source Node: (ID: 2; Energy: 10J; Location: 33.94; Max TR: 60; Bandwidth: 1; Expected Lifetime: 1)

- Sink Node: (ID: 1; Location: 585,349; Max TR: 60; Bandwidth: 1)

- Sensor Nodes (Purple) and Attack Nodes (25% of sensor nodes) were randomly distributed across the network (Figure 6).

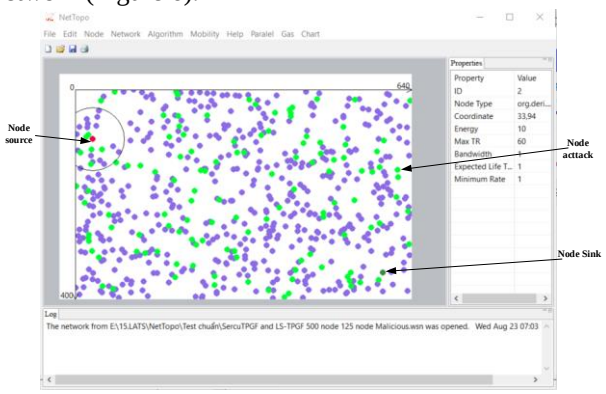

Figure 6. Network setup

- SecuTPGF and LS-TPGF algorithms were sequentially executed and compared.

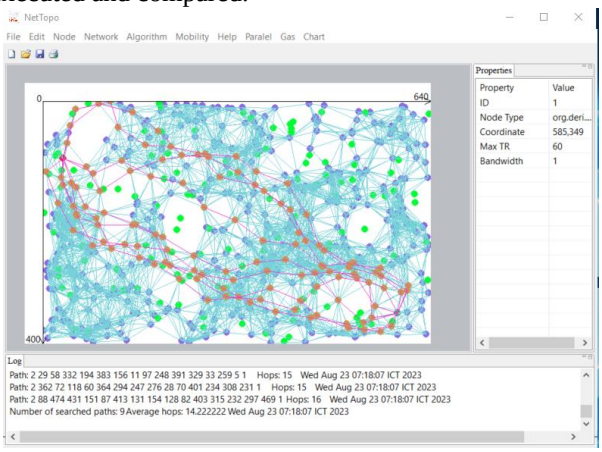

Figure 7. Simulation run results

The NetTopo implementation of LS-TPGF is shown in

Figure 7. Malicious nodes are excluded from transmission lines. Upon receiving routing requests, intermediate nodes ascertain if the base station is reachable within one hop. If affirmative, they establish the route and send an acknowledgment. In case of being an intermediate 1-hop node, they simply forward the request to the next secure 1-hop node. This process continues until the base station is reached.

B. Evaluation and Comparison

Table 2 compares simulation results for the average number of hops before and after optimization in route discovery using SecuTPGF and LS-TPGF algorithms.

Table 2. Average number of hops

Number node	Before optimization		After optimization	
	SecuTPGF	LS-TPGF	SecuTPGF	LS-TPGF
100	0	18	0	15
200	23	21	18	16
300	24	22	17	15
400	22	20	17	15
500	20	18	16	14
600	19	17	16	14
700	18	16	16	14
800	18	16	16	14
900	20	18	15	13
1000	19	19	14	12

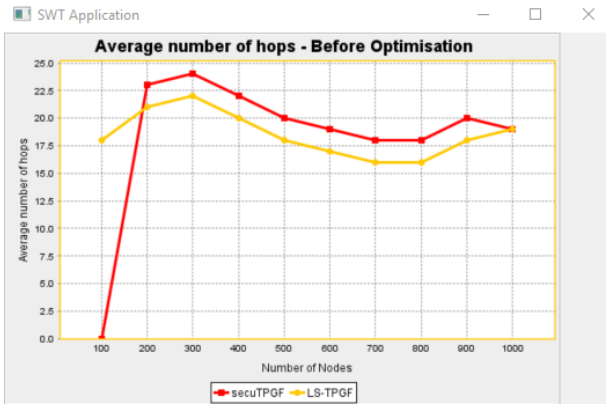

Figure 8. Average number of hops - before optimization (compare SecureTPGF and LSTPGF)

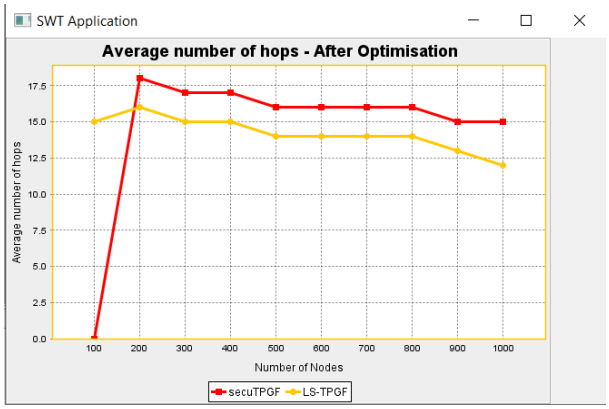

Figure 9. Average number of hops - after optimization (compare SecureTPGF and LSTPGF)

A similar setup to GSTP and GSR protocols was employed, yielding results in Figures 10 and 11.

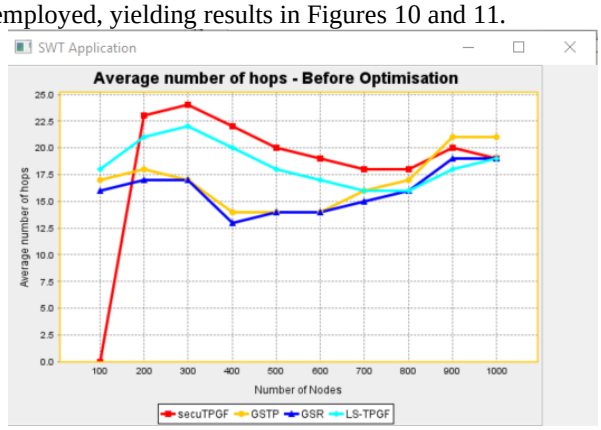

Figure 10. Average number of hops-before optimization (compare SecureTPGF, GSTP, GSR and LSTPGF)

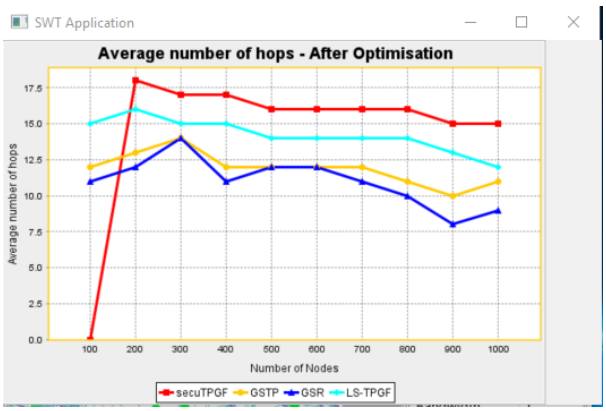

Figure 11. Average number of hops - after optimization (compare SecureTPGF, GSTP, GSR and LSTPGF)

Simulation outcomes indicate that the average number of hops achieved by the LS-TPGF protocol tends to be lower than that of SecuTPGF, attributable to implementing lightweight algorithms while ensuring attack prevention capabilities. However, due to the use of distinct algorithms, the LS-TPGF protocol exhibits slower performance than GSTP and GSR.

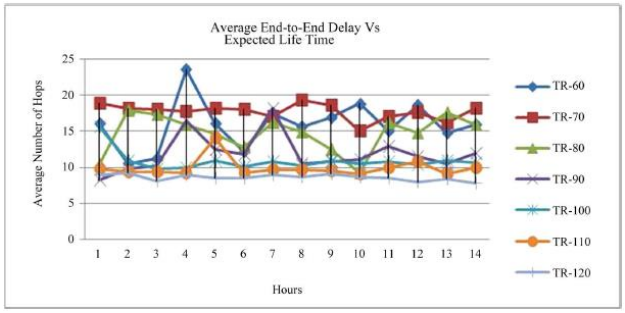

Figure 12. Average end-to-end delay vs expected life time

An observation is made when extending the transmission distance of sensor nodes from 60 to 120; the average number of hops proportionally diminishes (Figure 12).

This article's proposed lightweight encryption and authentication solution demonstrates robustness in safeguarding data against external threats. Specifically:

- CRC Method: CRC functions by appending a series of

check bits at the end of data to create a new data block. This fresh data block is transmitted via the communication channel. Upon reception, the CRC algorithm recalculates the check bit sequence's value and compares it with the transmitted value. Inconsistencies indicate potential data corruption, prompting retransmission. This methodology ensures data integrity and minimizes transmission errors. Although CRC-16-CCITT, with a 16-bit block length, can detect most data errors, it's not foolproof. However, its computational simplicity and swift processing speed make it a favored choice for applications necessitating rapid data transfers. Hence, it's amalgamated with ECC as elucidated in Section III to bolster reliability.

- ECC-128: ECC-128 is remarkably resilient against modern attack techniques such as brute force, upper bound, and fake key attacks. A brute force attack on ECC-128 would require trying all feasible decryption keys, which amounts to a staggering 2^{128} possibilities, rendering it impractical within reasonable timeframes—particularly for routing messages. Consequently, ECC is a potent and secure encryption mechanism that protects sensitive information.

Detecting and thwarting network-internal attacks poses growing complexity and challenge. While LS-TPGF cannot eradicate such attacks, it does ameliorate their impact across the network:

- Wormhole Attack: LS-TPGF addresses the Wormhole attack by implementing strategies like evaluating delays and path lengths between nodes—done by scrutinizing the maximum distances of nodes (roughly their transmission radius). It also fuses ECC encryption with CRC authentication techniques to ensure communication integrity.

- Sybil Attack: In the LS-TPGF protocol, infiltrating nodes with incorrect IDs is impeded because nodes without authenticated control cannot gain entry. Consequently, the feasibility of a Sybil attack is thwarted.

- Node Replication Attack: Routine monitoring of the base station diminishes the repercussions of this attack. Since all TPGF-defined routing paths are node-separated, it's deemed replicating if a node concurrently exists in multiple paths. Such nodes are blacklisted, their IDs revoked, and expelled from the WSN.

- Selective Forwarding Attack: LS-TPGF tackles the Selective Forwarding attack by vigilantly observing the transmission behavior of subsequent neighbors. This countermeasure helps mitigate the impact of the attack.

The multifaceted approach integrated into LS-TPGF underscores its potency in enhancing the security and resilience of WMSN communication.

V. CONCLUSION AND FUTURE WORK

In conclusion, this paper introduces a novel approach similar to SecuTPGF, utilizing two distinct algorithms for node authentication and routing messages. Unlike conventional resource-intensive algorithms, the proposed solution adopts lightweight methods—CRC and ECC—for node authentication and routing message encryption, ensuring reliability while accommodating resource-constrained devices like WSNs.

In addition to ECC-based encryption, digital signatures could bolster message authentication. This involves digitally signing messages using the Elliptic Curve Digital Signature Algorithm (ECDSA), which verifies the message's origin and integrity. Considering the sink node as a trusted authority within the WSN, it initiates and gathers transmitted data, raising challenges related to security and storage due to the centralized server/client model. A distributed model, such as blockchain, can address this within the WSN system. As a decentralized technology, blockchain holds promise for enhancing computation, management processes, and security in WSNs.

Future research directions entail exploring digital signatures as an alternative to the current solution and advancing sink node security through integrating Blockchain technology. This trajectory aims to fortify the security landscape of WSNs further, accommodating evolving needs and challenges.

REFERENCES

[1] L. Shu, Y. Zhang, L. Yang, Y. Wang and M. Hauswirth, "Geographic Routing in Wireless Multimedia Sensor Networks," In Proceedings of Second International Conference on Future Generation Communication and Networking, FGCN '08, Hainan Island, 2008.

[2] Lei Shu, Yan Zhang, Laurence T. Yang, YuWang, Manfred Hauswirth, Naixue Xiong, *TPGF: geographic routing in wireless multimedia sensor networks*, Telecommun Syst (2010) 44: 79–95.

[3] Taye Mulugeta1, Lei Shu, Manfred Hauswirth, Min Chen, Takahiro Hara, Shojiro Nishio, *Secured Two Phase Geographic Forwarding Protocol in Wireless Multimedia Sensor Networks*, 2010 IEEE Global Telecommunications Conference GLOBECOM 2010.

[4] B. Prathusha Laxmi, A. Chilambuchelvan, *GSTP: Geographic Secured Two Phase Routing Using MD5 Algorithm*, Circuits and Systems, 2016, 7, 1845-1855, Published Online June 2016 in SciRes.

[5] B. Prathusha Laxmi, A. Chilambuchelvan, GSR: Geographic Secured Routing using SHA-3 algorithm for node and message authentication in wireless sensor networks, Future Generation Computer Systems 76 (2017), pp 98–105.

[6] Piyush Gupta, S. K., 2014. *A comparative analysis of SHA and MD5 algorithm*, 5(Int. J. Comput. Sci. Inf. Technol), p. 4492–4495.

[7] Dan Kaminsky, *MD5 To Be Considered Harmful Someday*, Senior Security Consultant, Avaya 2004.

[8] Raghini Sharma, Dr. Umarani Chellapandy, *A Survey on Encrypted and Decrypted Text Algorithm Using CRC, SHA-256, MD5 and Caesar Cipher*, International Journal of Trend in Scientific Research and Development (IJTSRD) Volume 6 Issue 2, January-February 2022

[9] Elena Dubrova, Mats Naslund, Goran Selander, Fredrik Lindqvist, *Cryptographically Secure CRC for Lightweight Message Authentication*, Computer Science, Mathematics IACR Cryptol. ePrint Arch 2015.

[10] S. Gao and D. Panario, "*Tests and constructions of irreducible polynomials over finite fields*" in Foundations of Computational Mathematics (F. Cucker and M. Shub, eds.), pp. 346-361, Springer Berlin Heidelberg, 1997.

[11] Elena Dubrova, Mats Näslund, Goran Selander, Fredrik Lindqvist, *Lightweight Message Authentication for Constrained Devices*, WiSec '18: Proceedings of the 11th

ACM Conference on Security & Privacy in Wireless and Mobile Networks, 2018.
[12] Yang Yu. 2017. *Evaluation of Cryptographic CRC in 65nm CMOS*. M. Sc. Thesis, Royal Institute of Technology (KTH), Sweden.
[13] Elena Dubrova, Mats Naslund, Goran Selander, and Fredrik Lindqvist. 2018. *Message Authentication Based on Cryptographically Secure CRC without Polynomial Irreducibility Test*, Cryptography and Communications 10 (March 2018), 383-399. Issue 2.
[14] Dr. S. Vasundhara, *The Advantages of Elliptic Curve Cryptography for Security*, Global Journal of Pure and Applied Mathematics, ISSN 0973-1768 Volume 13, Number 9 (2017), pp. 4995-5011.
[15] L. Shu, C. Wu and M. Hauswirth, "*NetTopo: Beyond Simulator and Visualizer for Wireless Sensor Networks*," Technical Report of Digital Enterprise Research Institute, 2008.
[16] L. shu, M. Hauswirth, H.-C. Chao, M. Chen and Y. Zhang, "*NetTopo: A framework of simulation and visualization for wireless sensor networks*" Adhoc Networks, vol. 9, p. 799–820, 2011.

ĐỊNH TUYẾN AN TOÀN NHẸ MỚI DỰA TRÊN GIAO THỨC TPGF CHO WMSNs

Tóm tắt: Giao thức định tuyến an toàn nhẹ được thiết kế đặc biệt để đảm bảo tính an toàn và bảo mật trong quá trình định tuyến trên mạng không dây với các thiết bị có tài nguyên hạn chế như mạng cảm biến không dây (WSN) và mạng IoT (Internet of Things). Trong bài báo này, chúng tôi đề xuất giao thức định tuyến an toàn nhẹ theo địa lý (LS-TPGF) trong mạng cảm biến không dây đa phương tiện (WMSN) bằng cách sử dụng thuật toán kiểm tra dự phòng theo chu kỳ (Cyclic Redundancy Check - CRC) và mật mã đường cong Eliptic (Elliptic Curve Cryptography - ECC) để xác thực nút và tin nhắn. Giao thức này đặt nặng vào việc xác minh và xác định nguồn gốc tin cậy của các nút và bản tin định tuyến trong mạng. Điều này đảm bảo rằng chỉ các nút và bản tin từ các nguồn đáng tin cậy mới được chấp nhận và tham gia vào quá trình định tuyến. Bằng cách sử dụng các phương pháp xác thực nhẹ và thuật toán mã hóa hiệu quả, giao thức định tuyến này giúp ngăn chặn các cuộc tấn công giả mạo và đảm bảo tính bảo mật của thông tin truyền qua mạng. Hiệu quả của thuật toán đã được xác nhận thông qua phân tích bảo mật và đánh giá mô phỏng.

***Từ khóa*:** LS-TPGF, CRC, Secu-TPGF, GSR, GSTP, MD5, SHA-3, MAC, mật mã, định tuyến, WMSN.

Long Tran Huy received an electronics and telecommunications engineer from Electric Power University, in 2013 and a Master's degree from Posts and Telecommunications Institute of Technology in 2015. Currently, he is a postgraduate of the University of Communications and Transport, Hanoi. He is a lecturer at the Faculty of Telecommunications 1 - Institute of Post and Telecommunications Technology. His current research interest is information security, Routing security, wireless communications, WSN, UAV, and IoT.

Email: longth@ptit.edu.vn

Chinh Tran Thien was born in 1967. He got a Bachelor's degree from the University of Transport and Communications (UTC) in 1991 and hold visiting lecturers at the University. He then got a Ph.D. specialist in "Networks and Communications" from the Posts and Telecommunications Institute of Technology in 2005. Currently, he is deputy director of the Research Institute of Posts and Telecommunications. Main research directions: Research and development of Internet of Things (IoT) applications in the fields of telecommunications, smart transportation, smart agriculture, intelligent environment management, smart health, etc. Research and development of security and security applications in information communication technology (ICT), which focus on wireless sensor networks (WSN). Research and develop processing, control, and automation systems. Which focus on smart city, smart transportation.

Email: trthchinh@gmail.com

Hoai Trung Tran was born in 1976. He is the Deputy Head of the Department of Telecommunications Engineering, Faculty of Electrical - Electronic Engineering - University of Transport and Communications. His research directions are advanced wireless communication: cooperative and cognitive communication, mm-wave communication; digital signal processing, design, and production of wireless transceivers: beamforming, multiantenna, hybrid precoder, space-time coding, and spatial filter, massive MIMO, F- OFDM,

FPGA, etc.; application of new technologies integrated by information, communication, and electronics such as WSN; telecommunication using artificial intelligence and deep learning.

Email: trungth@utc.edu.vn.