WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5887

Security In Wireless Sensor Networks Based On Lightweight Algorithms: An Effective Survey

Huda lafta majeed [1,*], Mohammad Hasan Abd [2], Sif .K. Ebis3 [3]

[1] College of Agriculture, Wasit University, Iraq
[2]University Baghdad .College Of Education Ibn al-Haytham, Department of Computer Science
[3]Wasit Education Directorate, Ministry of Education ,Iraq

*Corresponding Author: Huda lafta majeed

DOI: https://doi.org/10.31185/wjcm.106
Received: January 2023; Accepted: February 2023; Available online: March 2023

ABSTRACT: At both individual and organizational levels, Wireless Sensor Networks (WSNs) find extensive applications across various sectors. Sensors play a pivotal role in industries such as agriculture, transportation, healthcare, and more. The utilization of sensors is closely linked to various technologies like wireless communication protocols, the Internet of Things (IoT), cloud computing, mobile computing, and other emerging innovations. The integration of these technologies often involves the transmission of critical data, emphasizing the necessity to safeguard this information from potential threats. However, safeguarding communication in WSNs presents a notable performance challenge due to the constrained computation and power capabilities of WSN components. Given the limited resources, implementing security measures becomes crucial, and in this context, employing lightweight encryption techniques becomes imperative. Traditional public-key and secret encryption methods, which involve extensive calculations, impose a significant performance overhead in WSNs. Therefore, the need arises for encryption techniques that are efficient in this resource-constrained environment. Security holds paramount importance in various applications involving sensor networks. Traditional cryptography has led to the development of several security protocols tailored for wireless sensor networks. Among the symmetric-key encryption techniques applied in sensor network configurations are AES, RC5, SkipJack, and XXTEA. However, these algorithms exhibit certain vulnerabilities, including susceptibility to chosen-plaintext attacks, brute force attacks, and computational complexity

Keywords: Confidentiality in Data, Cryptography Integration in Data, Wireless Networks Security

1. INTRODUCTION

Wireless sensor networks (WSN) have garnered considerable attention in recent years, owing to significant advancements in compacting technology and the development of low-power circuit designs that prove to be highly efficient in transmitting sensitive information via wireless communication. These networks find applications in various fields, including home automation, military operations, and health monitoring. However, WSNs encounter several challenges, such as limited processing power, short battery life, restricted memory, and the constraints of wireless communication channels. Consequently, security emerges as a predominant concern in these networks. The widely recognized limitations of WSNs make it impractical to employ traditional encryption techniques. This study presents an overview of existing cryptographic frameworks, comparing them with recently developed alternatives [1-4].

The recognition of sensor networks within the research community has elevated their prominence. These networks consist of arrays comprising tens of thousands of sensor nodes capable of wireless communication, which self-organize. Due to limited resources at the nodes, complex algorithms are impractical. The primary challenge in promoting widespread use of this network lies in ensuring security. Cryptography plays a vital role in securing Wireless Sensor Networks (WSN), and various algorithms, encompassing symmetric, asymmetric, and hybrid approaches, have been proposed [5-9].

Consequently, security measures must be implemented across the entire network. Cryptography is a commonly employed technique to ensure network security [10-13]. However, cryptographic algorithms in WSN should be designed to be robust while minimizing additional memory, power, or energy usage, as the algorithm's lifespan may vary depending on the application.

2. REQUIREMENTS OF SECURITY AND LIGHTWEIGHT ALGORITHMS

Confidentiality ensures that messages remain concealed from potential attackers, safeguarding the privacy of all communications within the sensor network. The considerations for addressing confidentiality in a Wireless Sensor Network (WSN) encompass the following aspects: (i) Sensor nodes should restrict access to their readings, allowing only authorized neighbors; (ii) the key distribution mechanism must be highly reliable; and (iii) public information, such as sensor identities and node public keys, should be periodically encrypted to thwart traffic analysis attacks [14-21].

Authentication, by verifying the origin of a message, ensures its reliability. This involves requiring authentication from other nodes, cluster chiefs, and base stations before releasing data or allocating finite resources. The authentication challenges in a WSN include ensuring that the transmitting node is legitimate and that the receiving node can unmistakably confirm the origin of received packets [22, 23].

Integrity involves verifying that a message hasn't been tampered with during its transmission, ensuring the accuracy of the data. Considerations for maintaining integrity in a WSN include: (i) restricting access to keys to only network nodes, with designated base stations being the sole entities allowed to alter the keys, preventing unauthorized updates; and (ii) safeguarding against active attackers attempting to disguise their attacks as noise [24].

Availability ensures that network resources or those provided by individual sensor nodes remain accessible as needed. Factors to address in the context of availability in a WSN include: (i) implementing security mechanisms as a unified connection control system to ensure effective message delivery to the target node; and (ii) ensuring constant accessibility of security mechanisms to prevent a single point of failure [25].

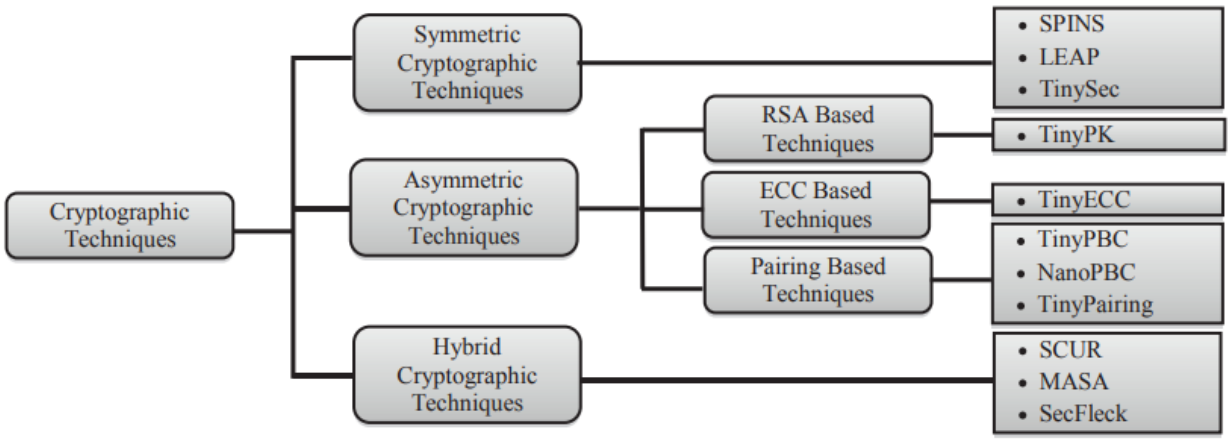

FIGURE 1 : Cryptography Approaches

In light of cryptography serving as a comprehensive solution for meeting security requirements, the meticulous selection of the most effective cryptographic technique is imperative. The cryptographic methods employed in Wireless Sensor Networks (WSNs) should not only cater to the demands of sensor nodes but also undergo evaluation based on criteria such as code size, data size, time consumption, and power consumption [26].

Consequently, it becomes necessary to either innovate new techniques or adapt existing approaches to meet the aforementioned stringent security criteria. The diagram in Figure1 show cryptography approaches. The current repertoire of cryptographic methods can be classified into three distinct categories: symmetric cryptographic methods, asymmetric cryptographic methods, and hybrid cryptographic methods. As for asymmetric cryptographic methods, they are further categorized into three types: RSA-based methods, ECC-based methods, and pairing-based methods.

3. SYMMETRIC CRYPTOGRAPHIC TECHNIQUES

Symmetric cryptographic algorithms involve the use of a single shared key for both encryption and decryption by the communicating nodes. Maintaining the secrecy of this key in the open network environment where WSNs operate can be challenging. Owing to its simplicity of implementation on constrained hardware and minimal energy requirements [27], symmetric cryptography predominantly constitutes the foundation of security algorithms for WSNs, particularly when implemented in hardware to mitigate performance degradation. Two distinct types of symmetric ciphers are employed: stream ciphers, which operate bitwise on the data, and block ciphers, which operate on blocks of a specific length. A stream cipher essentially functions as a block cipher with a block length of 1 bit.

4. ASYMMETRIC CRYPTOGRAPHIC TECHNIQUES

In asymmetric cryptography, a public key is utilized for data encryption and verification, while a private key is employed for decoding and data signing. While the public key can be openly disseminated, the private key necessitates

strict confidentiality. Asymmetric cryptography is alternatively referred to as public key cryptography [28]. Given that a majority of systems rely on extensive integer arithmetic, public key cryptography often proves to be resource-intensive. For years, many researchers dismissed public key cryptography as unfeasible with the constrained hardware prevalent in WSNs.

Utilizing public key algorithmic methodologies such as the Diffie-Hellman key agreement procedure or RSA signatures is discouraged within Wireless Sensor Networks (WSNs) due to concerns related to code size, data size, CPU utilization, and power consumption. Noteworthy examples of public key algorithms include Elliptic Curve Cryptography (ECC), Rabin's Scheme, Ntru-Encrypt, RSA, Pairing Based Cryptography (PBC), and Identity Based Encryption [29].

5. LIGHTWEIGHT CRYPTOGRAPHY

Lightweight Cryptography pertains to a collection of cryptographic symmetric and asymmetric algorithms expressly crafted to ensure effective security in systems constrained by specific requirements in running environments and power capabilities, as seen in Wireless Sensor Networks (WSNs). Achieving lightweight attributes in cryptography involves optimization at both the hardware and software levels. This optimization encompasses considerations such as chip size, energy consumption at the hardware level, code size, and RAM complexity at the software level, serving as benchmarks to assess the level of optimization achieved through the application of Lightweight Cryptography (LWC).

From a cryptographic standpoint, both hardware and software implementations undergo customization with the understanding that what may be deemed lightweight for a hardware implementation may not hold true for a software implementation, and vice versa. Traditional cryptographic schemes often incorporate large keys and multiple rounds in their design to bolster protection against malicious attacks. However, employing these conventional cryptographic schemes to secure WSN systems presents challenges, as they tend to consume substantial power during processing and necessitate extensive silicon real estate.

In contrast, WSNs require security measures that ensure adequacy while consuming less power, demanding fewer silicon resources, and showcasing high performance [30].

The diagram in Figure2 show lightweight cryptography. The Lightweight cryptography encompasses a set of cryptographic symmetric and asymmetric algorithms designed to ensure adequate security in systems with specific operational environments and power capabilities, such as Wireless Sensor Networks (WSNs). This term applies to both hardware and software implementations. The effectiveness of Lightweight Cryptography (LWC) is evaluated based on factors like chip size, energy consumption, and RAM complexity at the hardware level.

In traditional cryptographic methods, robust defenses against malicious attacks involve the use of large keys and multiple rounds. However, applying these conventional techniques to secure WSN systems presents challenges, as they necessitate significant silicon area and consume substantial power during processing. Therefore, to safeguard WSNs effectively, security strategies must be employed that utilize minimal silicon, consume less power, and deliver high performance. Cryptographers optimize both software and hardware implementations, recognizing that what may be considered lightweight in one aspect may not necessarily be lightweight in the other. The diagram in Figure 3 show performance and assorted aspects.

Designed with the specific intention of tackling the delicate balance between security and performance inherent in embedded systems and Cyber-Physical Systems, this investigation incorporates the examination of lightweight cryptographic techniques.

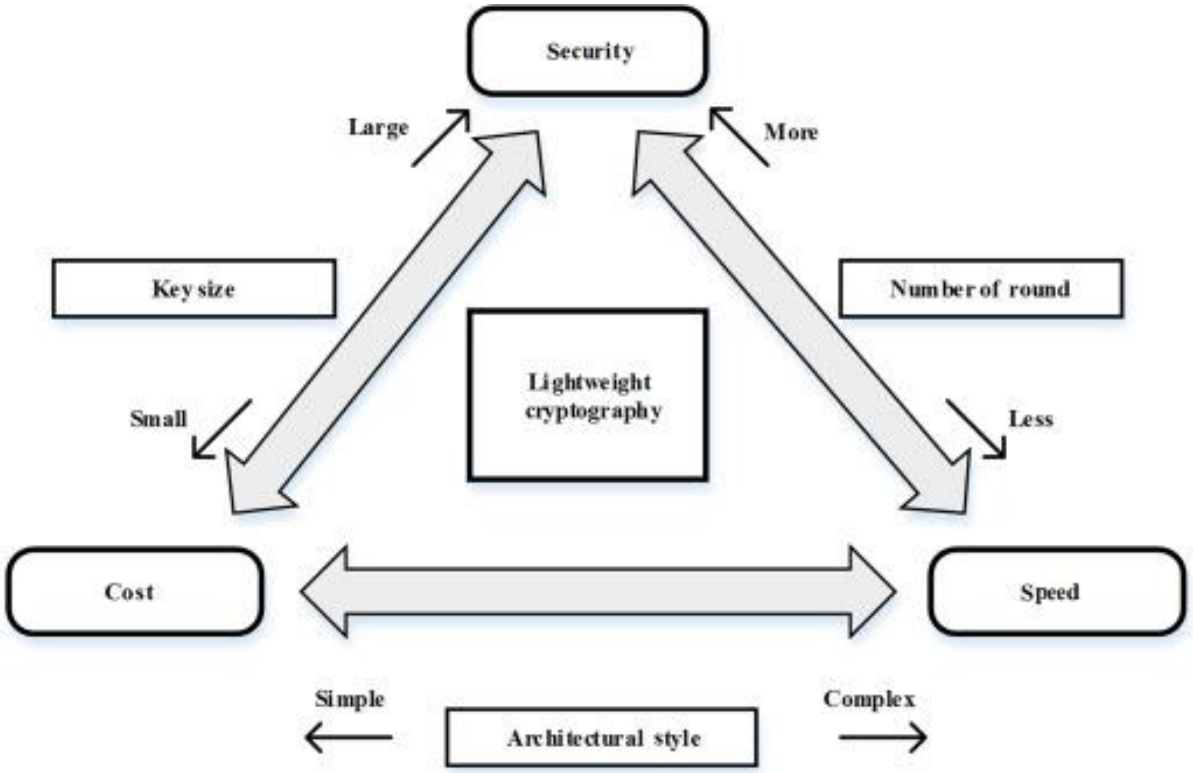

FIGURE 2 :Lightweight Cryptography

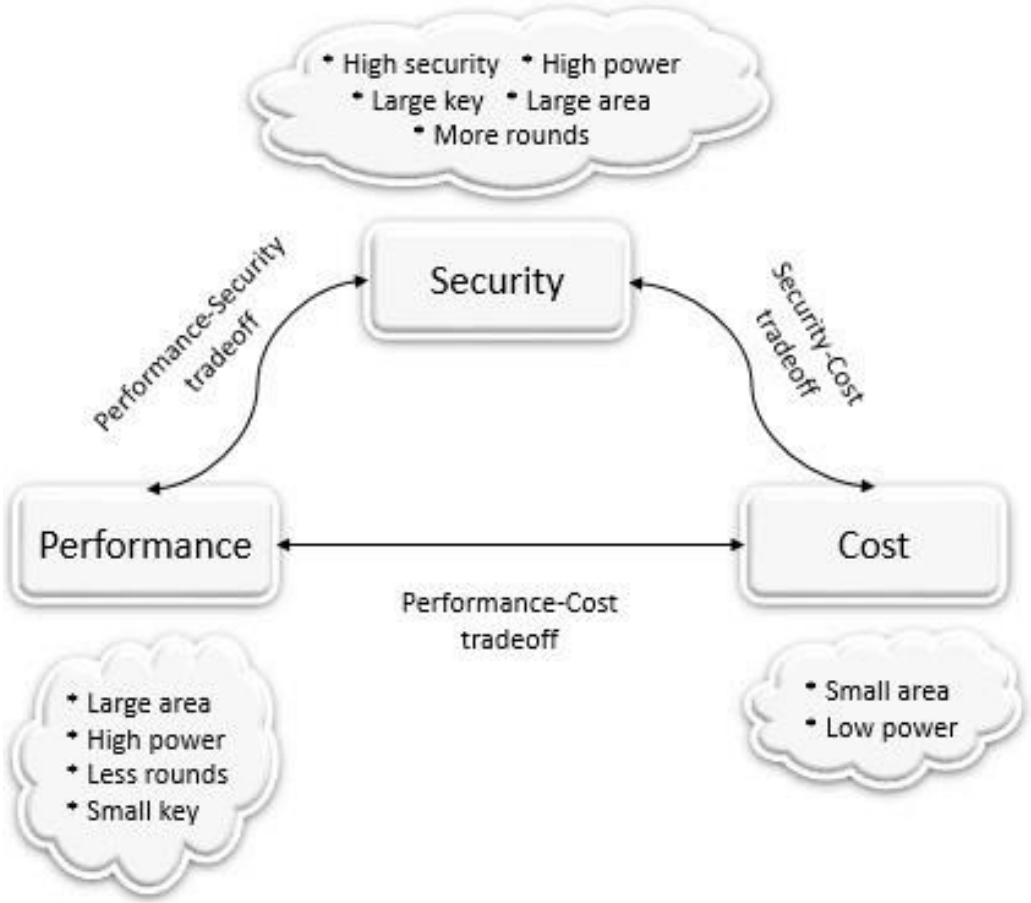

FIGURE 3 :Performance and Assorted Aspects

6. ECC BASED CRYPTOGRAPHIC FRAMEWORKS

In this article, we discuss cryptographic frameworks that utilize Elliptic Curve Cryptography (ECC) to ensure security. The cryptographic community recognizes the heightened complexity of the discrete logarithm and elliptic curve challenges, which form the foundation of the traditional Rivest-Shamir-Adelman (RSA) and Diffie Hellman public-key algorithms, compared to ECC's security.

ECC presents two primary advantages: first, ECC public keys are more compact while maintaining the same level of security as RSA or Diffie Hellman-based systems, reducing the amount of exchanged bits; second, ECC public-key operations require fewer computations than traditional public-key techniques [29].

The reduced size of keys offers benefits such as decreased bandwidth, energy, and storage usage, making them particularly suitable for sensor nodes with limited battery life. This also leads to lower processing and communication overhead. Liu and Ning introduced TinyECC in 2008, a customizable library for ECC operations in wireless sensor networks.

The main objective of TinyECC is to provide a readily available, open-source software package for ECC-based Public Key Cryptography (PKC) operations that can be easily tailored and integrated into sensor network applications. TinyECC incorporates various optimizations, each with distinct execution times and resource requirements, providing developers with flexibility when integrating TinyECC into sensor network applications [29].

7. PAIRING-BASED CRYPTOGRAPHIC FRAMEWORKS:

We delve into cryptographic frameworks using pairing-based cryptography (PBC) as the security mechanism. The emerging field of cryptography utilizing pairings, associated with ECC, has captured the attention of the global cryptography community. PBC allows for the development of innovative cryptographic schemes and enhances the performance of well-established cryptographic protocols. Table 1 show effective pairing computation for devices with limited resources. TinyPBC, introduced by Oliveira et al. in 2008, is based on the publicly accessible and open-source Multi-precision Integer and Rational Arithmetic C/C++ Library (MIRACL), a C library. The authors demonstrate how sensor nodes can exchange keys in a non-interactive, authorized manner. They present the fastest results for binary field multiplication on an 8-bit platform's quickest pairing calculation. TinyPBC computes pairings on an ATmega128L in approximately 5.45 seconds, taking only about half the time compared to NanoECC (10.96s).

For resource-limited devices, NanoPBC, presented by Aranha et al. in 2009, is a cryptography library where the authors constructed all significant number, finite field, and elliptic curve arithmetic from scratch to optimize platform performance. All time-critical procedures were developed using Assembly to enhance performance [30].

A crucial aspect contributing to performance was the implementation of López-Dahab's binary function f multiplication, which was fine-tuned to minimize memory access operations, as they are costly on the target platform. The authors achieved the fastest pairing calculation on an 8-bit platform and provided detailed insights into effective pairing computation for devices with limited resources.

Table 1 . Effective Pairing Computation For Devices With Limited Resources.

Framework	*Encryption*	*Cipher*	*Freshness (CTR)*	*Key Agreement*	*Code Requirement*	*Authentication*	*Cost (time/energy)*	*Support*
SPIN	CTR mode	RC5 (Block)	Yes	Master Key & Delayed Disclosure	2674B	CBC-MAC	7.24 ms	SmartDust
LEAP	RC5	RC5 (Block)	No	Pre-deployed (Master Varaible)	ROM: 17.9KB RAM: no. of neighbours	CBC-MAC	Variable (No. of neighbours)	Mica2
TinySec	CBC mode (Optional)	Cipher independent	No	Any	RAM: 728B program space: 7146B	CBC-MAC	RC5(C): 0.90ms Skipjack(C): 0.38ms RC5(C, assembly): 0.26ms	Mica, Mica2, & Mica2Dot
TinyPK	RSA	-	No	PK-RSA	13387B (512 bit key)	CA-signed Diffie-Hellman public value	3.8 s	Mica1, Mica2
TinyECC	ECIES	-	No	ECDH	20818B (micaz)	ECDSA	20266.47ms / 486.4 mJ (micaz)	Mica2/MicaZ, TelosB/Tmote Sky, BSNV3, & Imote2
TinyPBC	PBC	-	No	ID-NIKDS	Stack: 2,867B RAM: 368B ROM: 47,948B	ID-NIKDS	5.45s (pairing computation)	Mica2 & MicaZ

8. SECURITY GOALS AND ATTACK VECTORS

Wireless Sensor Networks (WSN) face a heightened vulnerability to security threats compared to wired networks due to their open communication nature. This openness exposes them to potential eavesdropping, interception, injection, and counterfeiting of transmitted data by adversaries. To uphold the essential protection goals encompassing Confidentiality, Integrity, Authenticity, Non-repudiation, Freshness, and Availability, cryptographic techniques can effectively thwart such attacks [31].

The discussion on Hybrid Cryptographic Frameworks is presented in Section 3. This section delves into symmetric and asymmetric cryptographic frameworks, amalgamating the strengths of both techniques. A groundbreaking approach, known as the Dynamically Secured Authenticated and Aggregation scheme (DSAA), was introduced by Jailin in 2011. DSAA uniquely combines public and symmetric key cryptography, enhancing security, computational speed, and memory utilization.

Addressing the imperative of maintaining secrecy within Wireless Sensor Networks, Tahir et al. (2008) proposed SCUR, a Lightweight Encryption Mechanism based on the Rabbit stream cipher. This mechanism not only meets the standards for energy efficiency but also ensures security. SCUR aims to minimize specific costs while maintaining requisite security levels, including (1) communication costs if the encrypted packet lifetime requires transmission and (3) utilized critical space [32].

9. FLIPPING ENCRYPTION

The outputs from sensors were included in the -bit string in the preceding section. Both the sensor nodes and the AFC already employed a pseudo-random function. Which bits of are to be flipped is determined by the function's output. The -th output of is flipped if it is inside the flipping interval; else, it is not. If the AFC and the sensor have the same initial, the same random sequence may be created synchronously at both locations, just like it can with a pseudo-random function. Information about the channel condition or a signal strength indication received can be used to identify (RSSI).

Because of the physical isolation and the channel autonomy, the EFC can only know the circuit state from itself to the sensors and not the circuit state from the AFC to the sensors. At each duplex interval, the AFC provides pilot signals to the sensor before the data transfer. The sensor receives the pilot channel and then delivers a develop significant to the AFC. The identical seed is initialised with RSSI in this manner by both the AFC and the sensor. The AFC creates the same after acquiring the sensor signal, and the sensor makes a new each time it gets the pilot signal [33].

In Wireless Sensor Networks (WSNs), data security is crucial. It is difficult to build and apply an encryption method that is safe yet won't use up a lot of the WSN devices' limited resources due to the architecture of WSN devices, which have limited resources.

10. LIGHTWEIGHT WSN SECURITY CONCEPT

The diagram in Figure 4 illustrates the foundational stages of filter-based processing, forming the basis for the proposed security paradigm designed to be lightweight. To sift through inappropriate (malicious or corrupted) communications, incoming messages undergo processing across stages with varying resource requirements, ranging from low (stage 1) to high (stage 6). This is accomplished in a resource-efficient manner.

It is recommended to implement Phases 1-4 to ensure sufficient security for the transmitted data. However, the consideration of stages 5 and 6 as optional depends on the availability of resources in a sensor node. The subsequent stage could potentially identify an adversary who successfully navigated the previous one. Even devices facing severe resource limitations can be accommodated by adopting suitable lightweight strategies for Phases 1-4.

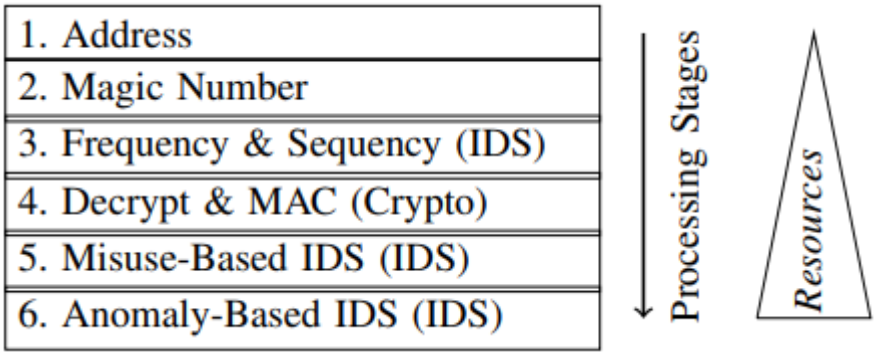

FIGURE 4: Filter Based Processing

Messages that do not align with the designated recipient's address are screened out during the Address step. A recommendation is put forth to employ a whitelist technique, ensuring that only transactions directed towards specified destination addresses are accepted. This prevents unnecessary energy expenditure when a message is not intended for the target node. In Stage 2, measures are proposed to mitigate issues such as battery depletion attacks. This involves the use of alternating codes or small, unique One-Time Login information for each communication, as outlined in reference [34].

The utilization of the Magic Number is suggested to swiftly ascertain the authenticity of a message without the need for initial decryption or MAC verification. These processes can be resource-intensive and contingent on the message size. Both parties, relying on a pre-established shared common secret from Stage 4, generate identical sequences of Magic Numbers. These numbers are then appended to the message header, facilitating easy detection of forgeries.

A re-synchronization option is available in case the pseudo-random new number deviates due to the periodic refreshing of the seed used in generating the pseudo-random alphanumeric code during each key renewal. This scenario may arise if an attacker successfully guesses the correct number within a brief timeframe.

To reduce the potential attack surface, opting for a sufficiently large width for the Magic Numbers is recommended. Given the resource-intensive nature of techniques like the Keccak-based Pseudo-Random Number Generator (PRNG), it is suggested to consider non-cryptographic but swift XorshiftPRNGs for a more lightweight number sequence.

As the integrated stage involves verifying message integrity, a non-cryptographically secure method suffices for promptly filtering out incorrect communications. Due to the delay in the subsequent stage's frequency filter, an adversary attempting to delay a message, acquire the genuine pseudorandom number, and replay a falsified message would be detected.

11. CONCLUSION

Wireless sensor networks are expanding and are being employed in a variety of applications. As a result, security becomes essential. However, the network of wireless sensors has several challenges, including low energy, restricted computing power, and low storage capacity, among others. To address this difficulty, several cutting-edge security protocols and procedures have been created. Cryptography is one method of ensuring security among many others. In order to provide security services in WSNs, choosing the suitable cryptography approach for sensor nodes is essential. In this article, we analyse the existing cryptographic frameworks. There is also a comparative of the suggested systems.

Funding

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] K. Biswas, V. Muthukkumarasamy, and E. Sithirasenan, "Maximal clique based clustering scheme for WSNs," in Proc. 8th IEEE International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Melbourne, Australia, 2013, pp. 237–241.

[2] K. Biswas, V. Muthukkumarasamy, E. Sithirasenan, and M. Usman, "An energy efficient clique based clustering and routing mechanism in WSNs," in Proc. 9th IEEE International Wireless Communications and Mobile Computing Conference (IWCMC), Italy, 2013, pp. 171–176.

[3] M. Shazly, E. S. Elmallah, J. Harms, and H. M. F. AboElFotoh, "On area coverage reliability of WSNs," in Proc. 36th IEEE Conference on Local Computer Networks (LCN), 2011, pp. 580–588.

[4] "AES proved vulnerable by Microsoft researchers," Computerworld Magazine, 2011.

[5] Intel Corporation, "Intel architecture software developer's manual," 1997.

[6] E. Biham, A. Birykov, and A. Shamir, "Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials," J. of Cryptology, vol. 18, no. 4, pp. 291–311, 2005.

[7] W. K. Koo, H. Lee, Y. H. Kim, and D. H. Lee, "Implementation and Analysis of New Lightweight Cryptographic Algorithm for WSNs," in Proc. International Conference on Information Science and Applications (ICISA), 2008, pp. 73–76.

[8] E. Yarrkov, "Cryptanalysis of XXTEA," 2010. [Online]. Available: http://eprint.iacr.org/2010/254.pdf.

[9] M. Amara and A. Siad, "Elliptic Curve Cryptography and its applications," in Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2011, pp. 247–250.

[10] H. Alzaid and M. A. S. A. Alfaraj, "End-to-End Data Security in Sensor Networks Using a Mix of Asymmetric and Symmetric Approaches," in Proc. 2nd IEEE International Conference on New Technologies, Mobility and Security.

[11] D. Aranha, J. Lopez, L. Oliveira, and R. Dahab, "NanoPBC: Implementing Cryptographic Pairings on an 8-bit Platform," in Proc. Conference on Hyperelliptic Curves, Discrete Logarithms, Encryption, etc. (CHiLE), Frutillar, Chile, 2009.

[12] B. Arazi, L. Elhanany, O. Arazi, and H. Qi, "Revisiting public-key cryptography for wireless sensor networks," IEEE Computer, vol. 38, no. 11, pp. 103–105, 2005.

[13] L. Batina, N. Mentens, K. Sakiyama, B. Preneel, and I. Verbauwhede, "Low-cost elliptic curve cryptography for wireless sensor networks," in Lecture Notes in Computer Science, vol. 4357, 2006, pp. 6–17.

[14] D. Carman, P. Kruus, and B. Matt, "Constraints and approaches for distributed sensor network security," NAI Labs, Technical Report, 2000.
[15] N. Fournel, M. Minier, and S. Ubeda, "Survey and Benchmark of Stream Ciphers for Wireless Sensor Networks," in Lecture Notes in Computer Science, vol. 4462, 2007.
[16] P. Ganesan et al., "Analyzing and modeling encryption over," in Proc. 2nd ACM Wireless Sensor Networks and Applications, New York, ACM Press, 2003.
[17] W. Hu and F. CecFleck, "A Public Key Technology Platform For Wireless Sensor Networks," in Proc. European Conference on Wireless Sensor Networks, Cork, Ireland, 2008.
[18] C. Karlof and N. Sastry, "TinySec: A Link Layer Security Architecture for Wireless Sensor Networks," in Proc. ACM Conference on Embedded Networked Sensor Systems (SenSys), Baltimore, MD, 2004.
[19] A. Canteaut, "Linear Feedback Shift Register," in Encyclopedia of Cryptography and Security, Springer, pp. 355–358.
[20] J. Burke, J. McDonald, and T. Austin, "Architectural support for fast symmetric-key cryptography," in Proc. International Conference on Application-Specific Systems, Architectures and Processors (ASAP), 2000, pp. 178–189.
[21] K. Choi and J. Song, "Investigation of feasible cryptographic algorithms for wireless sensor networks," in Proc. International Conference on Advanced Communication Technology (ICACT), 2006, p. 2.
[22] Perrig, A., R. Szewczyk, V. Wen, D. E. Culler, and J. D. Tygar. 2002. “SPINS: Security Protocols for Sensor Networks.” Wireless Networks, 8 (5): 521-534.
[23] Hwang, D. D., B. Lai, and I. Verbauwhede. July 2004. “Energy-memory-security tradeoffs in distributed sensor networks, in Proceedings of the 3rd International Conference on Ad-hoc Networks and Wireless (ADHOC-NOW), 70-81. Springer LNCS, Vol. 3158.
[24] M. Hell, T. Johansson, “Security Evaluation of Stream Cipher Enocoro-128v2,” CRYPTREC Technical Report, 2010.
[25] G. Gaubatz, J. Kaps, and E. Sunar, "Public key cryptography in sensor networks-Revisited 1st," .
[26] G. Gaubatz, J. Kaps, E. Ozturk, and E. Sunar, "State of the art in ultralow power public key cryptography for wireless sensor networks," in Proc. IEEE International Conference on Pervasive Computing and Communications Workshops, Washington, DC, USA.
[27] M. Healy, T. Newe, and E. Lewis, "Analysis of Hardware Encryption versus Software Encryption on Wireless Sensor Network Motes," in Proc. Smart Sensors and Sensing Technology, 2008.
[28] H. Kitayoshi, K. Sawaya, “Long Range Passive RFID tag for Sensor Networks,” Proc. Of 62nd IEEE Vehic. Tech. Conference, Dallas, USA, pp. 2696-2700, 2005.
[29] L. Mirowski, J. Hartnett, R. Williams, “An RFID Attacker Behaviour Taxonomy,” IEEE Pervasive Computing, Vol. 8, No. 4, pp. 79-84, 2009.
[30] Zhang, Y., J. Zheng, and H. Hu. 2008. Security in Wireless Sensor Networks, CRC Press, Taylor & Francis Group, Boca Raton, Florida, USA.
[31] Westhoff, D., J. Girao, and M. Acharya. 2006. “Concealed Data Aggregation for Reverse Multicast Traffic in Sensor Networks: Encryption, Key Distribution, and Routing Adaptation.” IEEE Transactions on Mobile Computing, 5 (10): 1417-1431.
[32] S-R. Lee, S-D. Joo, C-W. Lee, “An Enhanced Dynamic Framed Slotted ALOHA Algorithm for RFID Tag Identification,” Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, San Diego, California, pp. 166-174, July 2005.
[33] S. Rofde, T. Eisenbarth, E. Dahmen, J. Buchmann, C. Paar, “Fast Hash-Based Signatures on Constrained Devices,” CARDIS 2008, Surrey, UK, LNCS 5189, pp. 104-117, 2008.
[34] Deng, J., R. Han, and S. Mishra. November 2002. “INSENS: Intrusion-Tolerant Routing in Wireless Sensor Networks”, Technical Report CU-CS-939-02, Department of Computer Science, University of Colorado at Boulder, November 2002.