International Journal of Interdisciplinary Innovative Research & Development (IJIIRD)

ISSN: 2456-236X

Vol. 10 Issue 01 | 2025

Detection Node Replication and Ensuring Security in Wireless Sensor Networks- A Review

Mr. Yogeshwar L. Patil, Prof. Sudesh L. Farpat

[1,2] *Computer Engineering, Padm.Dr. V.B. Kolte COE Malkapur, Maharashtra, India*

DOI: 10.5281/zenodo.15751414

ABSTRACT

Wireless Sensor Networks (WSNs) are widely used in critical applications such as military surveillance, environmental monitoring, and healthcare systems, where they collect and transmit sensitive data like enemy movements or the location of individuals. Because wireless sensor networks (WSNs) deal with important and sensitive information, keeping them secure is very important. One of the biggest threats to WSNs is a replica node attack, where an attacker makes and inserts fake copies of real sensor nodes into the network. These fake nodes can then steal data, disrupt communication, or damage the network's performance.. These replicas can compromise data integrity, disrupt network operations, and pose serious security risks. This paper presents a comprehensive review of replica detection algorithms designed to address this threat. It examines various detection techniques, including centralized, distributed, location-based, and cryptographic methods, highlighting their strengths, limitations, and suitability for different WSN environments. The goal of this review is to provide insights into current strategies for replica detection and to identify potential directions for future research in securing WSNs.

Keyword: - Wireless sensor network (WSN), attack detection ,replication,network security, SPRT

1. .INTRODUCTION

Wireless Sensor Networks (WSNs) are used in a wide range of fields, including traffic management, disaster response, healthcare, environmental monitoring, and military operations. In these applications, sensor nodes are typically deployed in areas where they can be easily observed and maintained. However, WSNs face several limitations such as restricted memory capacity, limited battery life, challenging deployment environments, and reliance on insecure wireless communication.

Because they use wireless communication, WSNs are vulnerable to many different types of attacks. These include signal or radio jamming, node replication, denial of service (DoS), Sybil attacks, and coverage gaps. Such threats can be layer-specific—targeting different layers of the OSI model—or they can be independent of the layer structure. Attacks may also focus on specific network operations like routing, node positioning, data aggregation, or time synchronization.

To counter these security threats, a variety of defense strategies have been proposed. Many of these methods can effectively detect and reduce the impact of various types of attacks on the network.. Nonetheless, the development of efficient attack detection mechanisms remains essential to accurately identify and respond to the source of malicious activity.

Recent advancements in robotics have enabled the development of various architectures for autonomous wireless sensor networks (WSNs). If an adversary gains control of a single node, they can extract its cryptographic keys and create numerous replica nodes that mimic the original, using the same identity and security credentials. These fake nodes can then be widely deployed throughout the network. To counter this threat, one effective solution is the use of tamper-resistant hardware, which protects the stored key material from being accessed or extracted by attackers..

1.1 Tamper-resistant hardware

To secure cryptographic keys, tamper-proof devices are used. These hardware solutions are effective in applications such as secure communications and electronic payments. They offer a strong layer of protection by preventing

unauthorized access or tampering with stored keys. Typically, access to such devices requires authentication like a PIN or password.

1.2 Sequential Hypothesis Testing

This statistical method doesn't rely on a fixed sample size. Instead, it evaluates data as it is collected and stops further sampling when a predefined condition is met, reducing both time and cost.

1.3 Sequential Probability Ratio Test (SPRT)

SPRT is a specific form of sequential hypothesis testing originally used in industrial quality control and later applied in automated human testing systems. It helps determine if data supports rejecting a null hypothesis at a given confidence level, allowing efficient, data-driven decisions in network monitoring and intrusion detection.

1.4 Null and Alternative Hypotheses

The null hypothesis represents a default assumption—often, no effect or no difference—while the alternative hypothesis proposes a specific effect or relationship. In statistical testing, we use the data to decide whether to reject or not reject the null hypothesis. This helps us understand how the network is behaving and whether there might be any possible attacks.

2. SECURITY MANAGEMENT IN WSNS

In WSNs, nodes manage data routing and other functions autonomously. Ensuring network security is complex due to the limited resources and dynamic topology of these systems. Security can be assessed across the following principles:

A. Confidentiality (Access Control): Only authorized users should access sensitive data. Unauthorized access must be strictly prevented.

B. Integrity: Only trusted parties can modify data or system resources. This includes creating, deleting, or altering information, ensuring messages remain uncorrupted.

C. Authenticity: Messages should originate from legitimate nodes and be verifiable using shared cryptographic keys.

D. Anonymity and Privacy: Information about users or node ownership must remain private and protected from unauthorized disclosure.

E. Access Rights: User permissions must be granted by administrators, limiting system access to verified individuals.

F. Availability: means that important data and system resources should always be accessible to the right users whenever they need them, without delays or interruptions.

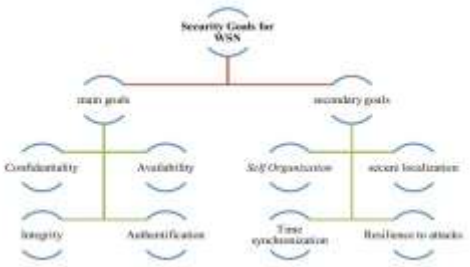

Fig -1 Security goals in WSN

3. IMPACT OF NODE REPLICATION ATTACKS ON WSN SECURITY

Both mobile and stationary Wireless Sensor Networks (WSNs) face similar security challenges, often requiring high-level protection measures. The main security goals in wireless sensor networks (WSNs) are authenticity, confidentiality, availability, and data integrity. However, these goals are severely threatened when a node replication attack occurs.

In this type of attack, an attacker copies a real sensor node by using its credentials and secret keys. If the network lacks a robust detection system to identify and eliminate these cloned nodes, the fake nodes can fully participate in network activities—encrypting, decrypting, and verifying communications just like the original node. Moreover,

because these replicas use valid credentials, they are mistakenly recognized as legitimate network members, making detection even harder.

As a result, node replication attacks can have severe consequences, compromising the integrity and security of the network. Attackers can:

- Monitor large portions of network traffic,
- Eavesdrop on sensitive data,
- Launch denial of service (DoS) attacks,
- Inject false sensor data,
- Disrupt data collection and reporting,
- Tamper with or block communications by exploiting network protocols.

One key security goal—authenticity—is particularly affected. Authenticity ensures that a node is communicating with a trusted sensor. But in replication attacks, the fake nodes appear genuine because they carry the same credentials as the original node. This makes it very difficult for the system to tell apart a real node from its replica, especially since current security techniques cannot detect such impersonation due to the valid credentials being used.

Data Integrity means making sure that the information sent from one device to another stays accurate and unchanged during transmission. It makes sure that the message received has not been changed and is exactly the same as the one that was sent.However, in a node replication attack, Attackers can use fake or copied sensor nodes to send false data or change the actual data in the network. This can lead to incorrect or misleading information being collected and shared, which damages the trust and reliability of the network's data.

4. RELATED WORKS

In this section, we look at different methods that have been studied. A review has been carried out on existing techniques, and their performance has been evaluated.

In [1], the authors suggested a method called SET. In this method, the entire area is divided into smaller parts, where each sensor node is close enough to talk to its neighbors directly (one-hop distance). Each small part has a leader, and these leaders form a tree structure, with the base station (BS) as the root.

Each leader sends its identity to the leader above it in the tree (its parent). The parent leader then checks for duplicate identities using an intersection operation. If there's any overlap, it means a duplicate node exists, and this is reported to the base station.

This method doesn't require any node or leader to store extra information like membership lists, so there's no extra storage needed. However, communication cost is higher—about O(N)—since every node takes part in choosing the subset leaders by sending out messages. Also, each leader still needs to send membership info to the base station.

In [2], the authors propose a distributed algorithm to detect clone attacks based on mobile energy prediction. A set of monitored sensor nodes periodically receives location claim messages from other nodes. Each message contains the node's ID, location, energy consumed during mobility, and the timestamp when the location claim was made. When a node receives the same location claim from two different nodes, it compares the predicted energy consumption during movement and the timestamp to identify any clone nodes. The results show that the proposed method helps reduce energy consumption while achieving high detection accuracy.

In [3], the author proposes two schemes for clone detection: a randomized multicast scheme and a line-selected multicast (LSM) scheme.

- Randomized Multicast Scheme: In this scheme, a node broadcasts its location claim, which is then sent by its neighbors to a randomly selected set of witness nodes. If a replica exists in the network, the replica's location claim is also sent to these witness nodes. Based on the birthday paradox theory, if each location claim is sent to N witness nodes, there is a high likelihood that at least one node will receive two different location claims from different nodes, causing a conflict. This conflict helps to detect replicas in the network.
- Line-Selected Multicast Scheme: In this scheme, the location claim is stored at every mediator node before being forwarded to the next hop, eventually reaching the witness node. This process creates a chain or path among the nodes. If the chain of location claims is broken, indicating a conflict of valid nodes, the mediator node detects it as a replica. Compared to the randomized multicast scheme, this scheme has a lower communication cost but higher storage overhead. For a network of size N, the communication overhead in the randomized multicast scheme is O(N), while in LSM, it is O(N^2), with an average path length of O(N). the randomized multicast scheme is more communication-efficient, while the line-selected multicast scheme reduces communication costs at the expense of increased storage overhead.

In [4], the authors propose a fingerprint-based scheme for clone detection. In this approach, each node calculates its fingerprint using information from its neighboring nodes. Before the network is deployed, each node is given a

unique codeword, created using superimposed s-disjunct codes. Using the codeword, each node creates its own unique fingerprint and also calculates the fingerprints of its neighboring nodes.

Each node then stores the fingerprints of its neighbors, and whenever it sends a message to the base station, it includes its own fingerprint as part of the message. When a node receives a message, it first verifies the authenticity of the sender by comparing the fingerprint included in the message with its own records. If the fingerprints do not match, a conflict is reported to the base station, and the sender is identified as a clone node.

In this scheme, clone detection can be performed either by the base station or by the nodes in the network. To calculate the fingerprint, each node sends its codeword to its neighbors. This process causes a communication overhead of O(N), where N is the total number of nodes in the network.Additionally, The scheme also needs O(d) storage space to save the codewords, where d is the number of neighbors each node has.

In [5], the authors propose a protocol where a node includes a list of its neighbors as part of a claim. A reporter node is selected from the nearby nodes to assist to help send the message. The process works as follows:

1. The claimer node sends a request for a signature to the reporter node.
2. The claimer verifies the signature received from the reporter before sending the claim.
3. The reporter node gets the claim and sends it to a designated witness node to verify its authenticity.
4. If at any point the witness node detects a conflict in the claim, the node making the claim is identified as a clone.

This method takes extra effort and resources because it has to choose a reporter node. Since the reporter node is responsible for sending the subset of the claim to the witness node, the communication overhead is O(N * g * N), where N is the total number of nodes and g is the number of reporter nodes. The storage overhead is O(g), as each node needs to store information related to its reporter nodes.this protocol helps detect clone nodes by using a reporting mechanism, but it introduces significant communication and storage overhead due to the involvement of reporter nodes.

In [6], the authors propose a tree-based clone detection scheme that works in two phases:

Phase 1: A binary search tree (BST) is created to arrange the network nodes in an organized way.

Phase 2: Clone detection is carried out based on the node's key, which in this case is the node's ID. The node ID serves as the primary evidence for identifying clone nodes.

The performance of the proposed scheme is evaluated using simulations in NS2, focusing on key performance parameters such as delay, residual energy, packet loss, and packet delivery ratio. The results are presented in three different scenarios, showing that the proposed scheme achieves lower communication costs and reduced memory usage compared to other methods.this tree-based approach helps efficiently detect clone nodes while keeping communication and memory costs low.

In [7], the authors propose a neighbor-based detection scheme for identifying clone nodes in a network. The process works as follows:

• When a node moves to a new place, it sends a re-joining message to its new nearby nodes. This message includes a list of the node’s old neighbors.

• Upon receiving this message, the new neighbor randomly selects a node from the old neighbor list and sends a verification request to that node to check the authenticity of the moving node.

• The selected old neighbor then checks its neighbor table to verify whether the node ID still exists at the old location.

O If the ID does not exist, it confirms the move as valid.

O If the ID is found, it assumes the node is still present at the old location and reports a clone to the base station.

• To verify the node's presence, the old neighbor may also send a direct message to the suspected node.

• If the new neighbor does not receive a cancellation message within a predefined time, it accepts the new node as a valid neighbor.

This verification process increases communication overhead, as every new neighbor sends a verification message with a certain probability p. The scheme results in O(d × p × N) communication overhead and O(d × p) storage overhead, where d is the average number of neighbors and N is the total number of nodes.

this scheme effectively detects clones through neighbor verification but incurs higher communication costs due to the random checks involved in the authentication process.

ISSN: 2456-236X

Vol. 10 Issue 01 | 2025

5. CONCLUSIONS

Nowadays, secure routing has become a critical area of research and has attracted significant attention from researchers. Various protocols have been developed to protect against routing attacks, each using different strategies. Some protocols rely on a central authority, while others use alternative mechanisms to distribute cryptographic keys for securing routing processes.

This paper presents an in-depth analysis of various algorithms used to detect replica nodes in wireless sensor networks. It also presents a classification of these techniques based on their methods and evaluates them in terms of communication and storage overhead.

6. REFERENCES

[1].] H. Choi, S. Zhu and T. F. La Porta, "SET: Detecting node clones in sensor networks," 2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops - SecureComm 2007, Nice, France, 2007, pp. 341-350.

[2]. Jeyaselvi, M. and C. Jayakumar. "Distributed Clone Attack Detection Algorithm Using Mobility Energy Prediction in Mobile Wireless Sensor Networks.", Sensor Letters, Vol. 16, Number 12, pp. 965-972, 2018.

[3]. B. Parno, A. Perrig and V. Gligor, "Distributed detection of node replication attacks in sensor networks," 2005 IEEE Symposium on Security and Privacy (S&P'05), Oakland, CA, USA, 2005, pp. 49-63.

[4]. K. Xing, F. Liu, X. Cheng and D. H. C. Du, "Real-Time Detection of Clone Attacks in Wireless Sensor Networks," 2008 The 28th International Conference on Distributed Computing Systems, Beijing, 2008, pp. 3-10.

[5]. Xiangshan Meng, Kai Lin, and Keqiu Li. A Note-Based Randomized and Distributed Protocol for Detecting Node Replication Attacks in Wireless Sensor Networks. In Algorithms and Architectures for Parallel Processing, pages 559 – 570. Lecture Notes in Computer Science, Springer Berlin Heidelberg, May 2010. [6]. Sachin Lalar, Shashi Bhushan & Surender, An efficient tree-based clone detection scheme in wireless sensor network, Journal of Information and Optimization Sciences, 40:5, 1003-1023, 2019.

[7]. L. Ko, H. Chen and G. Lin, "A Neighbor-Based Detection Scheme for wireless sensor networks against node replication attacks," 2009 International Conference on Ultra Modern Telecommunications & Workshops, St. Petersburg, 2009, pp. 1-6.