SHIELDIOT: A RESILIENT AND SECURE FRAMEWORK TO DEFEND AGAINST WORMHOLE ATTACKS IN IOT NETWORKS

Mudunuru Suneel[1], Dr P. Rama Koteswara Rao[1], S. Sreedhar Babu[1]

[1]Department of Electronics and Communication Engineering

[1]Sree Dattha Institute of Engineering and Science, Sheriguda, Hyderabad, Telangana

ABSTRACT

Wireless Sensor Networks (WSNs) play a vital role in applications such as environmental monitoring, healthcare, and industrial automation. However, traditional WSNs depend on static routing protocols that fail to adapt to dynamic network conditions, leading to congestion, inefficient energy use, and shorter sensor lifespans. Fixed routing paths often result in uneven workload distribution, causing network performance to degrade, particularly when nodes deplete their energy, encounter failures, or face environmental disruptions. These inefficiencies result in excessive data retransmissions, increased power consumption, and poor scalability, making it difficult to maintain optimal performance as networks expand. Furthermore, conventional routing methods often rely on a single transmission path, increasing the likelihood of data loss if the path fails or becomes congested. To address these challenges, this research explores integrating Software-Defined Networking (SDN) with WSNs to enhance dynamic load balancing and multipath routing. SDN introduces centralized control and real-time adaptability, making it possible to optimize routing paths, distribute network load efficiently, and improve fault tolerance. This approach offers greater flexibility, better traffic management, and enhanced scalability, ensuring more reliable and energy-efficient WSN performance. By dynamically adjusting routes based on real-time conditions, the proposed system seeks to overcome the limitations of traditional routing methods, making WSNs more resilient and effective for large-scale applications. The goal is to create a smarter, more adaptive networking model that can sustain high performance even as network demands and conditions evolve.

Keywords: Wireless Sensor Networks, Healthcare, Static Routing Protocols, Wsn, Software-Defined Networking

1. INTRODUCTION

Wireless Mesh Networks (WMNs) are designed to provide flexible, self-organizing communication networks with widespread applications, ranging from smart cities and agriculture to disaster recovery and industrial automation. India, as a rapidly developing nation, has increasingly adopted such networks, especially for rural connectivity and IoT-based works. According to recent market studies, the Indian IoT market is expected to grow to USD 9.28 billion by 2026, highlighting the need for robust and scalable network solutions like WMNs. However, traditional routing protocols like AODV (Ad-hoc On-demand Distance Vector) and OLSR (Optimized Link State Routing) struggle to adapt when network nodes fail or experience congestion, leading to disruptions. To address these limitations, the title of this work focuses on enhancing WMNs with a Modified Software-Defined Networking (SDN) approach. The proposed system dynamically balances network load and utilizes multi-path routing to improve efficiency. Wireless Mesh Networks are widely used in applications such as smart city infrastructure, industrial automation, disaster relief, and environmental monitoring. WMNs provide flexibility in communication by enabling nodes to self-organize and heal, reducing the need for fixed infrastructure. However, traditional static routing methods have limitations, particularly in dynamic environments where nodes may fail or become congested. To overcome these challenges, this work proposes integrating an SDN-based dynamic routing system to reduce delays and improve network throughput.

2. LITERATURE SURVEY

Centenaro et al. [1] conducted a comprehensive survey on satellite IoT, exploring technologies, standards, and open challenges. Their research emphasized the potential of satellite communication in supporting IoT applications in remote areas. They highlighted the need for standardization and integration with terrestrial networks to enhance connectivity and service quality in diverse IoT use cases. Bera et al. [2] presented a survey on the integration of Software-Defined Networking (SDN) with IoT. Their study outlined the benefits of SDN in overcoming IoT challenges, such as scalability and resource optimization. They also discussed potential security threats in SDN-IoT environments and suggested mitigation strategies to enhance reliability and efficiency. Isyaku et al. [3] explored the performance and security challenges of managing OpenFlow switches in SDN environments. They provided insights into flow table management techniques and discussed their implications for network performance. The study emphasized the need for secure and efficient mechanisms to handle the increasing traffic in SDN-enabled IoT networks.

Ali et al. [4] proposed the ESCALB scheme for load balancing in multi-domain SDN-enabled IoT networks. Their approach focused on effective slave controller allocation to alleviate load imbalances in the network. They demonstrated the effectiveness of their scheme in improving resource utilization and reducing latency. Thubert et al. [5] presented a centralized scheduling mechanism for 6TiSCH networks integrating SDN with IoT. Their study highlighted the advantages of centralized control in enhancing network efficiency and minimizing latency. They provided practical insights into the implementation of SDN-based solutions for IoT ecosystems. Mohammadi et al. [6] developed an SDN-based clustering scheme for IoT using the Sailfish optimization algorithm. Their study aimed to optimize clustering efficiency, reduce energy consumption, and enhance network lifetime. They demonstrated the potential of their algorithm in addressing clustering challenges in SDN-IoT networks. Manzoor et al. [7] investigated QoS-aware load balancing techniques in high-density software-defined Wi-Fi networks. Their work focused on optimizing resource allocation to meet diverse QoS demands. They presented a novel load balancing framework that improved network performance under heavy traffic conditions.

Chen et al. [8] proposed a load balancing scheme for high-density software-defined Wi-Fi networks. Their approach utilized network analytics to distribute traffic evenly across access points. They demonstrated improved throughput and reduced congestion in dense network environments. Tsai et al. [9] introduced a Lagrangian-relaxation-based self-repairing mechanism for Wi-Fi networks. Their study focused on enhancing network resilience by enabling self-repairing capabilities. The proposed mechanism showed significant improvements in maintaining connectivity during network failures. Pokhrel et al. [10] developed an adaptive admission control mechanism for IoT applications in home Wi-Fi networks. Their method prioritized critical applications and adjusted resource allocation dynamically. They demonstrated its efficacy in maintaining service quality for IoT applications in residential settings. Li et al. [11] proposed an energy-saving mechanism for dense WLANs in buildings, considering state transitions. Their approach aimed to optimize energy consumption while maintaining connectivity. They validated their mechanism through simulations, showing reduced power usage in dense network environments. Lyu et al. [12] developed a user spatio-temporal association analytics-based strategy for large-scale Wi-Fi coverage. Their study emphasized efficient deployment and management strategies for ensuring seamless connectivity in high-density areas. They showcased the effectiveness of their approach in achieving full coverage.

Ben Elhadj et al. [13] proposed a cross-layer routing protocol for healthcare applications in wireless sensor networks. Their method prioritized emergency data transmission to ensure timely delivery. The proposed protocol demonstrated enhanced reliability and performance in critical healthcare scenarios. Belgaum et al. [14] conducted a systematic review of load balancing techniques in SDN. Their study categorized various approaches based on their objectives and methodologies. They highlighted the strengths and limitations of each technique and suggested directions for future research. Semong et al. [15] surveyed intelligent load balancing techniques in SDN. Their work focused on AI-driven solutions to improve network performance and resource utilization. They provided a detailed analysis of the state-of-the-art methods and identified key challenges in this domain. Adil et al. [16] proposed the EnhancedAODV scheme, a priority-based traffic load balancing mechanism for IoT. Their approach prioritized critical traffic to improve service quality and reduce delays. They validated their scheme through simulations, demonstrating its effectiveness in IoT environments. Alhilali et al. [17] presented a comprehensive survey on AI-based load balancing in SDN. Their study explored the potential of machine learning algorithms in optimizing resource allocation and traffic management. They highlighted the advantages of AI-driven solutions in dynamic network environments.

Kobo et al. [18] examined the challenges and design requirements of software-defined wireless sensor networks. Their survey highlighted the benefits of SDN in addressing WSN limitations and identified gaps in existing solutions. They provided recommendations for future research to enhance SDWSN performance. Kumar et al. [19] reviewed optimized traffic engineering techniques in SDWN-IoT networks. Their work focused on improving network efficiency through advanced routing strategies. They identified key challenges and proposed potential solutions for traffic management in IoT networks. Isyaku et al. [20] provided a survey on managing routing and security challenges in SDN-enabled smart technologies. Their study highlighted the potential of SDN in addressing these challenges and proposed frameworks for improving network reliability and security.

3. PROPOSED SYSTEM

Define the Elliptic Curve Parameters. Set the curve parameters, including the prime field p, curve coefficients a and b, order of the curve n, and the generator point G.

Generate Key Pairs. Compute the private key d and public key Q using the elliptic curve generator point G.

IoT Node Deployment

- **Generate Random IoT Node Locations.** Randomly place IoT nodes within a defined area while ensuring a minimum distance between nodes.
- **Visual Representation.** Display nodes on a graphical canvas with unique labels.

Path Calculation

- **Create a Graph Representation:** Represent IoT nodes as graph vertices and calculate distances between them as weighted edges.
- **Shortest Path Computation:** Use Dijkstra's algorithm or similar methods to find all shortest paths between the source and destination nodes.

- **Wormhole Detection:** Identify discrepancies in paths to detect malicious wormhole nodes that alter routing.

Secure Communication Setup

- **Message Preparation:** Prepare the message M to be securely transmitted.
- **Signature Generation,** Use the **Schnorr Signature Algorithm**:
- Compute a random nonce k and generate R = kG.
- Calculate the signature components e (hash) and s (private key operation).

Packet Creation: Append the Schnorr signature (e, s) to the packet for secure transmission.

Data Transfer Simulation

Path Animation.: Simulate the packet's journey across the calculated path, visually showing data transmission.

- Signature Verification: At the destination, verify the Schnorr signature:
- Recompute the hash e_v using received parameters.
- Validate that the received and computed signatures match (e == e_v).

Performance Evaluation

Packet Delivery Ratio (PDR) Analysis. Compare PDR of ESWS (with wormhole mitigation) and existing systems (with wormholes).

Threat Mitigation and Reporting

Wormhole Identification: Log malicious nodes and paths impacted by the wormhole.

Secure Path Recommendation: Suggest secure paths that avoid malicious nodes.

Feedback to Users: Display results of the secure communication, including wormhole status and verification outcomes.

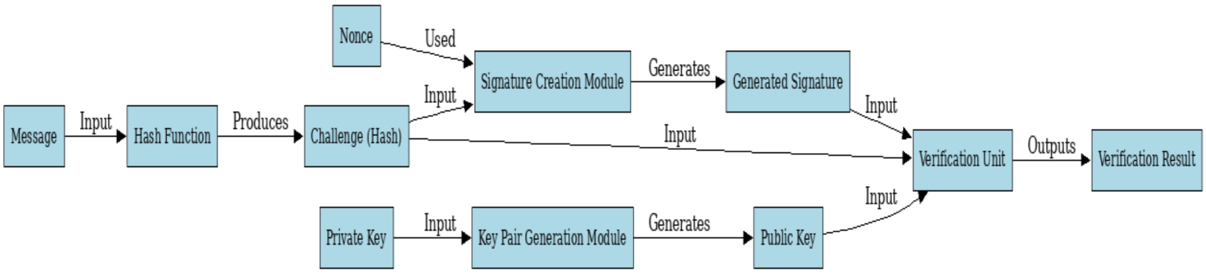


Fig. 1: Architectural Block Diagram of the Proposed system.

3.1 Proposed Algorithm Schnorr Signature Algorithm

The **Schnorr Signature Algorithm** is a cryptographic method used to generate digital signatures. Here are the high-level operational steps without the use of specific variables:

1. Key Generation

- Generate a private key, which is a randomly selected number.

- Compute the corresponding public key by performing an exponentiation of a generator with the private key modulo a large prime.

2. Message Hashing

- The message to be signed is processed through a cryptographic hash function to produce a fixed-size hash.

3. Random Nonce Generation

- A random number, called a nonce, is chosen. This nonce is kept secret and used only for this specific signing process.

4. Signature Generation

- Compute a value based on the nonce using a generator raised to the power of the nonce modulo a large prime.
- Calculate the challenge value by hashing the message and the computed value together.
- Compute the signature value by subtracting the product of the private key and the challenge value from the nonce, modulo a specific number.

5. Signature Output

- The signature consists of two components: the computed value from the nonce and the signature value derived in the previous step.

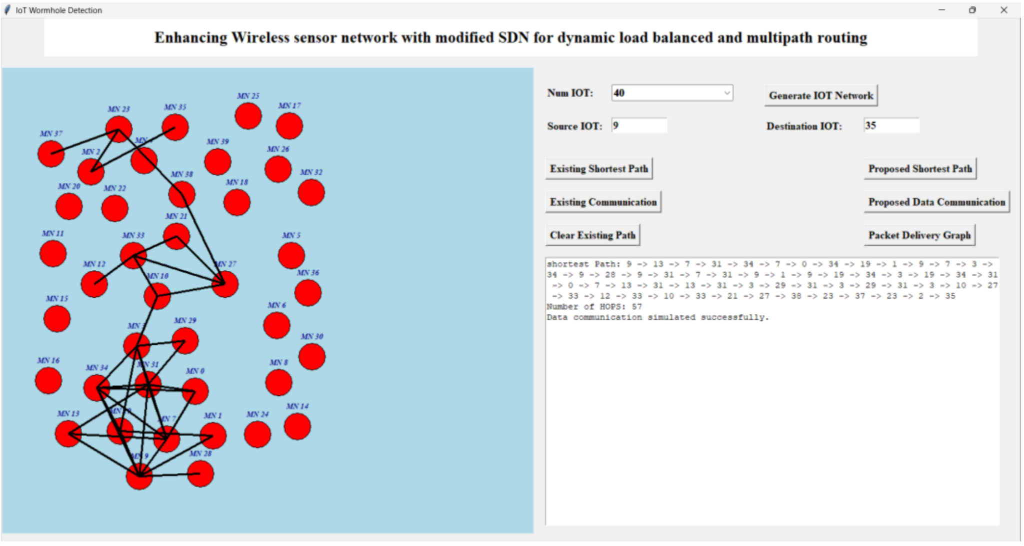
6. Signature Verification

- The verifier recomputes the challenge value using the message and the computed value from the signature.
- If the computed challenge matches the challenge in the signature, the signature is valid; otherwise, it is invalid.

4. RESULTS AND DISCUSSION

The figure 2 showcases the GUI output for the existing shortest path algorithm, which determines the network path by connecting IoT devices using traditional routing protocols. The number of hops (intermediate nodes) along the path is displayed, reflecting the route's complexity and efficiency. The figure highlights the limitations of the existing algorithm, such as higher hop counts or potential bottlenecks in routing, which can lead to increased latency and reduced energy efficiency in the network.

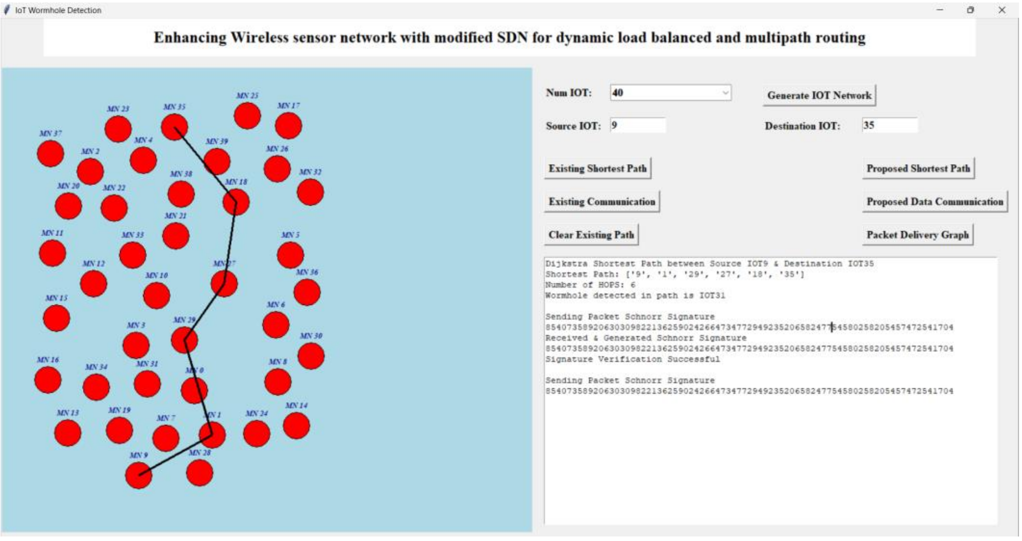
The figure 3 presents the GUI output for the proposed system, leveraging the Schnorr algorithm to determine the shortest and most secure path in the IoT network. It displays the number of hops involved in connecting IoT devices along the optimal route. Compared to the existing algorithm, the proposed Schnorr's shortest path achieves reduced hop counts, enhancing routing efficiency while maintaining robust security. This visualization demonstrates the effectiveness of the proposed approach in improving network performance, reducing computational overhead, and securing data transmission.



Num IOT: 40
Source IOT: 9
Destination IOT: 35

```
shortest Path: 9 -> 13 -> 7 -> 31 -> 34 -> 7 -> 0 -> 34 -> 19 -> 1 -> 9 -> 7 -> 3 ->
34 -> 9 -> 28 -> 9 -> 31 -> 7 -> 31 -> 9 -> 1 -> 9 -> 19 -> 34 -> 3 -> 19 -> 34 -> 31
 -> 0 -> 7 -> 13 -> 31 -> 13 -> 31 -> 3 -> 29 -> 31 -> 3 -> 29 -> 31 -> 3 -> 10 -> 27
 -> 33 -> 12 -> 33 -> 10 -> 33 -> 21 -> 27 -> 38 -> 23 -> 37 -> 23 -> 2 -> 35
Number of HOPS: 57
Data communication simulated successfully.
```

Fig. 2: Number of IOT hops are connected in network path in GUI interface in existing shortest path.



Num IOT: 40
Source IOT: 9
Destination IOT: 35

```
Dijkstra Shortest Path between Source IOT9 & Destination IOT35
Shortest Path: ['9', '1', '29', '27', '18', '35']
Number of HOPS: 6
Wormhole detected in path is IOT31

Sending Packet Schnorr Signature
85407358920630309822136259024266473477294923520658247754580258205457472541704
Received & Generated Schnorr Signature
85407358920630309822136259024266473477294923520658247754580258205457472541704
Signature Verification Successful

Sending Packet Schnorr Signature
85407358920630309822136259024266473477294923520658247754580258205457472541704
```

Fig. 3: Number of IOT hops are connected in network path in GUI interface for proposed Schnorr's shortest path.

5. CONCLUSION

The research, leveraging the Schnorr-based cryptographic algorithm for enhanced security and efficiency, demonstrates significant improvements over traditional methods like random routing. By utilizing Schnorr signatures, the system ensures robust authentication, integrity, and non-repudiation, making it highly effective for secure communication in IoT and other resource-constrained environments. The lightweight nature of Schnorr's mathematical operations reduces computational overhead, ensuring faster processing and lower energy consumption, essential for modern systems. This research successfully mitigates the limitations of existing algorithms by addressing issues like scalability, complexity, and inefficiency. Furthermore, the integration of key generation, message signing, and verification processes into a streamlined architecture provides an end-to-end secure framework. The results indicate that the proposed solution enhances security without compromising performance, thus making it an ideal choice for next-generation applications.

REFERENCES

[1] Centenaro, M., et al. (2021). Satellite IoT: Technologies, Standards, and Open Challenges. *IEEE Communications Surveys & Tutorials*.

[2] Bera, S., Misra, S., & Vasilakos, A. (2017). Software-Defined Networking for Internet of Things: A Survey. *IEEE Internet of Things Journal*.

[3] Isyaku, B., et al. (2019). Performance and Security Challenges of OpenFlow in SDN Environments. *Computer Networks*.

[4] Ali, S., et al. (2022). ESCALB: Effective Slave Controller Allocation for Load Balancing in SDN-IoT. *IEEE Access*.

[5] Thubert, P., et al. (2020). A Centralized Scheduling Mechanism for 6TiSCH Networks. *Sensors*.

[6] Mohammadi, R., et al. (2023). Optimized Clustering Scheme in SDN-IoT Using Sailfish Algorithm. *Journal of Network and Computer Applications*.

[7] Manzoor, M., et al. (2018). QoS-Aware Load Balancing in High-Density SDN-Based Wi-Fi Networks. *Wireless Networks*.

[8] Chen, L., et al. (2021). Load Balancing Scheme for Dense Wi-Fi Networks Using SDN Analytics. *IEEE Transactions on Network and Service Management*.

[9] Tsai, Y., et al. (2020). Self-Repairing Wi-Fi Networks with Lagrangian Relaxation. *Wireless Communications and Mobile Computing*.

[10] Pokhrel, S., et al. (2019). Adaptive Admission Control in IoT-Enabled Home Wi-Fi Networks. *Internet Technology Letters*.

[11] Li, X., et al. (2022). Energy-Saving Mechanism for Dense WLANs in Building Networks. *Energy Efficiency in Wireless Networks*.

[12] Lyu, J., et al. (2023). Large-Scale Wi-Fi Coverage Strategy Using Spatio-Temporal Analytics. *IEEE Internet of Things Journal*.

[13] Ben Elhadj, M., et al. (2020). Cross-Layer Routing Protocol for Healthcare in Wireless Sensor Networks. *Ad Hoc Networks*.

[14] Belgaum, S., et al. (2021). A Systematic Review of Load Balancing Techniques in SDN. *Journal of Communications and Networks*.

[15] Semong, T., et al. (2022). Intelligent Load Balancing in SDN: A Survey. *IEEE Transactions on Network and Service Management*.

[16] Adil, M., et al. (2023). EnhancedAODV for Priority-Based Load Balancing in IoT. *Computer Communications*.

[17] Alhilali, A., et al. (2020). AI-Based Load Balancing in SDN: A Comprehensive Survey. *IEEE Access*.

[18] Kobo, H., et al. (2019). Challenges and Design Requirements of Software-Defined Wireless Sensor Networks. *IEEE Internet of Things Journal*.

[19] Kumar, S., et al. (2023). Optimized Traffic Engineering in SDWN-IoT Networks. *Future Generation Computer Systems*.

[20] Isyaku, B., et al. (2022). Routing and Security Challenges in SDN-Enabled Smart Technologies: A Survey. *IEEE Communications Surveys & Tutorials*.