Enhancing Security and Traffic Management in SDN-Enabled Wireless Sensor Networks

Guang Yang

Western Norway University of Applied Sciences
Bergen, Norway
guang.yang@hvl.no

Abstract. Wireless Sensor Networks (WSNs) play a critical role in Internet of Things (IoT) applications such as environmental monitoring, healthcare, and industrial automation. However, WSNs face severe challenges, including limited energy resources, unreliable communication links, and growing exposure to security threats. Software-Defined Networking (SDN) provides centralized programmability and global visibility, enabling dynamic traffic management and security enforcement. In this work, we propose an SDN-based WSN solution that integrates traffic analysis for anomaly detection and security policy enforcement for attack mitigation. The SDN controller collects flow statistics from sensor nodes, analyzes traffic patterns, and enforces security policies through dynamic flow rule installation. We implemented the proposed architecture in OMNeT++, and preliminary results show that the framework can detect abnormal flows and mitigate them by blocking malicious nodes.

Keywords: Wireless Sensor Network · Network Security · SDN

1 Introduction

Wireless Sensor Networks (WSNs) are widely used in various IoT applications; however, their performance and security are often limited due to resource constraints. Traditional networks with distributed routing approaches struggle to adapt to changes in traffic dynamics and to enforce security policies effectively. Software-Defined Networking (SDN) provides a solution by separating the control and data planes, allowing for centralized traffic management and consistent security enforcement. An SDN controller can gather traffic statistics, analyze traffic patterns, and establish flow rules to implement security policies. This work addresses these challenges by integrating traffic analysis and security policy enforcement into an SDN-enabled WSN, with evaluations conducted through simulations on OMNeT++/INET framework.

Numerous applications of SDN in WSNs have been extensively studied in recent years due to its potential to address the limitations of traditional WSNs. In [1], authors provide a comprehensive survey highlighting how SDN programmability and centralized control enhance the scalability and flexibility of WSNs. At the same time, security remains a pressing concern, as mentioned in [2] and

[8], which surveyed threats and mitigation techniques across SDN control and data planes, including emerging AI-driven approaches. [4] evaluated controller architectures that directly influence scalability and reliability. [5] examined centralized and distributed intrusion detection approaches, showing the trade-offs between detection accuracy and overhead. In parallel, traffic analysis and classification have emerged as key tools for anomaly detection and adaptive policy enforcement. Despite these advances, the integration of traffic-aware analysis and dynamic security policy enforcement in SDN-enabled WSNs remains underexplored.

2 Proposed Approach

Based on a classical SDN implementation, we propose a network architecture as Fig. 1 shows. In this architecture, there are two network applications utilised: *traffic analyzer* and *security enforcement mechanism*. Both interact with the SDN controller, which generates flow instructions for connected switches. The controller collects flow statistics such as packet count, byte count, and flow duration from sensor nodes. These statistics are reported to the traffic analyzer to identify anomalous traffic. Network anomaly detection involves identifying devia-

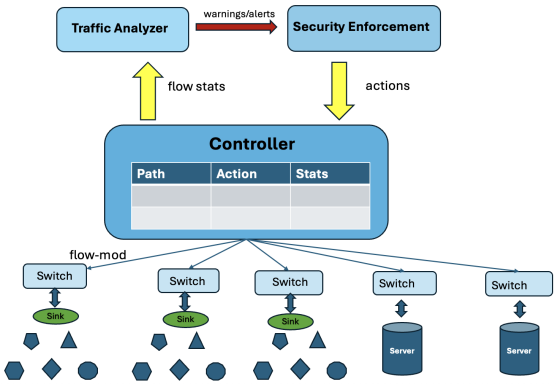

Fig. 1. An SDN-based WSN framework

tions from a network´s usual patterns of activity, which could indicate something malicious is happening. When an anomaly is detected, the traffic analyser will send out a warning or alert depending on the severity of the issue. This triggers the security enforcement mechanism to apply the appropriate security policies. Basic security countermeasures may include actions such as dropping, rerouting, or rate-limiting malicious flows. For critical traffic, additional measures such as authentication and encryption may be necessary.

In terms of sensor networks, sensors are typically power-constrained devices with limited capacities. In most applications, their primary functions are to collect data and send it to sink nodes at specified intervals. For stationary sensors, the traffic path is relatively straightforward, although a secondary path may be required for redundancy. For mobile sensors, dynamically optimizing their routes is crucial.

In typical use cases, sink nodes are integrated into a gateway router that forwards data from wireless networks to wired networks, such as from 802.15.4 to Ethernet. The diagram does not specify how to forward sensor data after it reaches the sink nodes, as this process can vary based on the specific applications of WSN. For instance, sensor data may be collected and sent to different servers for further processing. In many real-world scenarios, these data packets are forwarded to cloud storage or similar platforms.

3 System Model and Simulation Setup

We consider a system consisting of multiple static sensor nodes, denoted as $\mathcal{N}$, that report data to a single sink or gateway. This gateway is connected to an SDN switch and an SDN controller. The controller receives flow statistics and installs mitigation rules to enhance network security. To model potential threats to the system, we assume that an adversary can compromise a subset $\mathcal{A} \subset \mathcal{N}$ of these nodes, launching volumetric traffic attacks such as UDP floods or SYN floods targeting the sink/gateway. The adversary's goal is to decrease the packet delivery ratio, induce congestion, and drain energy from the network.

In this study, we utilized OMNeT++ 6.1 with INET 4.5 for our simulations. We established three separate networks, each containing 50 sensor nodes, for the initial phase of our experiments. By default, the sensors forward their data to a designated sink node, which also functions as a gateway router connecting to the SDN switch. This SDN switch, in turn, connects to the controller, which issues instructions for forwarding data packets to the servers at specified intervals.

At the sink, we deploy a lightweight traffic monitor that processes packets in fixed windows of length $\mathcal{W}$ seconds. For each packet, the monitor extracts the tuple (src node, des service, protocol) and within each window t, aggregates the packet rate r_t (pkts/s), including protocol-specific counts (e.g., ICMP). To track normal behaviour, we maintain a **robust baseline** using an exponentially weighted moving average (EWMA):

$$\mu_t = \alpha r_t + (1 - \alpha)\mu_{t-1} \tag{1}$$

where $\alpha \in (0, 1)$ controls responsiveness. An alarm is raised if K consecutive windows exceed a preset threshold, with special attention to unknown or unregistered sources. This persistence rule reduces spurious alerts from transient bursts while reliably flagging sustained anomalies.

We generated regular sensing traffic, with each sensor transmitting an application data payload of 56 bytes every 5 seconds. To emulate a denial-of-service (DoS) attack, we introduce malicious nodes that send small-payload packets

(8–16 bytes) at a high rate (e.g., 50 pkt/s) toward the sink. We evaluated the performance of the network based on the packet delivery ratio (PDR) and detection accuracy. When security policies (rate-limiting and selective drop) are enabled, malicious traffic is suppressed and PDR increases, at the cost of a modest rise in controller overhead for rule installs and control messages per second.

4 Discussion and future work

In this work, we proposed an SDN-based WSN framework that integrates traffic analysis and security enforcement. A proof-of-concept simulation performed in OMNeT++/INET demonstrates the feasibility of this approach, and preliminary evidence suggests it improves reliability and resilience against malicious activities. While we plan to conduct several additional simulation scenarios for further investigation, we would like to discuss some current issues.

Defining "unusual" traffic for the traffic analyzer can be both straightforward and challenging. In the current simulation, we only monitor sudden spikes in traffic as indicative of DoS attacks. In real-world use cases, there are various other phenomena that may be suspicious, such as unexpected protocols being used, sensor nodes sending more data payloads than usual, or traffic occurring outside of normal operating hours, etc. To gain a better understanding of abnormal network traffic, incorporating Machine Learning models could be a more effective solution for the traffic analyzer to detect complex network activities.

In our simulation, we found that the security enforcement mechanism effectively stabilized energy consumption by reducing retransmissions. Currently, we only block suspicious flows, but we believe additional operations—such as encryption and authentication—should also be implemented. In a WSN, authenticating valid devices is crucial to prevent both insider and outsider threats. However, due to the power constraints of wireless sensor devices, authentication and encryption often consume too much power to be implemented. Our SDN-based WSN framework addresses this issue by offloading the security workload to the Switch/Sink, thereby reducing energy consumption for each individual sensor.

Beyond security, improving the reliability of network communication raises interesting considerations regarding the implementation of network coding. For example, using Random Linear Network Coding (RLNC) could be beneficial for this framework. However, this comes with trade-offs between power consumption (which may lead to longer latency, too) and reliability. The decision often depends on the type of applications or use cases, specifically how critical the data is and how time-sensitive it may be.

This proposed framework is designed to be extendable to hybrid networks that incorporate both wireless and wired devices. Integrating it with other systems, including various IoT systems, should be relatively straightforward. In the case of a comprehensive system structure that includes edge and fog nodes, integration should be feasible because a Switch/Sink node can serve as an edge or fog node with minimal modifications. However, one concern is that the position

of the controller creates a single point of failure within the network. Therefore, it is essential to optimize the controller's placement to mitigate this vulnerability and to improve scalability as more devices connect to the system.

References

1. Abhishek Narwaria, Arka Prokash Mazumdar. "Software-Defined Wireless Sensor Network: A Comprehensive Survey". Journal of Network and Computer Applications, Volume 215 (2023), https://doi.org/10.1016/j.jnca.2023.103636
2. Maleh, Y., Qasmaoui, Y., El Gholami, K. et al. "A comprehensive survey on SDN security: threats, mitigations, and future directions". J Reliable Intell Environ 9, pp. 201–239 (2023). https://doi.org/10.1007/s40860-022-00171-8
3. Keshari, S.K., Kansal, V. & Kumar, S. "A Systematic Review of Quality of Services (QoS) in Software Defined Networking (SDN)". Wireless Pers Commun 116, pp. 2593–2614 (2021). https://doi.org/10.1007/s11277-020-07812-2
4. Zhu, Liehuang & Karim, Md M. & Sharif, Kashif. & Xu, Chang. & Li, Fan. & Du, Xiaojiang. & Guizani, Mohsen. "SDN Controllers: A Comprehensive Analysis and Performance Evaluation Study". ACM Comput.Surv., vol. 53. Association for Computing Machinery (2021). https://doi.org/10.1145/3421764
5. G. A. N. Segura, A. Chorti and C. B. Marg. "Centralized and Distributed Intrusion Detection for Resource-Constrained Wireless SDN Networks". IEEE Internet of Things Journal, vol. 9, no. 10, pp. 7746-7758 (2022). https://doi.org/10.1109/JIOT.2021.3114270
6. Ahmad Azab, Mahmoud Khasawneh, Saed Alrabaee, Kim-Kwang Raymond Choo, Maysa Sarsour. "Network traffic classification: Techniques, datasets, and challenges". Digital Communications and Networks, Volume 10, Issue 3, pp.676-692 (2024). https://doi.org/10.1016/j.dcan.2022.09.009
7. R. Kumar, M. Swarnkar, G. Singal and N. Kumar. "IoT Network Traffic Classification Using Machine Learning Algorithms: An Experimental Analysis," in IEEE Internet of Things Journal, vol. 9, no. 2, pp. 989-1008 (2022). https://doi.org/10.1109/JIOT.2021.3121517
8. A. H. Abdi et al. "Security Control and Data Planes of SDN: A Comprehensive Review of Traditional, AI, and MTD Approaches to Security Solutions," in IEEE Access, vol. 12, pp. 69941-69980 (2024). https://doi.org/10.1109/ACCESS.2024.3393548
9. INET Developer´s Guide, https://inet.omnetpp.org/docs/developers-guide/index.html, last accessed 2025/09/03