Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.058963

ARTICLE

Machine Learning-Based Detection and Selective Mitigation of Denial-of-Service Attacks in Wireless Sensor Networks

Soyoung Joo[#], So-Hyun Park[#], Hye-Yeon Shim, Ye-Sol Oh and Il-Gu Lee[*]

Department of Future Convergence Technology Engineering, Sungshin Women's University, Seoul, 02844, Republic of Korea

*Corresponding Author: Il-Gu Lee. Email: iglee@sungshin.ac.kr

#Co-first authors

Received: 25 September 2024; Accepted: 13 December 2024; Published: 17 February 2025

ABSTRACT: As the density of wireless networks increases globally, the vulnerability of overlapped dense wireless communications to interference by hidden nodes and denial-of-service (DoS) attacks is becoming more apparent. There exists a gap in research on the detection and response to attacks on Medium Access Control (MAC) mechanisms themselves, which would lead to service outages between nodes. Classifying exploitation and deceptive jamming attacks on control mechanisms is particularly challengingdue to their resemblance to normal heavy communication patterns. Accordingly, this paper proposes a machine learning-based selective attack mitigation model that detects DoS attacks on wireless networks by monitoring packet log data. Based on the type of detected attack, it implements effective corresponding mitigation techniques to restore performance to nodes whose availability has been compromised. Experimental results reveal that the accuracy of the proposed model is 14% higher than that of a baseline anomaly detection model. Further, the appropriate mitigation techniques selected by the proposed system based on the attack type improve the average throughput by more than 440% compared to the case without a response.

KEYWORDS: Distributed coordinated function mechanism; jamming attack; machine learning-based attack detection; selective attack mitigation model; selective attack mitigation model; selfish attack

1 Introduction

As wireless communication is a key technological enabler in nearly all domains, it is vital that its components meet the required performance and security demands. Internet of Things (IoT) and wireless sensor networks (WSNs) are examples of such components, which support various operations using networks of heterogeneous industrial devices connected wirelessly [1–3]. IoT networks involving sensor nodes typically operate on battery power, making energy-efficient, low-power communication protocols crucial to meet low-power consumption requirements. In WSNs, protocols such as ZigBee or Institute of Electrical and Electronics Engineers (IEEE) 802.15.4 are commonly used to this end. Additionally, the IEEE 802.11 ah (Wi-Fi HaLow) standard is a Wireless Local Area Network (WLAN) standard designed for long-range communication operating in the sub-1 GHz unlicensed band, excluding the TV white space band, which is used for long-range communication in IoT applications.

IEEE 802.11 standard defines communication protocols for WLAN. Due to its backward compatibility and ongoing development, WLAN technology demonstrates excellent scalability, making it a widely adopted communication technology for wireless devices not only in WSNs but also in IoT networks. In next-generation WLAN standards, reliability has gained increasing significance, becoming as important as

communication efficiency. To ensure WLAN reliability, both communication performance and security during the communication process must be considered. Since communication reliability directly affects performance, WLAN security must be addressed to achieve optimal performance and efficiency.

Conventional WLAN technologies have primarily focused on enhancing communication performance and efficiency. IEEE 802.11n standard was developed to achieve high throughput by leveraging multiple-input and multiple-output (MIMO) antenna technology. It can accommodate up to four antennas at both the transmitter and receiver, utilizing up to four spatial streams while supporting beamforming. The IEEE 802.11ac standard increases transmission efficiency by expanding the bandwidth to 160 MHz and improving overall system throughput through downlink multi-user MIMO (MU-MIMO). To address efficiency challenges in dense networks, the IEEE 802.11ax standard introduces orthogonal frequency division multiple access and uplink MU-MIMO, ensuring more efficient use of frequency resources. Most recently, IEEE 802.11be standard incorporates multi-link operation, allowing simultaneous communication over multiple channels, further improving communication efficiency.

Although WLAN security technologies have been extensively researched [4], traditional WLAN designs have focused more on maximizing performance and efficiency than on enhancing security, leaving them vulnerable to potential attacks. This is a critical shortcoming, esecially because WLAN security extends beyond direct techniques like data encryption to include defenses against attacks that exploit system blind spots. For instance, Medium Access Control (MAC) layer header is transmitted without encryption, making it highly vulnerable to attacks that tamper with unprotected data. An attacker can interfere with normal communication by altering critical information in the MAC frame header, such as the STA address, packet number, and duration, to trigger malicious behavior.

Message traffic suffers performance degradation due to interference, collisions, hidden nodes, and deceptive attacks [5–8]. To address this, the IEEE 802.11 MAC protocol specifies virtual and physical carrier sensing mechanisms that effectively avoid or mitigate collisions between wireless nodes while competing for channel access. The Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), an access control technique implemented in the MAC layer, prevents frame collisions based on virtual carrier sensing. To minimize collisions caused by simultaneous data transmissions, stations (STAs) first perform physical carrier sensing to verify the availability of the wireless channel before initiating data transmission. After waiting for a random backoff period, STAs transmit data transmission on idle channels. If the channel is busy, the random waiting time is increased exponentially. This waiting time implemented to avoid collisions is determined by the contention window (CW), which ranges between the minimum (CWmin) and maximum (CWmax) values. STAs with smaller CW values gain access to the medium more quickly. Since CW values are randomly assigned at each instance, fair competition among all STAs attempting to access the medium is maintained in a normal network.

However, channel access mechanisms are susceptible to exploitation attacks that can disrupt nodes' ability to communicate with access points (APs) [9]. For example, denial-of-service (DoS) attacks overwhelm target networks with excessive traffic, preventing nodes from accessing the communication medium. DoS attacks typically fall into two categories: **selfish attacks**, where attackers manipulate the backoff counter value to monopolize network resources, and **jamming attacks**, where attackers intentionally transmit continuous noise to interfere with specific devices. These attack types are explored in further detail below.

In a selfish attack, attackers "selfishly" occupy communication channels with an AP in a WLAN. Such attacks are carried out by manipulating the CW value of a STA. Generally, CWmin is set to 15 (except in the 802.11b standard) and CWmax to 1023. Selfish nodes are attackers that exploit the fair access mechanism by consistently gaining priority access to the medium. All parameters of selfish nodes are configured identically to those of normal nodes, except their CWmin and CWmax values are reduced to ensure lower backoff

counter values. This allows them to gain unfair access to the communication medium. As a result, selfish attacks degrade network performance by lowering overall data transmission throughput and reducing network availability by blocking other nodes from acccessing the medium. While the selfish node's individual throughput increases, the overall network throughput suffers significantly.

On the other hand, jamming attacks represent another major threat that compromises the availability of network systems. In fields such as military security, jammers are used to block data communication with external networks and unauthorized users. However, attackers can misuse jammers to launch DoS attacks, disrupting communication signals between legitimate users. Since jamming signals interfere with transmissions from legitimate senders, the signal-to-interference-plus-noise ratio (SINR) deteriorates significantly under such attacks, preventing receivers from correctly decoding transmitted messages.

In energy-constrained environments like WSN and IoT-based networks, DoS attacks pose an even greater threat. If data transmission fails in WLAN, the automatic repeat request mechanism retransmits the data. As a result, persistant DoS attacks not only degrade throughput significantly compared to normal operating conditions but also consume a large amount of the device's energy resources, leading to higher communication costs. Moreover, DoS attacks can escalate into battery depletion attacks, deliberately draining the batteries of sensor and IoT devices that rely solely on battery power. IoT devices are particularly vulnerable not only to jamming attacks that disrupt availability but also to battery depletion attacks that can cause complete system shutdowns. Therefore, detecting jamming attacks is a crucial priority for IoT and WSN environments.

In normal network congestion scenarios, issues such as reduced throughput and availability of STAs can occur naturally, making it challenging to distinguish between regular network congestion and malicious DoS attacks. This is especially true for selfish attacks because nodes' CW values are set randomly under normal operation. Similarly, distinguishing between heavy interference caused by environmental factors and intentional jamming attacks can be equally difficult.

Machine learning models offer promising attack detection solutions for detecting such attacks without requiring complex or resource-intensive modifications to the MAC protocol [10]. They have been effectively applied to detect critical attacks, including DoS attacks [11–14]. With numerous studies exploring their use in IoT and WSN environments [15–18]. In particular, DoS attacks can be identified by analyzing their side effects, such as reduced throughput across multiple nodes, increased delays, and altered signal reception patterns within the network. The behavior of selfish and jamming attacks increases the energy consumption of APs by raising service demands, ultimately blocking multiple connected nodes from communicating.

In this paper, we propose a model for detecting selfish and jamming attacks in IEEE 802.11-based WSNs operating in diverse, overlapping, and large-scale environments. The model addresses these attacks using approproate selective response mitigation techniques. To achieve this, a simplified basic service set environment is considered for each attack, where the machine learning model detects malicious behavior by analyzing communication occupancy frequencies, packet durations, and CW values. This approach enables the application of effective defense mechanisms against detected attacks. Specifically, conflicting numbers of control packets and average throughput are compared to determine the optimal backoff counter value for selfish attack responses. The main contributions of this paper are summarized below:

- **Modeling of Selfish and Jamming Attacks:** Selfish and jamming attacks are modeled in IEEE 802.11-based wireless communication system environments.
- **Attack Pattern Analysis:** Attack patterns are distinguished by analyzing the effects of the attacks and the available packet trace log data.
- **Machine Learning-Based Attack Detection:** The proposed machine learning attack detection method classifies attacks based on the duration of packet communication, the number of communication

repetitions with the AP, and the status of the node's CW value (manipulated *vs.* not-manipulated). Its performance is validated by comparing its attack detection rate with that of a state-of-the-art anomaly detection model.

- **Selective Attack Mitigation Model (SAMM):** A novel selective attack mitigation model (SAMM) is proposed that applies appropriate countermeasures based on detected attacks.

The remainder of this paper is organized as follows. In Section 2, related works on selfish and jamming attacks are reviwed. In Section 3, an overview of the IEEE 802.11 protocol is presented. The simulation environment and attack models are described in Section 4, and SAMM and extant DoS attack detection methods are described in Section 5. In Section 6, the proposed model's performance is validated and its efficiency is assessed. Finally, the paper is concluded in Section 8.

2 Related Works

In WLANs, the virtual carrier sense mechanism predicts the state of a channel at any time using a network allocation vector (NAV) by analyzing the duration of the previous frame. Notably, the NAV mechanism is vulnerable to false blocking, virtual jamming, and ready-to-send (RTS)/clear-to-send (CTS) attacks [19]. On the other hand, the physical carrier sense (i.e., clear channel assessment (CCA)) mechanism monitors busy or idle states of channels objectively and continuously and transmits the information to the wireless network's MAC sublayer. This process is vulnerable to DoS attacks that interfere with the availability of other nodes and prevent legitimate users from accessing the channel. This section presents an overview of the most relevant studies on deceptive selfish and jamming DoS attacks and appropriate response methods. There is a variety of research focused on detecting and responding to selfish attacks and jamming attacks individually, but few studies distinguish between these two attacks to detect and respond selectively.

2.1 Selfish Attacks

The distributed coordination function (DCF) mechanism, which is a CSMA/CA medium access protocol for WLANs, is vulnerable to selfish backoff attacks [20]. This is because selfish devices can set their backoff timers to very small thresholds, allowing them to access channels more frequently than other devices [21]. The CW size determines the range of the random backoff counter for all devices. Normally, minimum and maximum CW values are fixed within the standard. However, a selfish device can manipulate these values to override its priority. Selfish behaviors can include partial dropping, false accusation, packet dropping, and insufficient transmission power effects [22].

Several studies have analyzed selfish attacks, and their countermeasures are largely governed by game theoretic considerations related to managing routing paths, energy usage, and node confidence values. Konorski et al. proposed a game theory-based approach to handle greedy and honest nodes, enabling the latter to overcome their throughput disadvantages, especially with increasing number of nodes. Their simulation results demonstrated that the throughput of greedy nodes gradually decreased because small manipulations of the CW parameter were equally effective for disruptions and remedies [20].

Fihri et al. proposed a support vector machine-based nonlinear classifier to detect backoff manipulation attacks. The model exhibited the shortest execution times among machine learning classifiers, especially when supplied with radial basis function kernel classifiers. Moreover, it exhibited the lowest computational complexity [23]. Kim et al. proposed a method for detecting selfish attacks by mathematically analyzing selfish backoff attacks using logistic classifiers [21]. Malicious nodes typically set the CW value to 1 or 2—their study proved that when it is set to 1, the attacker can immediately access the channel without waiting for a backoff; however, when it is set to 2, other devices can access the channel first. Chakraborty et al. used backoff properties to address collisions during simultaneous transmissions; however, their algorithm did

not work properly with large networks [24]. Nonetheless, their results demonstrated that a random-access game theoretical protocol exhibited higher average throughput and lower access delay compared to DCF in the case of WLAN. Odedra et al. combined threshold-based detection methods with watchdog surveillance techniques [23] to improve network performance and reduce the impact of selfish nodes. However, the proposed method was unable to detect cooperating selfish nodes and exhibited limited detection of the incapability of nodes to reengage in routing after isolation.

2.2 Jamming Attack

Jamming attacks are commonplace DoS attacks that cause intentional interference to stifle network communication [25]. Jammers are generally categorized as proactive or reactive, and the proactive type can be further divided into constant, deciphered, and random types [26].

Various studies have suggested methods for detecting and responding to deceptive jamming attacks, but few have suggested an integrated response system that distinguishes between selfish and jamming attack types. Vadlamani et al. conducted a survey and concluded that although a deceptive jammer is similar to a constant jammer, the former transmits a legitimate initial bit sequence to impersonate a legitimate node [27]. Deceptive versions also implement defense strategies against transmission power adjustment, frequency-hopping spread spectrum, channel switching, and directional antenna defenses.

Kanwar et al. proposed the JamSense model, which classifies and detects interference and jamming attacks in WLANs [28]. Their study distinguished between interference, constant jammers, and deceptive jammers. The authors classified the preamble and start-of-the-frame delimiters of the constant and deceptive jammers' packets in terms of their transmission status. As such, attacks could be identified with up to 96% accuracy. Despite the excellent findings, the impact of these network attacks has not been analyzed further.

2.3 Threat Detection and Response

Recent advancements in wireless network security have explored various mitigation frameworks aimed at enhancing system reliability and defense against adversarial threats. Liu et al. proposed the RFL-APIA framework, addressing federated learning vulnerabilities by identifying and mitigating model poisoning attacks through robust aggregation mechanisms, which parallels the proposed selective attack mitigation model's emphasis on adaptive response strategies in WSNs [29]. Bai et al. introduced a Throughput Maximization Model for secure multipath transmissions in wireless *ad-hoc* networks, focusing on optimizing network throughput while protecting from potential eavesdroppers [30]. Gong et al. explored Computation and Privacy Protection for Satellite-Ground Digital Twin Networks, emphasizing secure data mapping and resource optimization, an idea relevant to dynamic resource allocation in mitigating DoS attacks [31]. These works collectively underscore the necessity of adaptive, intelligent frameworks for real-time network threat detection and response, forming the conceptual basis for the proposed model's selective attack mitigation strategy.

3 Key Features of IEEE 802.11

3.1 DCF

Fig. 1 presents typical infrastructure for WLAN, where wireless devices communicate with the AP within a coverage region. The underlying MAC mechanism of the IEEE 802.11 WLAN standard [32] is DCF, which uses competition-based algorithms to provide access to shared media [33]. As depicted in Fig. 2, DCF adjusts channel access using the binary exponential backoff (BEB) algorithm [34], which prevents repeated retransmissions of the same packet to reduce network traffic [32]. If the sender detects an idle channel, it

waits in the distributed interframe space (DIFS) and transmits its frame. If the channel is detected to be busy, the DIFS time is added to the backoff counter, and the packet is transmitted once the counter reaches zero. The backoff counter is an arbitrary time determined by the BEB and depends on the CW value. If the frame is successfully transmitted, the sender resets the CW value to the minimum value. If the frames collide, the CW is doubled to the maximum value, and a random backoff counter value is selected in [0, CW-1]. When the channel is found to be inactive, its value decreases. When the channel is used, the process is reactivated when the channel is inactive [35].

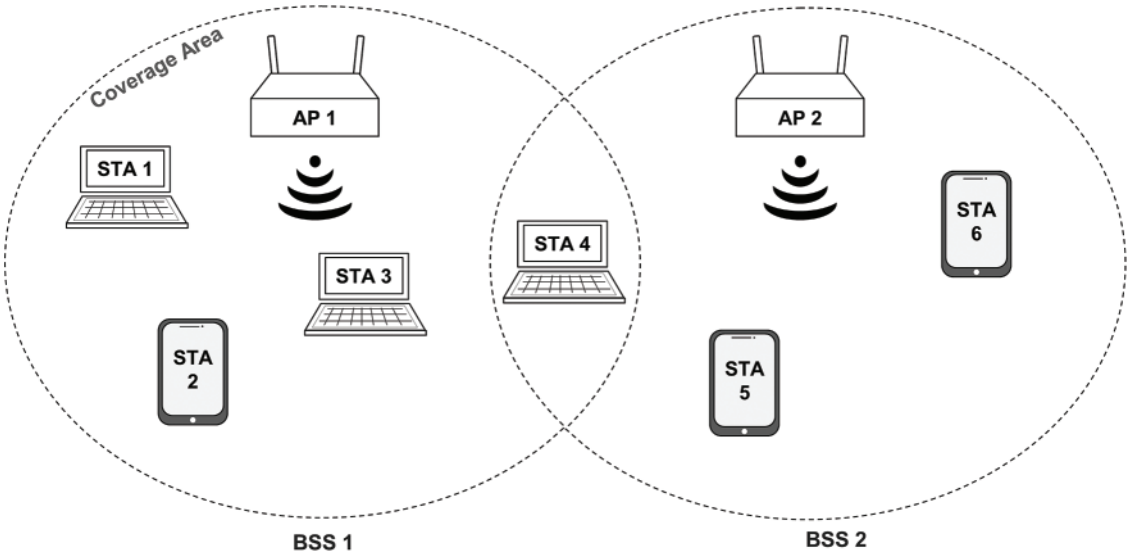

Figure 1: Infrastructure of a WLAN system

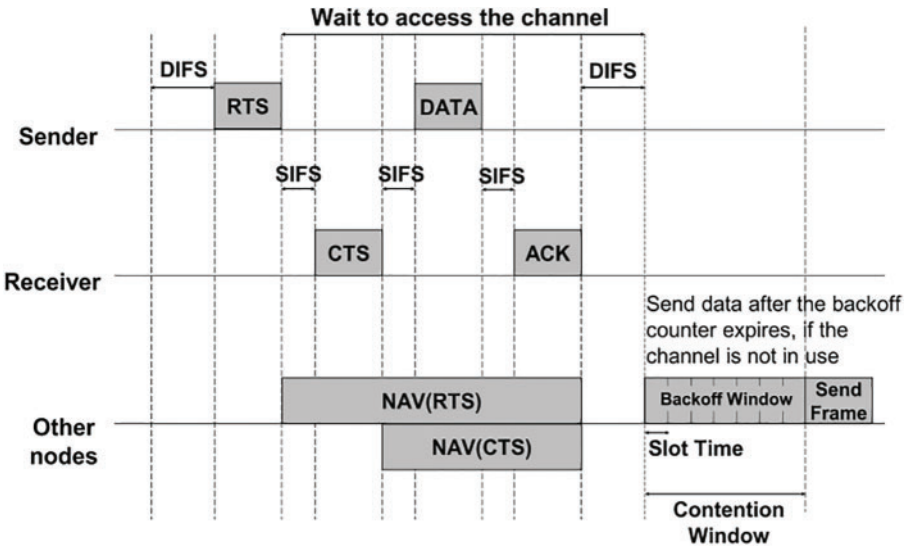

Figure 2: DCF mechanism

DCF uses two modes of transport—basic (two-handshake) and RTS/CTS (four-handshake). In the basic mode, the sender detects whether the channel is busy and transmits a data frame if idle. Once the transmitted frame is received successfully, the receiver responds with an acknowledgement frame. The RTS/CTS access mode reserves a channel before data transmission. After receiving the frame, the other nodes update their NAV values based on the field duration of the RTS/CTS value of the reserved frame. The other nodes transmit data frames through the DCF when the NAV value reaches zero.

3.2 Multi-Link Operation

The IEEE 802.11be standard is expected to be an extremely high-throughput amendment to improve the reliability of wireless communications and increase the maximum throughput by 30 Gbps while reducing latency [36]. Multilink operations comprise the core technology needed to achieve these high expectations. The multilink framework is advantageous because it exhibits multifrequency bands and low hardware costs, such that APs and STAs can simultaneously transmit and receive information on different links using multiple radio interfaces depending on the transmission mode [37].

Multilink transmissions generally include synchronous and asynchronous methods categorized in terms of their simultaneous uplink/downlink (UL/DL) transmission capabilities. In both types, asynchronous transmissions and receptions on one or more links are allowed, and transmission on one link and reception on the other can be supported simultaneously. Additionally, DL frames that fail on one link can be retransmitted to other available links to reduce latency, and some traffic can be switched to other low-load links to improve the quality of service on overloaded links. During asynchronous operations, each link of the multilink device executes its channel process separately. Consequently, each link can achieve an independent maximum favorable throughput. In contrast, in synchronous operations, all links must wait for the idle state to begin transmission. Multiband and multichannel operations are discussed below in conjunction with multilink transmissions designed to improve performance despite heavy interference [35].

4 Simulation Environment

NetSim v.13.1, a commercial IEEE 802.11-based packet-level network simulator, is used to simulate SAMM in this study. NetSim visualizes the WLAN packet flow, and a trace log is produced that reports the packet arrival time, queuing time, type, payload, overhead, status, and source. Before simulating the integrated DoS attack environment, a simplified WLAN environment containing one AP and 3–10 connected STAs is modeled, as shown in Table 1.

The environment is assumed to exhibit limited frame aggregation and rate adaptation. Hence, all interface parameters are selected based on the 802.11n standard, and the RTS_Threshold is set to 800 bytes to activate the RTS/CTS mechanism. Wireless nodes can generate a constant bit rate and file transfer protocol services using either the transmission control protocol or the user datagram protocol. The traffic generation rate is obtained as follows:

$$CBRGenerationRate = \frac{PacketSize(bytes) \times 8}{ArrivalTime(microsec)}, \tag{1}$$

$$FTPGenerationRate = \frac{FileSize(bytes) \times 8}{ArrivalTime(microsec)}. \tag{2}$$

Maximum packet and file sizes are set to 1460 bytes, and the inter-arrival time is unified at 11.6 μs. As the link and simulation parameters determine the wireless channel conditions and simulation time, models combining free-space path loss, shadowing, and Rayleigh fading are used for the simulation.

Table 1: Simulation parameters

Components	Parameters	Values
Interface parameters	Standard	IEEE 802.11n
	Number of packets aggregated	1
	Channel	36 (5180 MHz)
	Rate adaptation	FALSE
	Short retry limit	7
	Long retry limit	4
	Dot11_RTS Threshold	800 bytes
	Buffer size	1 MB
	Guard interval	400 ns
	Bandwidth	20/40 MHz
	Frequency band	2.4/5 GHz
	Transmitter power	100 mW
	Antenna gain	0
	Antenna height	1 m
	Medium access protocol	DCF
	SlotTime	9 us
	SIFS	16 us
	CS Min/Max	15/1023
Application parameters	Application	CBR
	Packet size	1460 bytes
	Inter-arrival time	11.6 μs
Link parameters	PathLoss model	Friis free space
	Channel characteristics	PathLoss and fading and shadowing
	Fading model	Rayleigh
Simulation parameters	Simulation time	10000 ms

The network performance is illustrated in Fig. 3 in terms of the application and average link throughput. Link throughput considers all traffic passing through a link, including data and control packets, retransmissions, errors, and collisions. On the other hand, application throughput only considers the data packets successfully received at the destination from the source. Link throughput is calculated using Eq. (3), noting that an application throughput can be measured for each application. As depicted in Fig. 3, the maximum and minimum throughputs decrease as the number of STAs is increased. However, the link throughput is maintained at a constant value.

$$LinkThroughput(Mbps) = \frac{Totalbytestransmittedinthelink \times 8}{TransmissionTime(microsec)}. \tag{3}$$

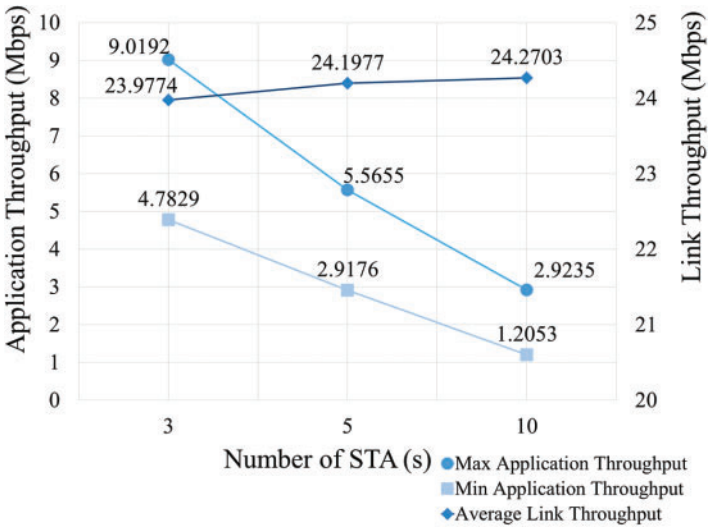

Figure 3: Network performance of the WLAN simulation model

5 Attack Models

5.1 Selfish Attack Model

Fig. 4 illustrates network performance as a function of the number of STAs during a selfish attack. The network's maximum application throughput is exceptionally high compared to a normal WLAN model because the maximum throughput is measured from the selfish node. Hence, the throughput of other nodes is diminished to values well below average minimum values. This increases the total bytes transmitted by the selfish node. Consequently, the average minimum application throughput of the other nodes is reduced from 2.96 to 0.28 Mbps, representing a 90.54% degradation in performance.

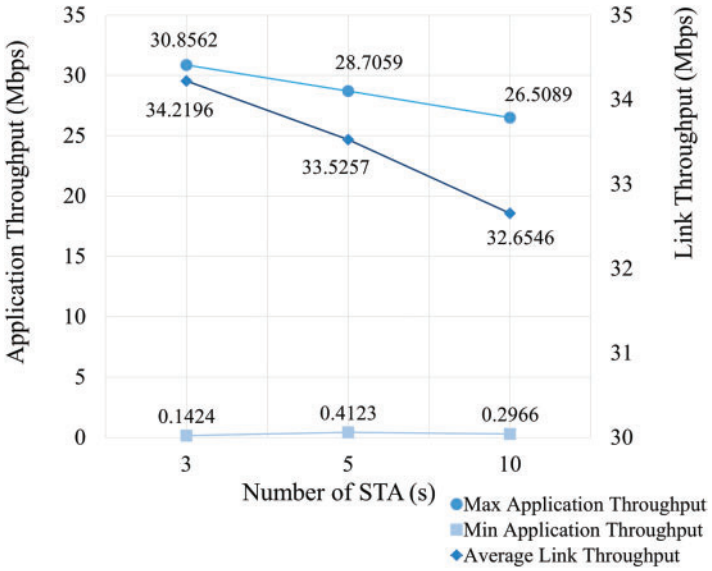

Figure 4: Network performance of the selfish attack model

5.2 Jamming Attack Model

A deceptive jammer is a type of radio frequency interference device that continuously transmits signals, similar to a constant jammer, but with a more sophisticated approach. Unlike constant jammers, which emit random noise or arbitrary bit sequences, deceptive jammers mimic legitimate communication signals. They transmit seemingly valid data packets or bit patterns that resemble those generated by legitimate devices in the network. Therefore, deceptive jamming attacks remain undetected for longer periods. Since their transmissions appear legitimate, distinguishing between real and fake data becomes challenging for the network. This tactic allows for prolonged disruption without immediate detection, making deceptive jammers more insidious and effective than constant jammers in denial-of-service (DoS) attacks. In addition, jammers adjust their signal strength to fine-tune their effects and avoid detection. Fig. 5 illustrates network performance as a function of the number of STAs during a jamming attack. Table 2 summarizes the simulation parameters, where the interaction time is used to determine the packet generation rate.

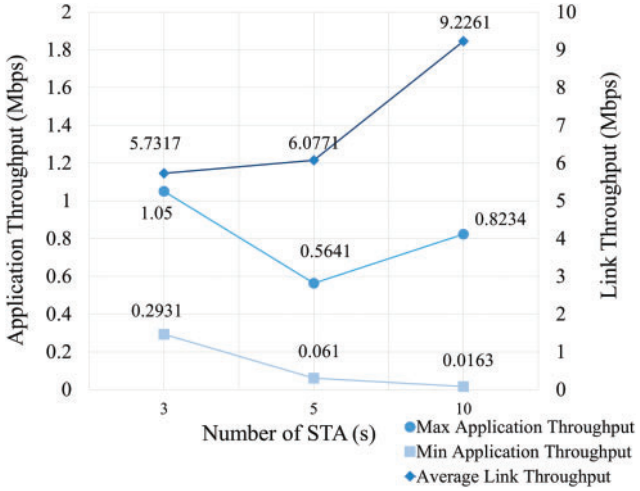

Figure 5: Network performance of jamming attack model

Table 2: Parameters for deceptive jammers

Components	Parameters	Values
Interface parameters	Standard	IEEE 802.11n
	Number of packet aggregated	1
	Channel	36 (5180 MHz)
	Rate adaptation	FALSE
	Short retry limit	7
	Long retry limit	4
	Dot11_RTS threshold	3000 bytes
	Buffer size	1 MB
	Guard interval	400 ns
	Bandwidth	20/40 MHz
	Frequency band	2.4/5 GHz

(Continued)

Table 2 (continued)

Components	Parameters	Values
	Transmitter power	100 mW
	Antenna gain	0
	Antenna height	1 m
	Medium access protocol	DCF
	SlotTime	9 μs
	SIFS	16 μs
	CS Min/Max	15/1023
Application parameters	Application	CBR
	Packet size	8 bytes
	Inter-arrival time	5.8 μs

Notably, reducing the inter-arrival time increases this rate. To validate the proposed model, a jammer with a transmission power of 100 mW is positioned randomly, and 8-byte packets are transmitted at a rate calculated to achieve 5.8-μs inter-arrival times, resulting in ~17,241 packets generated per second. Because all nodes in the WLAN are capable of a robust 6.5-Mbps data rate, the STAs' locations do not affect network performance significantly. In our simulation, the jammer follows the same standard as other STAs under normal circumstances. However, it also transmits periodic jamming packets without using the RTS/CTS mechanism. During the simulated attack, the number of packets transmitted to the AP is observed to be ~1000, disrupting its communications with other nodes. The attack is also observed to cause a loss of ~2000 data packets due to collision, whereas none typically crashed during normal WLAN operations. Thus, the jammer reduces the average throughput from 2.96 to 0.12 Mbps.

6 Selective Attack Mitigation Model

6.1 Dataset

Table 3 summarizes the data used in this study based on packet log data collected in an attack-integrated model and a related description.

Table 3: Dataset components

Feature	Description
Queuing_Delay	Queuing time between time of arrival of the packet at PHY/MAC layer
Transmission_Time	Transmission duration between packet transmission in the link and arrival at the PHY layer of the transmitter
Propagation_Delay	Propagation delay time between packet transmission in the link and arrival at the PHY layer of the receiver
Total_Packet_Travelling_Time	Sum of queuing time, transmission time, and propagation delay time
isContinuoslyOccupied	Count if the packet type is a data packet and has the same source node as the previous one

Packet trace log data, including type (data or control), source, destination, layer arrival time, payload size, collision status, queuing time, transmission time, propagation delay, total traveling time, and repetitions of communication occupancy, are simulated or collected from the attack model and compared with WLAN statistics in the absence of an attack.

Queuing delays are simulated by subtracting the time of arrival of the packet at the physical layer from that at the MAC layer [38]. The transmission time is simulated by subtracting the time of arrival of the packet at the physical layer from the time of initial transmission. The propagation delay is simulated by subtracting the time initial transmission time of the packet in the physical layer from the final transmission time in the physical layer. Finally, the total travel time is simulated by summing the queuing, transmission, and propagation delays.

Packets that monopolize communications with the AP and exhibit long durations are assumed to be related to attacks [26,28]. During model training, packet trace logs are analyzed to distinguish between attacks and ordinary packets, and the suspected attacks are classified as jamming or selfish attacks. The data are labeled by dividing the total number of transmitted packets by the number of packets obtained from the suspected jammer, the suspected selfish node, and normal packets.

6.2 Classification Algorithms

Fig. 6 describes the proposed attack detection system pipeline, and Algorithm 1 is used to detect jamming and selfish attacks. Fig. 7 depicts the SAMM mechanism's attack mitigation pipeline. Unintentional continuous occupancy of a single STA with an AP can occur when a DCF mechanism is used with random backoff counters. However, a certain number of constant occupancies can be assumed to be non-coincidental [16]. In the proposed algorithm, the continuous occupancy of a packet that occurs more than thrice is considered to be an attack.

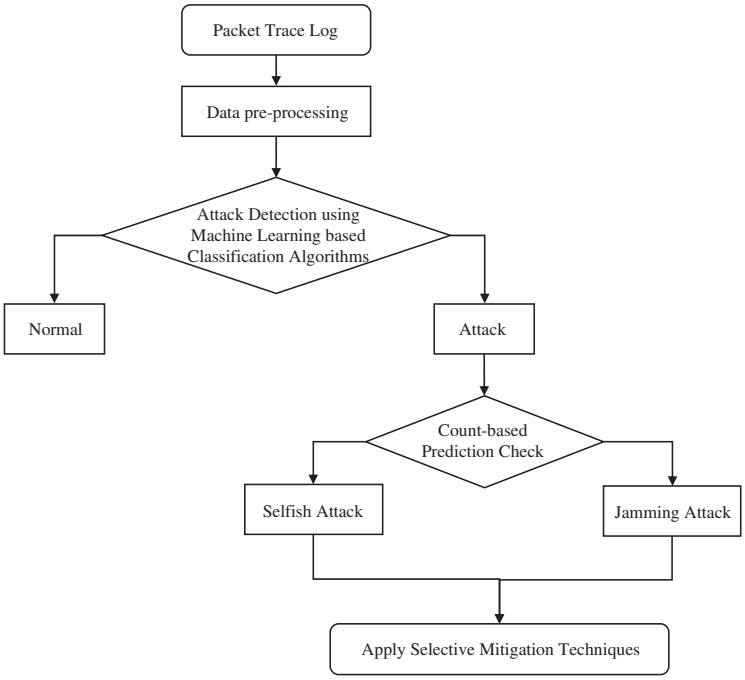

Figure 6: Attack detection pipeline of SAMM

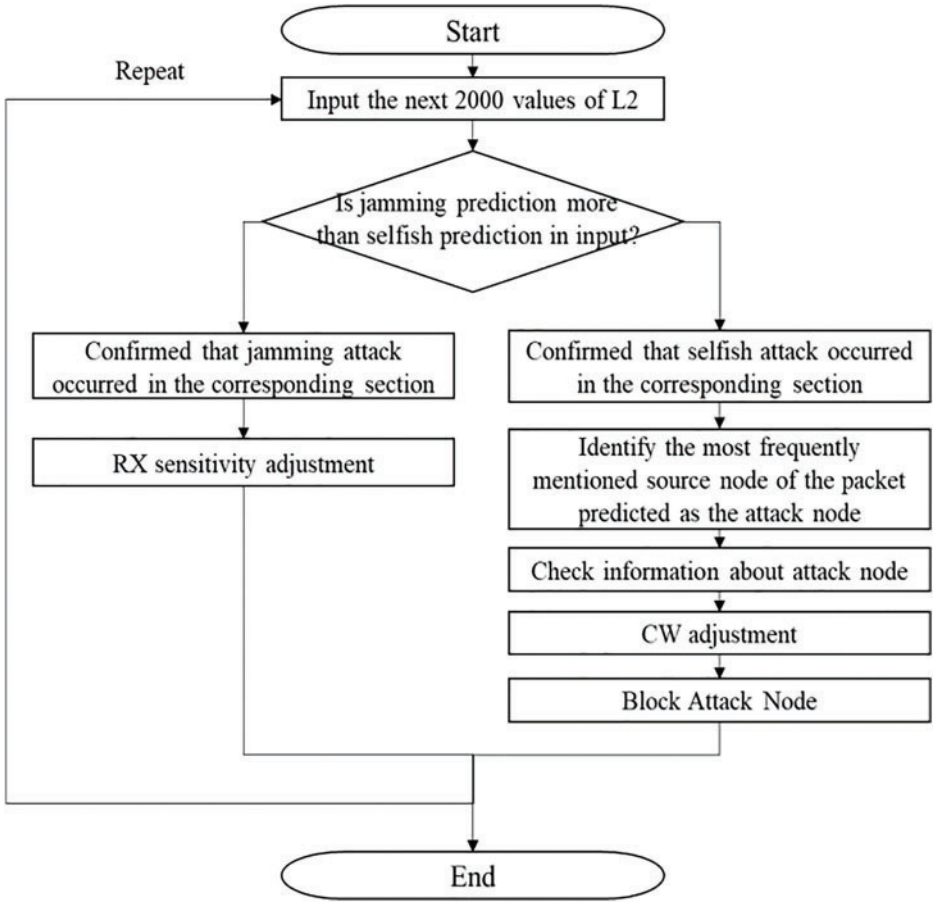

Figure 7: Attack mitigation pipeline of SAMM

Algorithm 1: Detection algorithm for jamming and selfish attacks

```
input packet_trace_log_data PTL
input attack_prediction_model M
output L2
list L1, L2
n = size(PTL)
s1, s2 = size(L1), size(L2)
j, k = 1
for i = 1 to n do
    L1[j] = M(PTL[i]) # "normal", "selfish", "jamming"
    j, i = j + 1, i + 1
    if j > s1 then
        if half of L1 is normal then
            L2[k] = "normal"
        else
            if there are more selfish than jamming instances then
                L2[k] = "selfish"
```

(Continued)

Algorithm 1 (continued)

```
            else
              L2[k] = "jamming"
            end if
          end if
          k = k + 1
          j = 1
          init L1
        end if
    end for
```

The SAMM classification is based on a light gradient boosting machine (lightGBM) decision-tree model [38] with improved functionality, which exhibits high computation speeds with reduced memory consumption. Additionally, a normalization process is included to avoid overfitting. Using the GBDT boosting type, we trained the model with 400 iterative trees, each with a maximum of 31 leaves. The learning rate was set to 0.1, and the maximum depth of the trees was set to −1. LightGBM is a highly efficient gradient-boosting framework that uses tree-based learning algorithms. Apart from other tree-based algorithms, It applies a more complex leaf-wise split approach to prevent overfitting [39]. In experiments comparing machine learning techniques, LightGBM had the fastest prediction time and model training time [40]. Due to deep learning generally requiring large volumes of labeled data to achieve optimal performance, SAMM applied a machine learning-based model requiring less labeled data for effective training.

The classification performance is evaluated in terms of accuracy and a binary confusion matrix that accounts for true-positive (TP), true-negative (TN), false-negative (FN), and false-positive (FP) predictions. Precision, Recall, F, and F1 scores are derived from these reports, where Precision = TP/(TP + FP), Recall = TP/(TP + FN), F score = weighted average of precision and reproduction rates, and F1 score = harmonic mean of precision and reproduction rates.

6.3 Count-Based Prediction

Although, the jamming node in a jamming attack may be unknown, the attacker is always obvious in a selfish attack. Hence, a count-based prediction algorithm is used to predict attacks using classified attack alarms while providing information on the suspected attacker node(s). Because multiclass classification algorithms detect attack types, accurate prediction based on the number of attack alarms is possible. In this paper, all predicted attack alarms are placed in a single section, and source-node ratios are calculated and compared.

6.4 Mitigation Techniques

Following detection and prediction using SAMM, a selective attack mitigation technique is used to respond selectively to the type of attack. In response to selfish attacks, the proposed mitigation system dynamically adjusts contention window (CW) values and backoff counters of non-attacking nodes to immediately reduce network degradation after identifying the attacker. By leveraging game-theoretic principles, the system minimizes control packet collisions during backoff counter reduction, ensuring efficient use of network resources. As illustrated in Fig. 8, smaller CW values significantly enhance the average throughput of non-attacking nodes while increasing control packet collisions. This trade-off demonstrates the system's capability to maintain network performance even under attack conditions. Notably, decreasing the backoff counter values improves throughput for legitimate nodes. However, if the selfish node detects

these adjustments and attempts to lower its backoff counter further, it can exacerbate network competition, causing widespread communication delays. To counter this, the system employs a final mitigation strategy: link switching. By virtually relocating the entire network—excluding the attacker—to a new channel, the system isolates the attacker, preserving the integrity and functionality of the legitimate network.

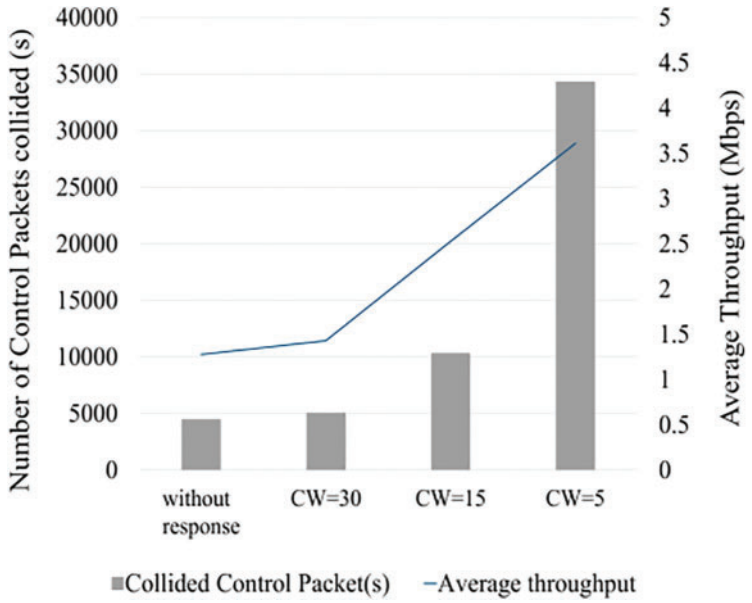

Figure 8: Comparative analysis of colliding control packets and average throughput with respect to CW values

In response to jamming attacks, the proposed mitigation model suggests adjusting the Receiver Sensitivity (RX sensitivity) and Clear Channel Assessment (CCA) threshold to reduce the impact of interference and maintain network performance. RX sensitivity refers to the minimum signal strength required at the receiver's antenna port to decode a signal accurately. Reducing RX sensitivity filters out weaker signals, such as those caused by the deceptive jammer, effectively decreasing its disruptive effects. On the other hand, increasing the CCA threshold allows devices to tolerate higher levels of background interference before deciding that the channel is occupied. This adjustment can enhance the ability of legitimate nodes to access the channel despite interference, improving throughput in dense environments. While CCA threshold adjustments primarily aim to mitigate interference rather than identify attackers, they play a crucial role in maintaining network stability by dynamically adapting to the jamming environment. Together, these techniques ensure robust mitigation against jamming attacks without compromising legitimate communication.

7 Performance Evaluation

7.1 Attack Response

A random forest-based anomaly detection model incapable of distinguishing between different attack type or average throughput of the nodes is simulated, and the results are compared with those of SAMM (Fig. 9) [41]. The average node throughput and the number of out-of-service nodes (throughput = 0) are measured during a simulated attack lasting 10 s. Applying the SAMM model is observed to reduce the number of out-of-service nodes significantly and maintain an average throughput of 2.69 Mbps, irrespective of the number of nodes. In comparison, the peak average throughput of the normal detection model is observed

to be 0.61 Mbps with no mitigation actions. Notably, surplus nodes impact network performance due to collisions. However, throughput is observed to improve when more than five nodes are implemented in the proposed SAMM environment. Anomaly detection using random forest delivers high performance with a relatively high true positive rate, but for a large dataset, it requires a lot of computational power and leads to long training times. In comparison, the SAMM model provides fast learning by applying LGBM and higher detection accuracy compared to the conventional model by adding a multi-classification detection algorithm to classify attacks apart from normal communication.

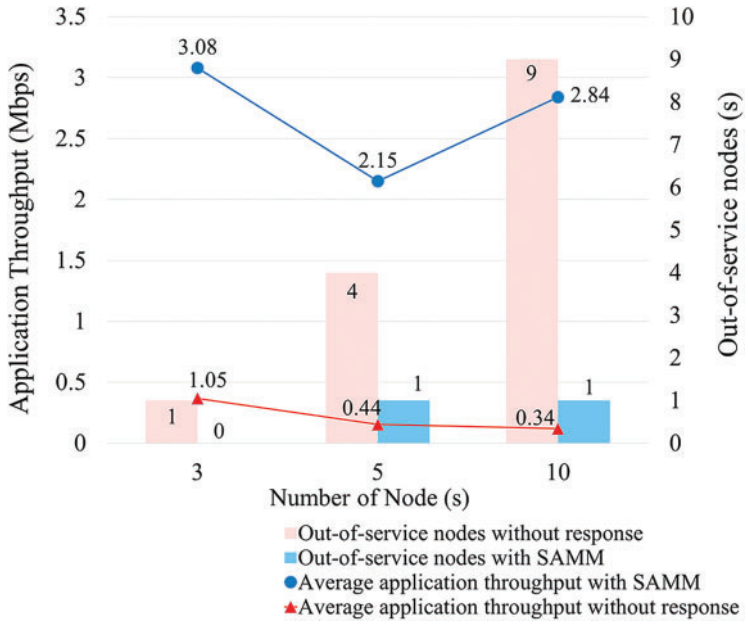

Figure 9: Performance evaluation with respect to the number of nodes

7.2 Attack Detection

Fig. 10 depicts the precision, recall, accuracy, and F1 score of SAMM and the ordinary anomaly detection model. Because the normal model assumes all packets apart from normal ones to be related to attacks, it achieves a precision of 100% but at the cost of a very high FN rate. Additionally, in the absence of an algorithm to determine attack types, it is incapable of mitigating network degradation despite exhibiting an accuracy of 82%. In contrast, SAMM achieves a classification accuracy of 96% and restores/retaines network performance successfully using mitigation techniques. In the experimental environment, the number of STAs was limited to 10, and a significant improvement in throughput was observed under these conditions. Although large-scale WSN environments were not tested, similar performance improvements are expected due to the adaptive nature of the proposed technique. Future research will focus on evaluating scalability and performance in large-scale WSN environments with resource-constrained devices.

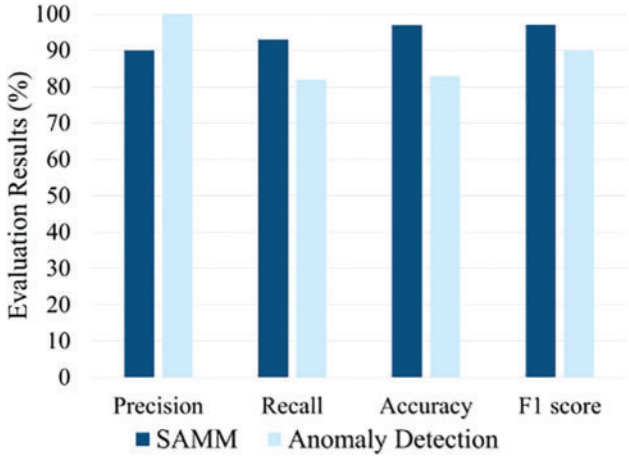

Figure 10: Comparison of attack detection model

8 Conclusions

Wireless communication systems based on the DCF mechanism are vulnerable to competition-based selfish and jamming attacks. Dense overlapping WLANs are particularly susceptible to interference from hidden nodes and intentional DoS attacks. However, further research is required to develop multiple attack detection and response capabilities. The SAMM model proposed in this paper is demonstrably effective at detecting and responding to selfish and jamming attacks based on performance comparison with a standard anomaly detection model. Its accuracy is higher than that of the standard model by more than 14%, and it is capable of choosing and implementing accurate responses quickly by distinguishing between the two types of attacks. For example, it is observed to restore network performance to normal levels in response to a simulated selfish attack. Nevertheless, the simulation scenario considered in this study is limited because performance results depend on the calculation methods and thresholds used. Future works should attempt to further improve the accuracy of attack detection by including attack patterns that do not depend on a single threshold. Furthermore, the algorithm is to be developed to respond not only to selfish and jamming attacks but also to DoS attacks such as battery depletion attacks, which cause network performance degradation and compromise device availability. In real wireless communication environments, performance is easily degraded due to various interferences that disrupt normal communication beyond the attacks presented in this paper. Enhancing the SAMM algorithm to distinguish not only intentional selfish and jamming attacks but also unintentional interference, makes it possible to improve the performance of real wireless network environments.

Acknowledgement: None.

Funding Statement: This work was supported by the Ministry of Trade, Industry and Energy (MOTIE) under Training Industrial Security Specialist for High-Tech Industry (RS-2024-00415520) supervised by the Korea Institute for Advancement of Technology (KIAT), and the Ministry of Science and ICT (MSIT) under the ICT Challenge and Advanced Network of HRD (ICAN) Program (No. IITP-2022-RS-2022-00156310) supervised by the Institute of Information & Communication Technology Planning & Evaluation (IITP).

Author Contributions: Study conception and design: Soyoung Joo, So-Hyun Park, Il-Gu Lee; data collection: Hye-Yeon Shim, Ye-Sol Oh; analysis and interpretation of results: Soyoung Joo, So-Hyun Park, Hye-Yeon Shim, Ye-Sol Oh,

Il-Gu Lee; draft manuscript preparation: Soyoung Joo, So-Hyun Park; Supervision: Il-Gu Lee; funding acquisition: Il-Gu Lee. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Hasan MZ, Hanapi ZM, Hussain MZ. Wireless sensor security issues on data link layer: a survey. Comput Mater Contin. 2023;75(2):4065–84. doi:10.32604/cmc.2023.036444.
2. Majid M, Habib S, Javed AR, Rizwan M, Srivastava G, Gadekallu TR, et al. Applications of wireless sensor networks and internet of things frameworks in the Industry Revolution 4.0: a systematic literature review. Sensors. 2022;22(6):2087. doi:10.3390/s22062087.
3. Landaluce H, Arjona L, Perallos A, Falcone F, Angulo I, Muralter F. A review of IoT sensing applications and challenges using RFID and wireless sensor networks. Sensors. 2020;20(9):2495. doi:10.3390/s20092495.
4. Angueira P, Val I, Montalban J, Seijo Ó, Iradier E, Fontaneda PS, et al. A survey of physical layer techniques for secure wireless communications in industry. IEEE Commun Surv Tutorials. 2022;24(2):810–38. doi:10.1109/COMST.2022.3148857.
5. Liu J, Aoki T, Li Z, Pei T, Choi Y-J, Nguyen K, et al. Throughput analysis of IEEE 802.11 WLANs with inter-network interference. Appl Sci. 2020;10(6):2192. doi:10.3390/app10062192.
6. Edalat Y, Obraczka K, Ahn JS. Smart adaptive collision avoidance for IEEE 802.11. Ad Hoc Netw. 2022;124(3):102721. doi:10.1016/j.adhoc.2021.102721.
7. Lee CK, Rhee SH. Collision avoidance in IEEE 802.11 DCF using a reinforcement learning method. In: 2020 International Conference on Information and Communication Technology Convergence (ICTC); 2020; Jeju, Republic of Korea. p. 898–901.
8. Choi WY. Fair MAC protocol for IEEE 802.11 wireless LANs with hidden node problem. J Electr Eng. 2020;71(5):365–7. doi:10.2478/jee-2020-0050.
9. Bellardo J, Savage S. 802.11 denial-of-service attacks: real vulnerabilities and practical solutions. In: 12th USENIX Security Symposium (USENIX Security 03); 2003; Washington, DC, USA.
10. Rath M, Mishra S. Advanced-level security in network and real-time applications using machine learning approaches. In: Machine learning and cognitive science applications in cyber security. New York, NY, USA: IGI Global; 2022. p. 84–104. doi:10.4018/978-1-5225-8100-0.ch003.
11. Thing VL. IEEE 802.11 network anomaly detection and attack classification: a deep learning approach. In: 2017 IEEE Wireless Communications and Networking Conference (WCNC); 2017; San Francisco, CA, USA. p. 1–6.
12. Agarwal M, Pasumarthi D, Biswas S, Nandi S. Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization. Int J Mach Learn Cybern. 2016;7(6):1035–51. doi:10.1007/s13042-014-0309-2.
13. Wang N, Jiao L, Wang P, Li W, Zeng K. Machine learning-based spoofing attack detection in mmWave 60GHz IEEE 802.11 ad networks. In: IEEE INFOCOM 2020-IEEE Conference on Computer Communications; 2020; Toronto, ON, Canada. p. 2579–88.
14. Upadhyaya B, Sun S, Sikdar B. Machine learning-based jamming detection in wireless IoT networks. In: 2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS); 2019; Singapore. p. 1–5.
15. Mamdouh M, Elrukhsi MA, Khattab A. Securing the internet of things and wireless sensor networks via machine learning: a survey. In: International Conference on Computer and Applications (ICCA); 2018; Beirut, Lebanon: IEEE. p. 215–8. doi:10.1109/COMAPP.2018.8460440.

16. R. Arunkumar J, Velmurugan S, Chinnaiah B, Charulatha G, Ramkumar Prabhu M, Prabhu Chakkaravarthy A. Logistic regression with elliptical curve cryptography to establish secure IoT. Comput Syst Sci Eng. 2023;45(3):2635–45. doi:10.32604/csse.2023.031605.
17. Lee YR, Park NE, Kim SY, Lee IG. Malicious traffic compression and classification technique for secure Internet of Things. Comput Mater Contin. 2023;76(3):3465–82. doi:10.32604/cmc.2023.041196.
18. Jeon SE, Lee SJ, Lee EY, Lee YJ, Ryu JH, Moon JH, et al. An effective threat detection framework for advanced persistent cyberattacks. Comput Mater Contin. 2023;75(2):4231–53. doi:10.32604/cmc.2023.034287.
19. Arafat Y, Yeaser K, Rahman A, Dasgupta A. A machine learning based approach for protecting wireless networks against DoS Attacks. In: Proceedings of the 7th International Conference on Networking, Systems and Security; 2020; New York, NY, USA. p. 126–32. doi:10.1145/3428363.3428377.
20. Konorski J. A game-theoretic study of CSMA/CA under a backoff attack. IEEE ACM Trans Netw. 2006;14(6):1167–78. doi:10.1109/TNET.2006.886298.
21. Kim J, Kim KS. Detecting selfish backoff attack in IEEE 802.15.4 CSMA/CA using logistic classification. In: 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN); 2018; Prague, Czech Republic. p. 26–7. doi:10.1109/ICUFN.2018.8436952.
22. Odedra L, Revar A, Lunagaria M. Detection and prevention of selfish attack in MANET using dynamic learning. IOSR JCE. 2016;18(3):54–61.
23. Fihri WF, Ghazi HE, Majd BAE, Bouanani FE. A machine learning approach for backoff manipulation attack detection in cognitive radio. IEEE Access. 2020;8:227349–59. doi:10.1109/ACCESS.2020.3046637.
24. Chakraborty S, Sanyal DK, Chattopadhyay S. Performance of random access games over an IEEE 802.11ac test bed. In: IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS); 2019; Goa, India. p. 1–4.
25. Xu W, Ma K, Trappe W, Zhang Y. Jamming sensor networks: attack and defense strategies. IEEE Netw. 2006;20(3):41–7. doi:10.1109/MNET.2006.1637931.
26. Djuraev S, Nam SY. Channel-hopping-based jamming mitigation in wireless LAN considering throughput and fairness. Electronics. 2020;9(11):1749. doi:10.3390/electronics9111749.
27. Vadlamani S, Eksioglu B, Medal H, Nandi A. Jamming attacks on wireless networks: a taxonomic survey. Int J Prod Econ. 2016;172:76–94. doi:10.1016/j.ijpe.2015.11.008.
28. Kanwar J, Finne N, Tsiftes N, Eriksson J, Voigt T, He Z et al. JamSense: interference and jamming classification for low-power wireless networks. In: 13th IFIP Wireless and Mobile Networking Conference (WMNC); 2021; Montreal, QC, Canada. p. 9–16. doi:10.23919/WMNC53478.2021.9619007.
29. Liu C, He A, Liu G, Wen Y. RFL-APIA: a comprehensive framework for mitigating poisoning attacks and promoting model aggregation in IIoT federated learning. IEEE Trans Ind Inform. 2024;20(1):123–34. doi:10.1109/TII.2024.3431020.
30. Bai L, Han P, Wang J, Wang J. Throughput maximization for multipath secure transmission in wireless ad-hoc networks. IEEE Trans Wirel Commun. 2024;23(5):987–1002. doi:10.1109/TCOMM.2024.3409539.
31. Gong Y, Yao H, Liu X, Bennis M. Computation and privacy protection for satellite-ground digital twin networks. IEEE Trans Commun. 2024;19(7):2564–78. doi:10.1109/TCOMM.2024.3392795.
32. Zerguine N, Aliouat Z, Mostefai M, Harous S. M-BEB: enhanced and fair binary exponential backoff. In: 14th International Conference on Innovations in Information Technology (IIT); 2020; Al Ain, United Arab Emirates. p. 142–7. doi:10.1109/IIT50501.2020.9299014.
33. Zhang C, Chen P, Ren J, Wang X, Vasilakos AV. A backoff algorithm based on self-adaptive contention window update factor for IEEE 802.11 DCF. Wirel Netw. 2017;23(3):749–58. doi:10.1007/s11276-015-1184-9.
34. Kobbaey T, Hamzaoui R, Ahmad S, Al-Fayoumi M, Thomos N. Enhanced collision resolution and throughput analysis for the 802.11 distributed coordination function. Int J Commun Syst. 2021;34(16):e4953. doi:10.1002/dac.4953.
35. Li T, Tang T, Chang C. A new backoff algorithm for IEEE 802.11 distributed coordination function. In: Sixth International Conference on Fuzzy Systems and Knowledge Discovery; 2009; Tianjin, China. p. 455–9. doi:10.1109/FSKD.2009.513.

36. Deng C, Fang X, Han X, Wang X, Yan L, He R, et al. IEEE 802.11be wi-fi 7: new challenges and opportunities. IEEE Commun Surv Tutorials. 2020;22(4):2136–66. doi:10.1109/COMST.2020.3012715.
37. López-Raventós A, Bellalta B. Multi-link operation in IEEE 802.11be WLANs. IEEE Wireless Commun. 2022;29(4):94–100. doi:10.1109/MWC.006.2100404.
38. Seok Y, Tsao W, Bajko G, Yee J, Liu J, Cheng P, et al. 'Enhanced multi-band/multi-channel operation,' IEEE 802.11 documents; 2019 May [cited 2024 Oct 20]. Available from: http://ieee.org/802.11/documents?is_dcn=0766&is_group=00be.
39. Islam MK, Hridi P, Hossain MS, Narman HS. Network anomaly detection using lightgbm: a gradient boosting classifier. In: 2020 30th International Telecommunication Networks and Applications Conference (ITNAC). 2020; Melbourne, VIC, Australia: IEEE. p. 1–7. doi:10.1109/ITNAC50341.2020.9315049.
40. Seth S, Singh G, Chahal KK. A novel time efficient learning-based approach for smart intrusion detection system. J Big Data. 2021;8(1):111. doi:10.1186/s40537-021-00498-8.
41. Elmrabit N, Zhou F, Li F, Zhou H. Evaluation of machine learning algorithms for anomaly detection. In: 2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security); 2020; Dublin, Ireland. p. 1–8. doi:10.1109/CyberSecurity49315.2020.9138871.