Engineering Science and Military Technologies
Volume (9) - Issue (2) - Sep 2025

ISSN: 2357- 0954
DOI:10.21608/EJMTC.2025.344143.1293

Mitigating version number attacks in RPL-based IoT networks : A machine learning approach

Original Article

Ammar Ibrahim El Sayed, Mahmoud Abdelaziz, Mohamed Hussein, Ashraf D. Elbayoumy

[1]Department of Electrical Engineering, Military Technical College, Cairo, Egypt

Keywords:
DoS, IoT, RPL, VNA, ML

Corresponding Author:
Mahmoud Abdel-Aziz, Department of Electrical Engineering, Military Technical College, Cairo, Egypt, **Tel:** 01110190871,
E-mail: mkaremm@ieee.org

Received : 13 December 2024
Accepted : 23 February 2025

Abstract
This paper investigates the Version Number Attack (VNA), a form of Denial of Service (DoS) threat, within the Routing Protocol for Low-Power and Lossy Networks (RPL) in IoT-based Wireless Sensor Networks (WSNs). The study assesses the impact of VNAs across various attack scenarios, including single, double, and triple attacker setups, on critical network performance metrics such as power consumption, packet loss, and delay. Using a simulated WSN environment, datasets are generated under normal and attack conditions to evaluate network behavior. A Random Forest (RF) machine learning (ML) model is employed for feature selection, identifying the most significant metrics for attack detection. The results demonstrate that increasing the number of attackers drastically affects network performance, particularly in power consumption and packet loss, leading to significant degradation in overall network reliability. This research contributes to developing efficient security strategies for IoT networks operating under RPL protocols by providing an in-depth analysis of VNA effects and leveraging ML for mitigation.

1. INTRODUCTION

The rapid growth in wireless sensor networks (WSNs) deployment has revolutionized numerous industrial and social sectors by enabling unprecedented levels of connectivity and data-driven insights[1]. However, the increasing complexity and scale of WSNs have introduced significant security challenges, particularly in low-power and lossy networks where resource constraints are typical[2]. Since wireless sensor devices are inherently vulnerable to cyberattacks, establishing robust security mechanisms is critical to ensure their functionality, reliability, and user trust. RPL is a foundational protocol that enables efficient routing in such environments. However, its intrinsic vulnerabilities underscore the urgent need for advanced security strategies to address sophisticated cyber threats effectively[3].

WSNs are vulnerable to cybersecurity threats such as eavesdropping and aggressive tampering[4]. One of the significant risks is Denial of Service (DoS) attacks, which exploit protocol vulnerabilities and compromise network stability and integrity[5]. A sophisticated form of such attack is the Version Number Attack (VNA), which targets the routing protocol for RPL, causing network disruption[6]. These attacks can result in routing issues, data misdirection, or network segmentation, leading to reduced performance and reliability of the network.

In order to protect nodes in WSNs from vulnerabilities, it is important to take a comprehensive approach that involves strengthening security features in protocols and implementing robust detection and mitigation strategies. This includes using cryptographic techniques such as authentication mechanisms and continuous monitoring as crucial elements of the defense strategy[7-9]. As attack techniques become more advanced, it is essential to understand the nature and impact of these attacks, particularly in scenarios involving multiple attackers. It is not just about safeguarding individual devices but also ensuring the resilience and reliability of the entire network. By analyzing the effects of VNAs and identifying effective countermeasures in multi-attacker scenarios, we can significantly contribute to the broader goal of securing WSNs against an ever-evolving array of cyber threats.

We surveyed RPL version number attacks in low-power and lossy networks. Our analysis covers three perspectives, including single and multiple attackers and version attacks. We explored the impact of these attacks on power consumption, packet delivery ratio, delay, and control packet overhead. The survey offers valuable insights and enhances security measures in low-power and lossy networks.

The attacks that can occur on RPL version numbers are mentioned in[10-12]. A probabilistic attacking model measures the potential impact, which analyzes power consumption, packet delivery ratio, delay, and control packet overhead. The results indicated that a higher probability of attacks could lead to longer delays and increased control packet overhead. In particular, the VNA is found to amplify the average delay up to six times in

the worst case. A research paper[13] examines the impact of RPL version number attacks by multiple attackers on IoT networks. The study found that multiple attackers can cause issues with the packet delivery ratio. Closer attackers can cause longer delays and higher power consumption. The research also evaluated a proposed mitigation technique. Other studies[14,15] examine the impact of RPL Destination Oriented Directed Acyclic Graph (DODAG) version attacks on routing optimization in constrained networks. The study analyzed various metrics, including overhead, delivery ratio, end-to-end delay, rank inconsistencies, and loops. The research found that attackers closer to the root cause more loops and inconsistencies, while further attackers lead to worse delivery ratios.

Our study aims to analyze the impact of multiple VNAs in RPL-based IoT networks. We are motivated to undertake this analysis because we identified a gap in the examination of VNAs' various impacts. We explore the relationship between the number of attackers and the effects of the attacks, aiming to identify the most affected performance metrics.

This work provides three key contributions. First, it presents a comprehensive analysis of the VNA on the RPL routing protocol in WSNs, assessing its impact on network performance. Second, a novel simulated dataset is generated, encompassing normal WSN conditions, scenarios with a single compromised node, and cases with multiple compromised nodes, to evaluate the varying effects of the attack. Third, a feature selection algorithm, Random Forest (RF), is applied to identify the most impacted metrics and features, reducing computational complexity and enabling efficient detection through machine learning (ML).

The paper follows a structured organization. Section 2 delves into detailed analyses of various attacking scenarios, comprehensively exploring the intricacies associated with DoS threats, mainly focusing on the VNA. In Section 3, the simulation environment and metrics are meticulously presented, offering insights into the experimental setup and the key performance indicators for evaluating attacks' impact on WSNs. Section 4, dataset construction and data analysis, provides complete data generation by simulation and analysis of all the metric comber in each typical and attacked scenario. Section 5, the ML model generation phases. Finally, the conclusion section synthesizes the findings.

2. ATTACKING SCENARIOS

This section aims to conduct an in-depth investigation of 12 smart devices across three distinct usage scenarios. An effort has been made to evaluate performance under varying conditions, create a dataset comprising these three scenarios, and build a model based on them. The energy consumption is first assumed to be expected (Normal), where average usage would indicate "normal" behavior. Subsequently, the energy consumption and operational performance are evaluated for the scenarios involving Planned DoS and Network DoS (PDoS and NDoS) concerning their impact on the system. For this study, only one attack is considered to create the "worst-case" conditions, aimed at providing insights for designing energy-efficient resilience strategies.

The contribution is made in the form of a planned DoS attack against a version attack carried out against RPL security mechanisms to observe and quantify its effects. In the third scenario, a simulation uses two or three attackers in a complex environment to explore adaptability and responsiveness under various attack complexities. Note that each of the selected DoS scenarios has been deliberately designed to introduce modifications in the security mechanisms. This design emulates real-world threats that exploit existing vulnerabilities in the routing protocol for RPL.

Fig. 1 summarizes the overall process, including network setup, data collection, and evaluation. It illustrates the flow from traffic capture to the application of ML models. The figure highlights the simulation setup for normal and malicious scenarios, traffic capture, analysis of key performance metrics (power consumption, packet loss, and delay), and the detection and evaluation process using RF for feature importance and attack detection.

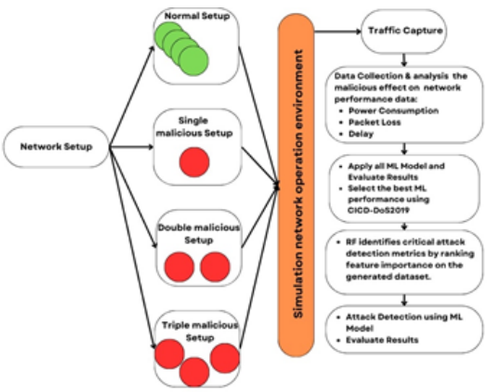
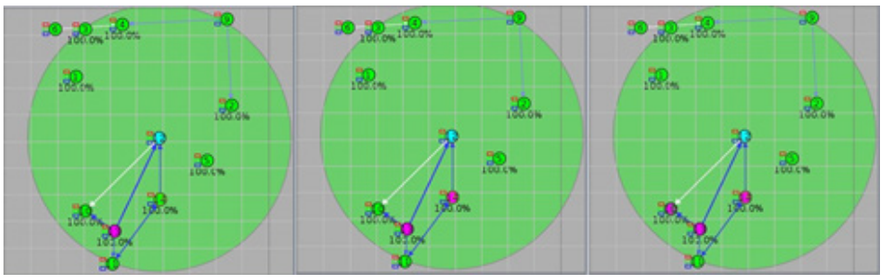


Fig. 1: Process network setup of each scenario and data generation



(a) One DoS Attack (b) Two DoS Attacks (c) Three DoS Attacks

Fig. 2: Impact of multiple DoS attacks on WSN

Fig. 2 presents three simulation scenarios: (a) one DoS attack, (b) two DoS attacks, and (c) three DoS attacks, illustrating the increasing impact on the WSN. The results show that more attackers significantly affect network performance metrics such as power consumption, packet loss, and delays. These scenarios allow us to evaluate the network's adaptability and highlight the need for effective mitigation strategies against multi-attacker challenges. Additionally, the simulation demonstrates the scalability of the proposed methodology, ensuring its effectiveness in diverse IoT environments. Insights gained from this research improve energy efficiency and attack detection, thereby enhancing the reliability and security of IoT networks.

Algorithm 1, "RPL Operations," is a systematic framework designed to efficiently manage routing operations in low-power and lossy networks using RPL. The RPL operations algorithm is based on RFC 6550, defining the IPv6-based low-power and lossy networks standard. The work of inspires specific implementations[16]. The algorithm begins with the RPL Initialize function, which configures RPL with essential parameters. The process starts by selecting a main node, known as the root node, which serves as the network's central hub. The root node determines that the system will handle two types of messages for network management. The Destination Advertisement Object (DAO) messages convey and update the routing information, specifying where particular data should go within the network. At the same time, DODAG Information Object (DIO) messages help nodes locate the root node and establish the network structure. This approach ensures a well-organized network and facilitates the proper transmission of information.

The coordination at the overall level is done in the RPL Operations function, which allows for the smooth running of RPL protocols, with energy efficiency in focus, keeping network stability. Similarly, the algorithm provides a security feature in the Algorithm 2 version attack function, which emulates a version attack against RPL. The function launches an attack, looking for malicious versions in the DIO messages received and performing appropriate logging of the attack or processing of legitimate DIO messages. This makes the algorithm more reliable regarding its resistance to potential adversarial actions. In general, the algorithm's modular design and strategic organization make it efficient and adaptive to deal with the challenges brought by the low-power and lossy network environments.

Algorithm 1 Enhanced RPL Operations

```
function RPL_Initialize(NodeInfo, DefaultParams)
    Initialize RPL protocol using NodeInfo and DefaultParams.
    Configure DODAG parameters and assign roles.
end function
function RPL_Determine_Root(NodeInfo)
    Evaluate root eligibility for each node in NodeInfo.
    return Node with the highest priority (based on rank, ETX, etc.).
end function
function RPL_Handle_DAO(NodeInfo, ReceivedDAOs)
    for all nodes n in NodeInfo do
        Process ReceivedDAOs and update the downward routing table.
    end for
    Update global network state.
end function
function RPL_Handle_DIO(NodeInfo, ReceivedDIOs)
    for all nodes n in NodeInfo do
        Process ReceivedDIOs and update the rank of each node.
    end for
    Reconfigure the DODAG structure as needed.
end function
function RPL_Operations(NodeInfo, DefaultParams, ReceivedMessages)
    Call RPL_Initialize(NodeInfo, DefaultParams).
    RootNode ← RPL_Determine_Root(NodeInfo).
    Call RPL_Handle_DAO(NodeInfo, ReceivedMessages.DAO).
    Call RPL_Handle_DIO(NodeInfo, ReceivedMessages.DIO).
end function
```

Algorithm 2 Version Attack Detection

```
function VERSION_ATTACK(NodeInfo, MaliciousVersion, ReceivedDIOs)
    Launch the version attack by broadcasting MaliciousVersion.
    for all ReceivedDIO in ReceivedDIOs do
        if Malicious version detected in ReceivedDIO then
            Log the version attack details.
        else
            Process ReceivedDIO normally.
        end if
    end for
end function
```

3. SIMULATION ENVIRONMENT AND METRICS

Our simulation setup is chosen for its practicality. It utilizes Contiki Cooja version 3.0 and its simulator, Cooja. Contiki is open-source and a comprehensive implementation of the standardized protocol stack[17]. The setup featured Sky motes as nodes, running the RPL-UDP application. In this scenario, UDP clients on nodes transmitted temperature readings to the UDP server running on the root node every minute, mirroring real-world IoT applications. We employed a distance-based loss model for the radio medium, with transmission and interference ranges set at 50 m and 100 m, respectively. The simulated topology covered an area of 150 m × 150 m.

For each attacking scenario, we conducted five independent simulations to ensure the reliability and consistency of the results. Each simulation is performed under identical network and environmental conditions, including the same topology, transmission ranges, and attack configuration. The simulations are executed sequentially, with each run lasting 5 minutes. This duration allows sufficient time for the network to stabilize and for performance metrics to be accurately measured under the specified conditions.

The results from these simulations are aggregated and analyzed to derive statistically significant metrics, with a 95% confidence interval applied to ensure robustness and reduce the impact of outliers. The primary focus of the analysis is on performance metrics, including average power consumption, which measures the overall energy usage across nodes; instance power, representing the instantaneous power usage during specific time intervals; and duty cycle evaluation, which assesses the percentage of time nodes spend in active states versus idle or low-power states. These metrics are crucial for comprehensively evaluating the network's behavior under different attacking scenarios, highlighting the system's stability and resilience. The experiments are executed on a Dell Inspiron laptop equipped with a 15.6-inch display powered by a Windows 10 64-bit operating system, ensuring a stable and consistent computational environment for the simulations.

The computer specifications are outlined in Table 1 below. In a Wireless Sensor Network (WSN), power consumption during regular operation remains constant, reflecting the typical energy usage of the network when it is not under attack. However, when an attack occurs, there is a noticeable increase in power consumption. This surge results from the disruption to normal network operations caused by the attack. As the number of attackers rises from one to two or three, the impact on power consumption becomes even more significant. The attack disrupts the network's routing table, leading to frequent retransmissions. When the number of attackers increases, their manipulation further intensifies the disruption to the routing table, causing an even more substantial increase in power consumption.

The standard radio duty cycle in a WSN operates normally under regular conditions, representing the typical duty cycle without any attacks. However, when an attack occurs, it disrupts the system, increasing the average radio duty cycle. This disruption becomes more significant when the number of attackers increases from one to two. Consequently, the attack has a greater impact on the network's expected functioning, resulting in an even higher average radio duty cycle. The increased duty cycle is due to the attack interfering with the regular communication patterns, causing nodes to remain active for longer periods.

The power consumption in a WSN usually remains steady during normal conditions. This level reflects the typical power usage without any attacks. However, the average power consumption increases during a DoS attack. The attack causes network disruptions, which leads to higher power usage than in normal conditions. The impact becomes even more significant when the number of attackers increases from one to two or three. The intensified attack further escalates power consumption because of its effect on the network's routing table. Due to the attack's effect on the routing table, the data retransmissions get repeated, resulting in a continuous cycle of increased power consumption. Furthermore, the escalation in attacks from one to two makes the situation worse, amplifying the impact on the routing table and causing a more significant increase in power consumption.

4. THE PROPOSAL DATASET AND DATA ANALYSIS

The proposal dataset is carefully constructed by emulating a WSN comprising 12 nodes in each scenario. The traffic is captured under four marked scenarios: normal operation, single attack, double and triple attack. Each scenario is indicated by a data label, which categorizes the data as normal, malicious (single attack), or severely malicious (double or triple attack). The proposal dataset encompasses 22 network metrics collected under these varied conditions. Each row corresponds to a unique node in the network, providing a comprehensive view of network performance and behavior.

Network Setup, this initial phase involves configuring a WSN with 37 rows. Each node is set up to communicate with its neighbors, creating a robust mesh network topology. Essential network parameters, such as node IDs, beacon intervals, and initial power levels, are configured.

Traffic Emulation. Normal operation, the network functions under standard conditions without any malicious activities. The traffic patterns, including data transmission and reception, are continuously monitored and recorded. Single attack, one node in the network is compromised to simulate a malicious attack. This could involve activities like injecting false data, jamming signals, or creating routing loops. The impact on network metrics due to this attack is documented. Double and triple attacks, two or three nodes are simultaneously compromised to simulate a more severe attack. The combined effects on network performance and metrics are meticulously captured.

Table 1: Computer, virtual machine, simulation, and experimental specifications

Computer Specifications	
Processor	Intel® Core™ i7-5500U CPU @ 2.40 GHz
RAM	8 GB
Virtual Machine Specifications	1 CPU
Processor	10 GB
Storage	2 GB
RAM	
Simulation and Experimental Details	
Simulator	Contiki Cooja (version 3.0)
Node Type	Tmote Sky motes
Number of Nodes	12 per scenarios
Number of Sink Nodes	1
Number of Sender Nodes	11
Number of Malicious Nodes	1, 2 & 3
Application	RPL-UDP
Transmission Range	50 m
Interference Range	100 m
Topology Dimensions	150 m × 150 m
Experimental Runs	
Number of Simulations	5
Duration of Each Simulation	5 minutes
Confidence Interval	95%

Data Preprocessing

1. Data Collection, throughout each scenario, various network metrics are gathered. These metrics include packet statistics (received, duplicates, lost), network dynamics (hops, routing metrics like Rtmetric and ETX), node churn, beacon intervals, reboots, and power consumption metrics (CPU power, LPM power, Listen power, Transmit power). Additional metrics related to packet timing (average, minimum, and maximum inter-packet times) and duty cycles (listening and transmitting duty cycles) are also collected.

2. Data Labeling, the gathered data is labeled according to the scenario it is captured in normal operation, single attack, or double attack. This labeling is crucial for developing and evaluating ML models for anomaly and intrusion detection in WSNs.

3. Dataset Compilation, raw data undergoes preprocessing to eliminate variations and normalize the metrics for better comparison. This step involves handling missing

values, scaling the data, and transforming the metrics into a format suitable for analysis. preprocessed data from all three scenarios is compiled into a single dataset. Each row represents a unique node's metrics at a specific time. The dataset is structured to facilitate analysis and modeling, with columns representing various metrics and rows representing individual data points from the nodes.

Table 2: Dataset metrics description

Metric	Description
id	Unique identifier for each node
received	Total number of packets received by a node
dups	Count of duplicate packets received
lost	Number of packets lost (not received by the destination)
hops	Number of intermediate nodes a packet passes through
Rtmetric	Routing cost associated with a particular route
ETX	Expected number of transmissions required to successfully deliver a packet
churn	Frequency of nodes joining or leaving the network
beacon_interval	Time interval between successive beacon signals
reboots	Number of times a node has rebooted
cpu_power	Power consumption of the node's CPU
lpm_power	Power consumed by the node in low-power mode
Listen_power	Power consumption while the node is in listening mode
Transmit_power	Power consumption while the node is transmitting data
power	Total power consumption encompassing all activities
on_time	Total time the node is active (not in sleep mode)
listen_duty_cycle	Ratio of time the node spends in listening mode to the total time
transmit_duty_cycle	Ratio of time the node spends in transmitting mode to the total time
avg_inter_packet_time	Average time interval between consecutive packets
min_inter_packet_time	Minimum observed time interval between consecutive packets
max_inter_packet_time	Maximum observed time interval between consecutive packets
status	Operational status of the network (no_attack, attack, double_attack, triple_attack)

4.1 Average Routing Metric

In RPL, the average routing metric is a key measure of the overall quality of routes inside the network. It takes into account aspects such as hop count, connection quality, and energy usage. The calculation process involves collecting metrics for all available routes to a location, adding them together, and then dividing the total by the number of routes. This average routing metric is instrumental in route selection and sustaining efficient network performance, particularly in resource-constrained contexts where load balancing and reliable communication are critical. The dataset has three labels in node status (no attack- attack- double attack- triple attack) as in Fig. 3.

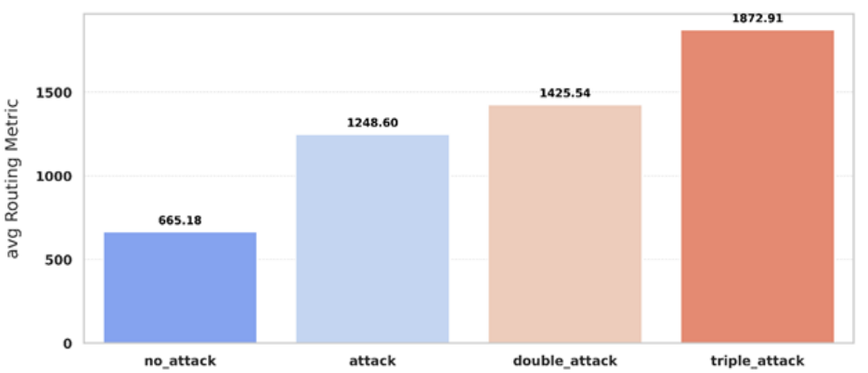


Fig. 3: Evaluation of average routing metric in no attack, single, double, and triple attack scenarios.

4.2 Average transmit duty cycle

The average transmit duty cycle, expressed as a percentage, measures how often a node transmits data relative to its total available time. For example, a device transmitting for 10 seconds every 100 seconds has a duty cycle of 10%. This metric is crucial in energy-limited environments like IoT networks, where optimizing the duty cycle enhances battery life and network efficiency. However, Denial of Service (DoS) attacks significantly impact the duty cycle. Under normal conditions, the duty cycle is low at 0.12%. During a single-node attack, it increases 20-fold to 2.11%, and in a triple-node attack, it rises 30-fold to 3.16%. This highlights the severe strain on network resources caused by malicious activities.

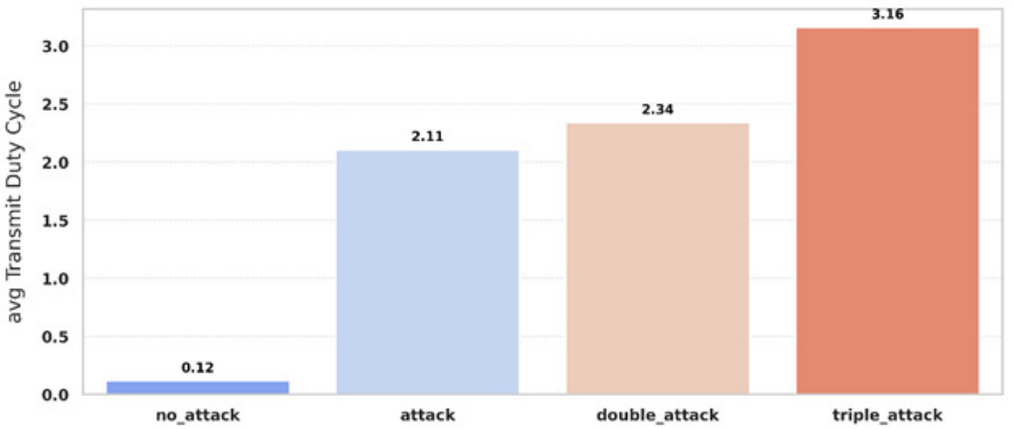


Fig. 4: Evaluation of average transmits duty cycle in no attack, single, double, and triple attack scenarios.

4.3 Average Transmit power

The average transmit power is the average amount of power utilized by a node during transmission operations during a certain time. It is commonly measured in watts or milliwatts (mW) and represents the device's power consumption during data transmission. For example, if a device transmits at different power levels but has an average power usage throughout time of 50 mW, its average transmit power is 50 mW. This statistic is critical for evaluating energy economy and maintaining battery life in wireless communication networks, especially in battery-powered or low-power devices as the attacks happen the transmit power is changed from 0.1 to 1 and above 1.2 these increases in the consumption power is affect the sensor lifetime and battery make the sensor vanishing very quickly as in Fig. 5

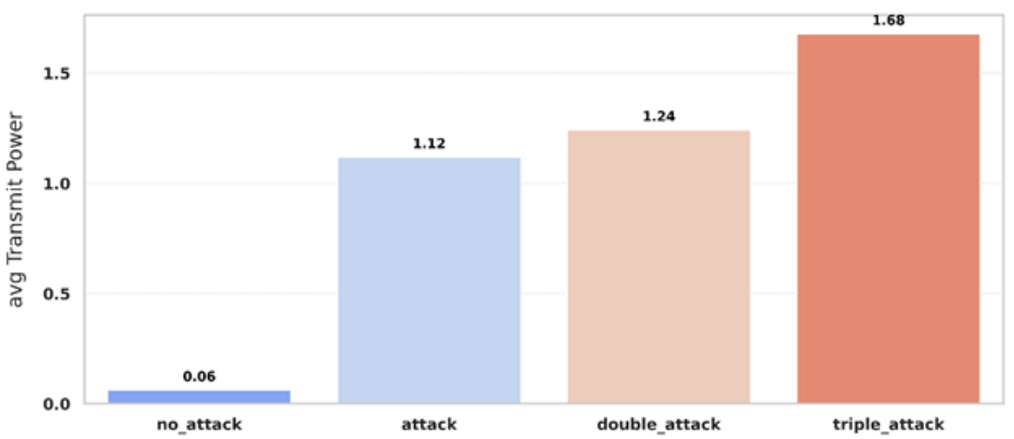


Fig. 5: Evaluation of average transmit power cycle in no attack, single, double, and triple attack scenarios.

4.4 Average inter packet time

When addressing network, the average time interval between two subsequent packets is referred to as inter-packet time (IPAT). An IPAT can be measured through network monitoring tools or from packet capture data. Understanding the components of IPAT enables network engineers to reduce congestion and latency. Fig. 6 illustrates that the average IPAT rises as more nodes in the network are targeted by attacks. When there are no attacks, the average is approximately 55,000 microseconds. If a single node is compromised, the average IPAT increases to 60,000 microseconds. When three nodes are compromised, the average IPAT increases to 90,000 microseconds. This implies that as more nodes are targeted, the time gap between packets reaching the network lengthens, resulting in reduced responsiveness.

Fig. 6: Evaluation of average inter packet time in no attack, single, double, and triple attack scenarios.

4.5 Average Packets lost

Average packet loss rate is an important metric for measuring network performance and represents the percentage of packets that do not reach their destination. Network monitoring analyzers are important for measuring this parameter because they track the number of packets sent against the number of packets received. It is worth noting that the average packet loss rate will increase during a DoS attack. This is because DoS attacks flood the network with large amounts of traffic, causing congestion and causing many packets to be lost, thus increasing the average packet loss. As in Fig.7 the no attack case has no lost packet where the number of the lost packet increases as the number of target node increases.

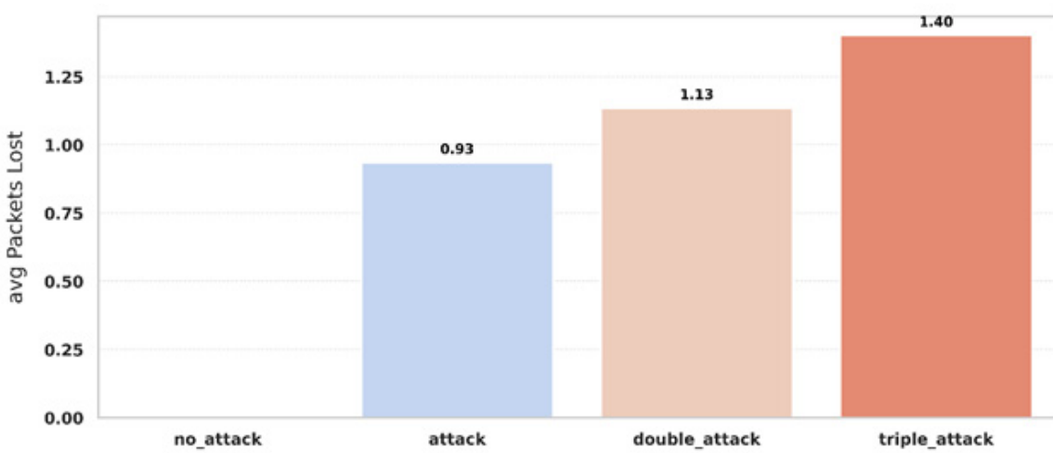


Fig. 7: Evaluation of average packet loss in no attack, single, double, and triple attack scenarios.

4.6 Average CPU Power

During a DoS attack, the average CPU power or CPU utilization of the affected network equipment or server is often significantly impacted. DoS attacks flood the system with heavy traffic or high-cost requests, causing the CPU to work harder to process and manage incoming requests. As a result, the average CPU power increases due to the additional computational demands, as illustrated in Fig. 8. High CPU utilization can lead to performance degradation, slower response times, and even system crashes if the CPU becomes overloaded. Monitoring changes in centralized CPU power is, therefore, essential to understanding the source of DoS stress in the network structure and implementing timely corrective measures. The figure shows that the average CPU power increases substantially as attack severity escalates. In a no-attack scenario, the CPU power remains low at 0.34, but this value nearly triples to 0.94 in a triple-attack scenario, highlighting the critical impact of DoS attacks on network resource utilization and performance.

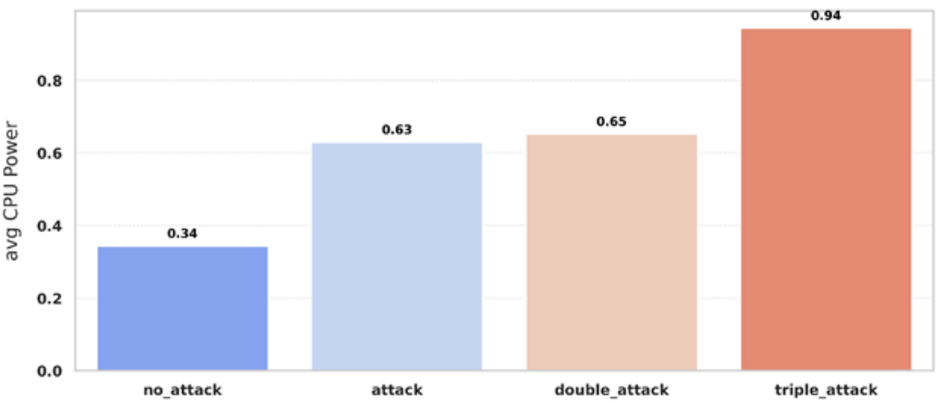


Fig. 8: Evaluation of average CPU power in no attack, single, double, and triple attack scenarios.

4.7. Average expected transmission count ETX

The average expected transmission count (ETX) is a metric used to evaluate the performance of wireless networks by estimating the number of transmissions required to successfully deliver a packet from a source to a destination. ETX serves as an indicator of connection reliability by accounting for factors such as packet loss and retransmissions. An increase in the ETX value often occurs when Denial of Service (DoS) attacks flood the network with excessive traffic. Such attacks can lead to higher packet collision rates and increased packet loss, necessitating additional retransmissions to successfully deliver packets. Consequently, the average ETX value rises, as illustrated in Fig. 9. Monitoring ETX is crucial for identifying network failures caused by attacks or other issues. It facilitates enhanced network management and troubleshooting, enabling timely interventions to maintain network reliability and performance.

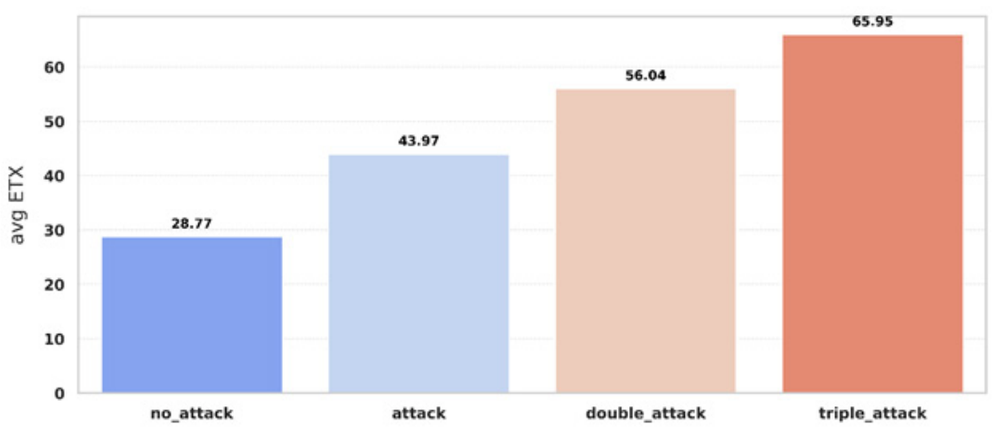


Fig. 9: Evaluation of average expected transmission count (ETX) in no attack, single, double, and triple attack scenarios.

4.8 Average Listen duty Cycle

The Average Listen Duty Cycle measures the percentage of time a device spends actively listening for signals in a wireless network. During a DoS attack, this duty cycle may increase due to the high volume of incoming traffic, leading to higher power consumption and reduced network efficiency. Conversely, network congestion from the attack might also lower the duty cycle if devices struggle to receive signals effectively as in Fig 10. The figure demonstrates that the Average Listen Duty Cycle increases significantly under attack scenarios. While the duty cycle is relatively low in the no-attack condition at 0.69, it rises sharply to 1.82 under a single attack and reaches 2.73 during a triple attack, indicating the heightened demand on devices to process incoming traffic during attacks. This escalation reflects the additional strain imposed on network devices in the presence of malicious activities.

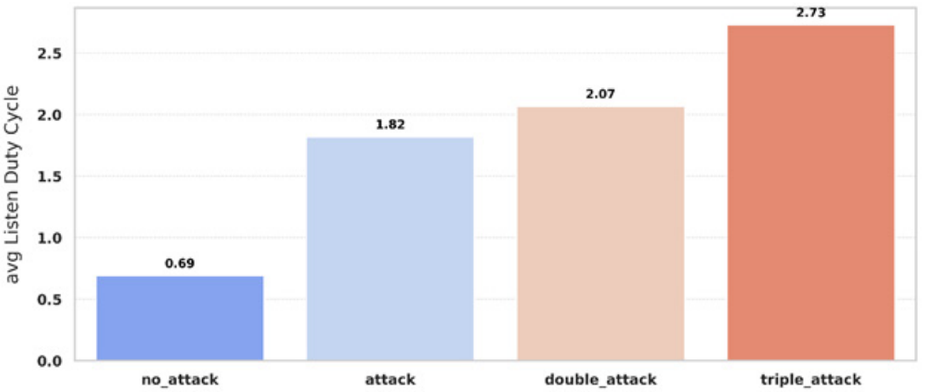


Fig. 10: Evaluation of average listen duty cycle in no attack, single, double, and triple attack scenarios

4.9 Average power consumption

Average power consumption measures the typical amount of energy used by a device or network component over time. During a DoS attack, average power consumption often increases because the device must process a higher volume of traffic or handle excessive computational tasks. This heightened activity can lead to greater power use, as the system works harder to manage the attack's impact, potentially affecting overall efficiency and operational costs. The Fig. 11 show that the power increases triple time just when one node is compromised and as the target node increases the overall consumption power increase which make the network lifetime is very short.

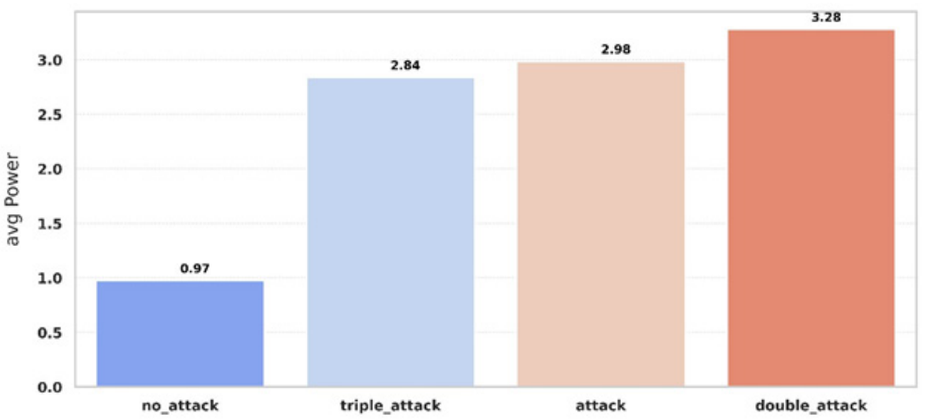


Fig. 11: Evaluation of average power consumption in no attack, single, double, and triple attack scenarios.

In Table 3, the no Attack scenario results in performance metrics that show the expected behavior of the network in the normal case. This provides useful context for evaluating the impact of subsequent attacks. These results show the direct impact of an attack on network performance, revealing vulnerabilities that attackers can exploit. Gain insight into multiple attacks at once. This comparison shows the reduction in network performance in terms of a combination of metrics such as energy efficiency and overall reliability. Increasing challenges in the context of development phase threats.

Table 3: The proposal dataset metric average score comparison

Metric Average score	No Attack	Attack	Double Attack	Triple Attack
Received	14.0667	21.4667	8.9333	32.2
Lost	0	0.933	1.133	1.4
Hops	2.1333	2.0947	2.6235	3.1421
Rtmetric	665.18	1248.6	1425.54	1872.91
ETX	28.77	43.97	56.04	65.95
Churn	0	6.467	2.2	9.7
Beacon Interval	675331.1	12721.07	13066.07	19081.6
CPU Power	0.3437	0.6296	0.6522	0.9444
LPM Power	0.1531	0.1444	0.1438	0.2167
Listen Power	0.4145	1.0914	1.2401	1.6371
Transmit Power	0.0621	1.1184	1.2418	1.6776
Power	0.9734	2.9839	3.2778	4.4758
Listen Duty Cycle	0.6908	1.819	2.0668	2.7285
Transmit Duty Cycle	0.1169	2.1063	2.3386	3.1594
Avg Inter-Packet Time	54951.53	59863.4	62243.2	89795.1
Min Inter-Packet Time	18933.33	17933.33	25066.67	26900

5. MATHEMATICAL MODEL FOR ATTACK DECISION ANALYSIS

This section presents a mathematical model to identify, classify, and optimize network performance amid various attack scenarios. The model incorporates key components, including characterizing different attack scenarios, their potential impacts on network behavior, and the mechanisms for detecting and classifying these attacks. The model uses mathematical formulations to quantify critical performance metrics such as throughput, delay, packet loss, and energy consumption under different attack conditions. This model aims to understand network vulnerabilities and implement effective countermeasures to mitigate attacks. Let A denotes the attack scenario as a discrete set:

$$A = \{A_0: \{No\ Attack\}, A_1: \{Single\ Attack\}, A_2: \{Double\ Attack\}, A_3: \{Triple\ Attack\}\} \quad (1)$$

Define the following performance metrics as a vector M:

$$M = \begin{Bmatrix} R_A \\ L_A \\ H_A \\ P_A \\ T_{avg,A} \\ B_A \\ C_A \end{Bmatrix} \quad (2)$$

Where:

R_A: Packets received.
L_A : Packets lost.
H_A: Average hop count
P_A: Total power consumption (mW)
B_A: Beacon interval (ms)
C_A: Churn rate
$T_{avg,A}$: Average inter-packet time (ms)

Performance index formula, for each attack A, the index is expressed as a function of the non-attack event:

Packets Received:

$$R_A = R_{N_0} - k_1 \cdot f_R(A) \quad (3)$$

Packets Lost:

$$L_A = L_{N_0} - k_2 \cdot f_L(A) \quad (4)$$

Average Hop Count:

$$H_A = H_{N_0} - k_3 \cdot f_H(A) \quad (5)$$

Total Power Consumption:

$$P_A = P_{N_0} - k_4 \cdot f_P(A) \quad (6)$$

Average Inter-Packet Time

$$T_{Avg,A} = T_{Arg,N_0} - k_5 \cdot f_T(A) \quad (7)$$

Beacon Interval:

$$B_A = B_{N_0} - k_6 \cdot f_B(A) \quad (8)$$

Churn Rate:

$$C_A = C_{N_0} - k_7 \cdot f_C(A) \quad (9)$$

Where k_i $(0 < i \leq 6)$ represents the sensitivity coefficients for each metric under attack conditions, and f_x (A) where $x \in \{R,L,H,P,S,T,B,C\}$, denotes nonlinear functions characterizing the impact of the attack on the respective metrics. The changes in metrics can encapsulated by:

$$\Delta M = M_A - M_{Ao} \tag{10}$$

Where:

$$\Delta M = \begin{Bmatrix} \Delta R_A \\ \Delta L_A \\ \Delta H_A \\ \Delta P_A \\ \Delta T_{avg,A} \\ \Delta B_A \\ \Delta C_A \end{Bmatrix} \tag{11}$$

The attack Classification function C(A) is defined as:

$$C(A) = \begin{cases} 1 & if\ \Phi_A > \theta_\Phi\ and\ L_A > \theta_L \quad \text{attack} \\ 0 & otherwise.\ C(A) \quad no\ \text{attack} \end{cases} \tag{12}$$

To classify attack scenarios, define a composite performance impact function Φ_A:

$$\Phi_A = \sum_{i=1}^{n} w_i . \Delta M_i, \tag{13}$$

Where, ΔM_i represents the change in metrics, and w_i denotes the weight assigned based on the importance of each metric, determined using optimization methods such as linear programming or ML. θ_Φ represents a threshold that defines the boundary between attack and no attack scenarios based on overall performance impact. θ_L signifies the threshold for packet loss that signals a potential attack.

6. ML MODEL SELECTION AND EVALUATION

This section is structured into two subsections. The first subsection involves a comparative analysis of several well-established ML algorithms to identify the model that performs best across various evaluation metrics. This comparison is essential for selecting the most suitable model for the task at hand. In the second subsection, the selected model is applied to different dataset, with efforts focused on optimizing its performance to achieve high accuracy tailored to the specific application. This systematic approach serves as a comprehensive guideline for choosing and implementing ML techniques, ultimately improving the efficiency and accuracy of the solution.

6.1 ML Model Selection

The evaluation of the ML models on the CICD-DoS2019 dataset reveals promising results across various metrics[18]. The Fig. 12 is shown the overall testing and training operation for all models and Table 4 summarizes all model metrics, Logistic Regression (LR) achieves a training accuracy of 99.12% and a testing accuracy of 99.14%, demonstrating its consistency in data classification. Both K-Nearest Neighbors (KNN) and RF models achieve perfect training accuracy of 100%, with testing accuracies of 99.94% and 99.96%, respectively, highlighting their effectiveness in classification tasks. The Decision Tree (DTree) model also performs well, with a training accuracy of 99.99% and a testing accuracy of 99.95%. Linear Discriminant Analysis (LDA) achieves slightly lower accuracies of 98.85% (training) and 98.87% (testing), while Naive Bayes (NB) lags behind, with 96.35% training accuracy and 96.20% testing accuracy. These results underline the superiority of models like KNN and RF in accurately detecting patterns in network data, making them suitable candidates for further use in network security applications.

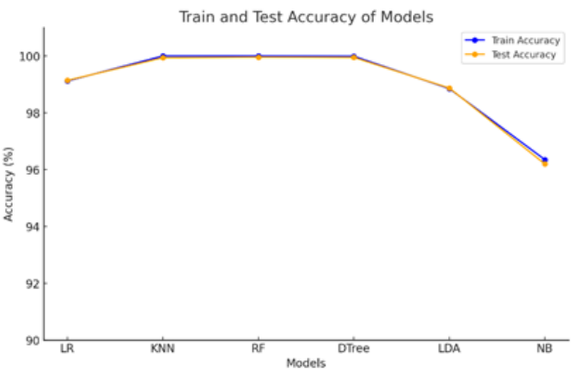


Fig. 12: Comparison accuracy of various ML models

Table 4: Machine learning Models on CICD-DoS2019

Model	Train Acc.	Test Acc.	Train Prec.	Test Prec.	Train Recall	Test Recall
LR	99.12	99.14	99.12	99.14	99.12	99.14
KNN	100	99.94	100	99.94	100	99.94
RF	100	99.96	100	99.96	100	99.96
DTree	99.99	99.95	99.99	99.95	99.99	99.95
LDA	98.85	98.87	98.85	98.87	98.85	98.87
NB	96.35	96.2	96.41	96.27	96.35	96.2

The RF model stands out as the most effective technique among the algorithms evaluated, consistently delivering superior accuracy thanks to its ensemble learning approach and inherent robustness. By constructing multiple decision trees using bootstrapped datasets and employing random feature selection at each split, RF effectively minimizes overfitting while enhancing generalization across diverse datasets. This ensemble strategy aggregates the outputs of weaker learners, resulting in robust and stable predictions, which makes RF particularly resilient to noise and outliers. Moreover, its capability to efficiently handle large, high-dimensional datasets positions it as a

formidable candidate for intrusion detection and anomaly detection tasks. Additionally, the computation of feature importance within RF aids in identifying critical features, improving interpretability without sacrificing accuracy. These qualities render RF an ideal choice for complex classification challenges, especially in network security and IoT applications, where scalability, reliability, and adaptability are essential.

6.2 Performance Evaluation

In this analysis, RF is selected as the primary model for further refinement. The model is trained and tested on a dataset where the network state is categorized into two binary categories: "no attack" and "attack" (with multiple attack types combined into a single category). The dataset is divided into training and testing subsets to rigorously evaluate the model's classification performance.

The performance is measured using accuracy scores, confusion matrices, Receiver Operating Characteristic (ROC) curves, and Area Under the Curve (AUC). Accuracy scores assess the overall effectiveness of the model during training and testing phases. Confusion matrices provide insight into classification errors and help identify misclassified cases, enabling iterative improvements. The ROC/AUC scores evaluate the model's ability to discriminate between attack and non-attack states effectively. Additionally, precision-recall curves analyze the balance between precision and recall, offering deeper insights into the model's performance on imbalanced datasets.

Feature importance analysis is also conducted, particularly for the RF model, to determine the factors most critical to its predictive accuracy. As highlighted in Fig. 13, this analysis aids in refining the model's decision-making process, ensuring optimal performance by focusing on relevant features.

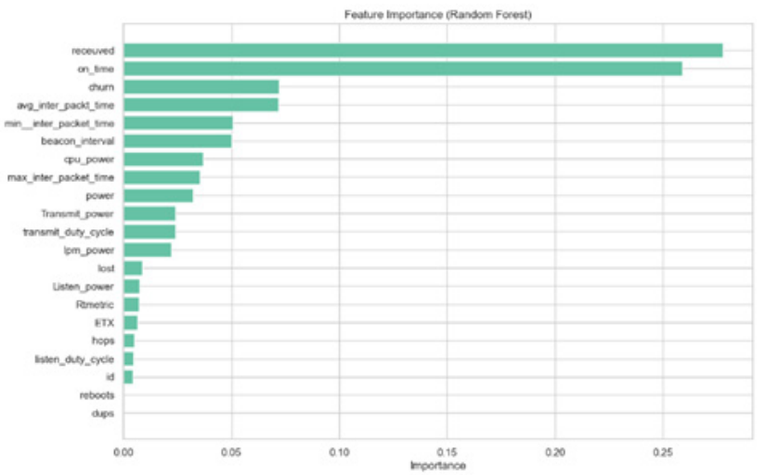


Fig. 13: Feature importance analysis using RF classifier

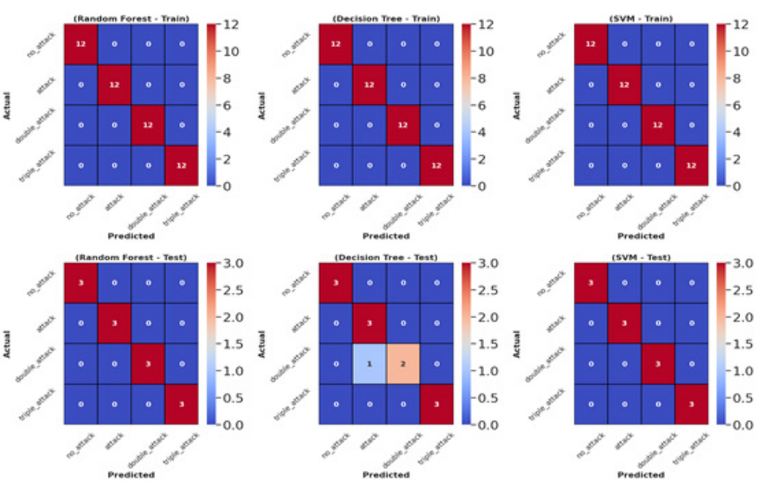


Fig. 14: Confusion matrix for training and testing

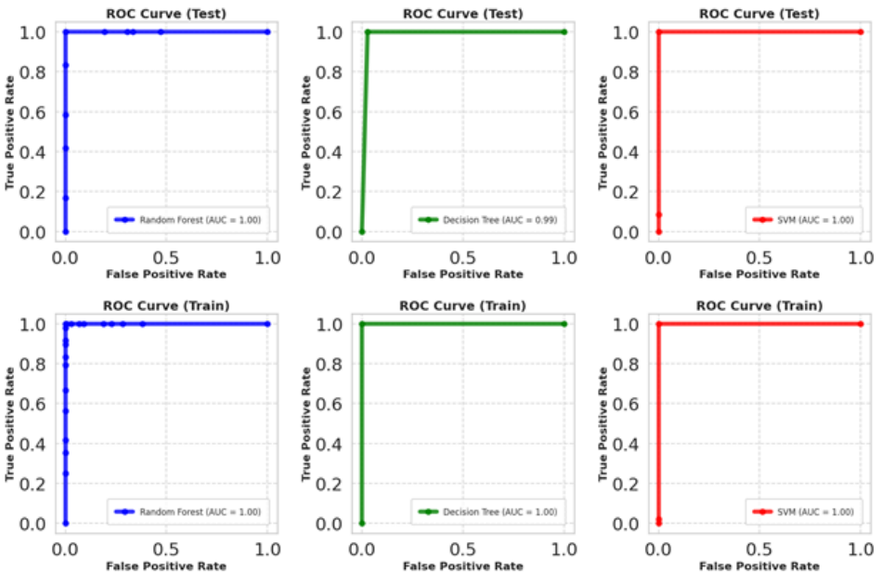


Fig. 15: Receiver operating characteristic testing and training

Table 5: Comparison analysis of proposed work and existing work

Ref.	ML Model	Dataset	Accuracy
[20]	RF	RPL-DoS Dataset	94%
[21]	SVM	CICIDS-2017	91%
[22]	GBM	IoT-IDS Dataset	92%
[23]	CNN	Bot-IoT Dataset	90%
[24]	KNN	CICD-DoS2019	91%
proposed work	LR	CICD-DoS2019	99.14%
	KNN		99.94%
	RF		99.96%
	DTree		99.95%
	LDA		98.87%
	NB		96.20%
proposed work	RF	The proposal dataset	100%
	SVM		99%
	DT		100%

7. FUTURE WORK

While this research has provided valuable insights into VNA detection and mitigation, several areas warrant further exploration:

1. Real-world validation, future studies should implement the proposed methods in real-world IoT deployments to validate their effectiveness under practical conditions, including dynamic network topologies and heterogeneous devices.

2. Advanced attack scenarios, expanding the analysis to include more sophisticated attack types, such as coordinated multi-vector attacks, would provide a broader understanding of network vulnerabilities and enhance the robustness of mitigation strategies.

3. Integration with emerging technologies, investigating the integration of the proposed techniques with blockchain, federated learning, or zero-trust architectures could further improve security and scalability in RPL-based networks.

4. Energy optimization, developing energy-aware machine learning models to minimize computational

overhead while maintaining high detection accuracy is critical for extending the lifetime of IoT devices.

5. Generalization across protocols, extending the study to other IoT routing protocols, such as AODV and DSR, would help generalize the proposed approach and provide a comprehensive framework for securing IoT networks.

8. CONCLUSION

This study provides an in-depth analysis of the VNA, a critical DoS threat targeting the RPL protocol in IoT-based WSNs. The research evaluates the impact of VNAs across single, double, and triple attacker scenarios, highlighting their detrimental effects on network performance metrics such as power consumption, packet loss, and inter-packet time. Results show that the severity of these attacks increases significantly with the number of attackers, leading to degraded network reliability, increased energy consumption, and reduced operational efficiency. The study leverages RF machine learning techniques for feature selection and attack detection, effectively identifying critical network metrics influenced by VNAs. By employing a novel dataset encompassing normal and attack conditions, the proposed approach demonstrates the capability of RF to optimize attack detection and mitigate the adverse effects of VNAs. This contributes to developing robust, scalable, and energy-efficient security solutions for RPL-based IoT networks, particularly in resource-constrained environments.

9. REFERENCES

[1] M. A. Jamshed, K. Ali, Q. H. Abbasi, M. A. Imran, and M. Ur-Rehman, "Challenges, Applications, and Future of Wireless Sensors in Internet of Things: A Review," IEEE Sensors Journal, vol. 22, no. 6, pp. 5482–5494, 2022.

[2] H. Almutairi and N. Zhang, "A Survey on Routing Solutions for Low-Power and Lossy Networks: Toward a Reliable Path-Finding Approach," Network, vol. 4, no. 1, p. 132, 2024.

[3] S. Ghannbari *et al.*, "A New Lightweight Routing Protocol for Internet of Mobile Things Based on Low Power and Lossy Network Using a Fuzzy-Logic Method," Pervasive and Mobile Computing, vol. 97, p. 101872, 2024.

[4] D. E. Boubiche, S. Athmani, S. Boubiche, and H. Toral-Cruz, "Cybersecurity Issues in Wireless Sensor Networks: Current Challenges and Solutions," Wireless Personal Communications, vol. 117, pp. 177–213, 2021.

[5] A. B. de Neira, B. Kantarci, and M. Nogueira, "Distributed Denial of Service Attack Prediction: Challenges, Open Issues and Opportunities," Computer Networks, vol. 222, p. 109553, 2023.

[6] V. K. Verma and S. Sharma, "Investigations on Information Solicitation and Version Number Attacks in Internet of Things," IEEE Sensors Journal, vol. 23, no. 3, pp. 3204–3211, 2023.

[7] H. Sharma, R. Kumar, and M. Gupta, "A Review Paper on Hybrid Cryptographic Algorithms in Cloud Network," in Proc. 2023 2nd International Conference for Innovation in Technology (INOCON), pp. 1–5, 2023.

[8] F. Yu and H. Yin, "A Security Analysis of the Authentication Mechanism of Password Managers," in Proc. 2021 IEEE 21st International Conference on Communication Technology (ICCT), pp. 865–869, 2021.

[9] I. Arshad, S. H. Alsamhi, Y. Qiao, B. Lee, and Y. Ye, "A Novel Framework for Smart Cyber Defence: A Deep-Dive into Deep Learning Attacks and Defences," IEEE Access, vol. 11, pp. 88527–88548, 2023.

[10] A. Aris, S. F. Oktug, and S. B. Ors Yalcin, "RPL Version Number Attacks: In-Depth Study," in Proc. NOMS 2016 - IEEE/IFIP Network Operations and Management Symposium, pp. 776–779, 2016.

[11] G. Sharma, J. Grover, and A. Verma, "Performance Evaluation of Mobile RPL-Based IoT Networks Under Version Number Attack," Computer Communications, vol. 197, pp. 12–22, 2023.

[12] T. A. Al-Amiedy *et al.*, "A Systematic Literature Review on Attacks Defense Mechanisms in RPL-Based 6LoWPAN of Internet of Things," Internet of Things, p. 100741, 2023.

[13] A. Aris and S. F. Oktug, "Analysis of the RPL Version Number Attack with Multiple Attackers," in Proc. 2020 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), p. 18, 2020.

[14] A. Mayzaud *et al.*, "A Study of RPL DODAG Version Attacks," in Proc. Monitoring and Securing Virtualized Networks and Services (AIMS 2014), Lecture Notes in Computer Science, vol. 8508, A. Sperotto *et al.*, Eds., Springer, Berlin, Heidelberg, pp. 62–74, 2014.

[15] D. Vyas and R. Patel, "A Survey: Specific Aspect of the RPL Protocol and Its Enhancements," International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 14s, pp. 294–308, 2024.

[16] J. Granjal, E. de Souza, J. Sá Silva, and P. Thubert, "A Security Threat Analysis for the Routing Protocol for Low-Power and Lossy Networks (RPL)," RFC 7416, Internet Engineering Task Force, 2015. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7416.html.

[17] S. Vastrad and K. R. Shobha, "Power-Efficient Trickle Algorithm with Dynamic Adjustment of Idle Interval for Internet-of-Things Application," IEEE Sensors Journal, vol. 24, no. 3, pp. 3853–3862, Feb. 2024.

[18] Canadian Institute for Cybersecurity (CIC), "DDoS 2019 Dataset," University of New Brunswick, 2019.

[19] N. Saran and N. Kesswani, "A Comparative Study of Supervised Machine Learning Classifiers for Intrusion Detection in Internet of Things," Procedia Computer Science, vol. 218, pp. 2049–2057, 2023.

[20] Y. Zhao *et al.*, "Detection of Version Number Attacks in RPL-Based IoT Networks Using Random Forest," IEEE Transactions on Industrial Informatics, vol. 19, no. 3, 2023.

[21] X. Liu *et al.*, "Machine Learning Approaches for Mitigating DDoS and Version Number Attacks in IoT Networks," IEEE Access, vol. 10, pp. 12345–12356, 2022.

[22] J. García *et al.*, "Gradient Boosting for Security Threats in IoT Networks," IEEE Internet of Things Journal, vol. 8, no. 7, 2021.

[23] Z. Zhou *et al.*, "Deep Learning for Attack Detection in IoT Networks: A CNN-Based Approach," IEEE Transactions on Network and Service Management, vol. 17, no. 4, 2020.

[24] Y. Wang *et al.*, "K-Nearest Neighbors for Network Intrusion Detection in IoT Systems," IEEE Transactions on Network Security, vol. 22, no. 9, 2022.