Securing IoT Networks from DDoS Attacks Using a Redundant IP Pool Strategy

Nuha A. Ismail [1], Ahmad Hani El Fawal [2,*], and Ali Mansour [2]

[1] Information Technology Center, University of Technology, Baghdad, Iraq
[2] Lab-STICC, UMR 6285-CNRS, ENSTA IP Paris, 29806 Brest, France
Email: nuha.a.ismail@uotechnology.edu.iq (N.A.I.); elfawal@ieee.org (A.H.E.F.); mansour@ieee.org (A.M.)
*Corresponding author

***Abstract*—The security of Machine-to-Machine (M2M) communication in the Internet of Things (IoT) is the primary focus of this research article. Despite the increasing reliance on IoT as a cornerstone of future technological advancements, M2M devices face significant challenges, including processing and storage limitations that make them vulnerable to security breaches and Distributed Denial of Service (DDoS) attacks. This paper introduces an innovative security measure against DDoS, called the Redundant IP Pool Strategy (RIPPS). RIPPS utilizes temporary IP pools during an attack to thwart unauthorized access by limiting the time available for infiltration. We conducted a simulation experiment using the OMNeT++ simulator to examine the behavior of IoT networks under DDoS attacks. Our study focuses on traffic management in typical scenarios, including Human-to-Human (H2H) traffic (e.g., Voice over IP and File Transfer Protocol traffic) and M2M traffic. The simulation results demonstrate that implementing RIPPS significantly enhances network stability by reducing packet loss and latency across all traffic scenarios. The RIPPS effectively bolsters the security and resilience of IoT environments against DDoS attacks.**

***Keywords*—Distributed Denial of Service (DDoS), Internet of Things (IoT), OMNeT++, Simu5G, File Transfer Protocol (FTP), Voice over Internet Protoco (VoIP)**

I. Introduction

The Internet of Things (IoT) refers to a worldwide network in which objects are interconnected, allowing physical items to communicate with each other [1, 2]. IoT enables the connectivity of billions of devices, enhancing computing and communication capabilities [3]. IoT technologies facilitate objects to become smart, capable of analyzing data and interacting seamlessly with one another [4]. Embedded sensors in smart objects monitor, sense, and collect extensive data on equipment, the environment, and human social activities [5, 6]. Securing IoT networks from potential Distributed Denial of Service (DDoS) attacks is the main concern of this research article. Processing and communication of this massive amount of data are subject to several constraints, including the limited processing and storage capacity of small IoT nodes [5], which makes them vulnerable to security breaches. These susceptible nodes can turn them into bots or zombies that can launch DDoS attacks, causing serious consequences [7, 8]. For example, in an IoT network, a hacker launches a DDoS attack against Dynamic Host Configuration Protocol (DHCP) devices (also known as a "DHCP starvation attack").

Initially, it is helpful to summarize the approaches for identifying DDoS attacks. Detection strategies can generally be categorized into contemporary techniques. Conventional approaches consist of traffic analysis, which consistently observes volumes, connection requests, and bandwidth used to identify abnormal surges deviating from normal patterns, and signature-based detection employing Intrusion Detection Systems (IDS) such as Snort or Suricata, which depend on established attack signatures. Modern techniques concentrate on anomaly-based detection, which looks for deviations from normal traffic patterns using statistical and behavioral analysis. Machine learning algorithms are often used to improve these techniques.

The coordination and management of IP pools among distributed servers are critical to post-detection mitigation. IP pools are groups of IP addresses assigned to protect services from unavailability, often caused by attacker traffic. Key strategies include centralized synchronization, load balancing to avoid server congestion, DNS redirection for traffic control, automated blacklisting of rogue IPs, and scalability to handle large traffic volumes. This approach enhances resistance to DDoS attacks by managing traffic more effectively and separating attacks.

It may prevent legitimate devices from obtaining Internet Protocol (IP) addresses, thereby compromising their availability. The attacker can assign all IP addresses of the IP pool to himself by broadcasting malicious DHCP discovery messages [9]. Although technological advancements have significantly improved security solutions and provided robust protection in many instances, there is an ongoing need for these solutions to evolve and address emerging security challenges. As IoT networks continue to expand and integrate more devices, the complexity and scale of potential threats also increase [10]. Therefore, it is crucial to develop adaptive security measures that can effectively counteract these evolving

Manuscript received November 15, 2025; revised December 28, 2025; accepted January 13, 2026; published April 24, 2026.

threats, ensuring the continued reliability and safety of IoT systems.

In this paper, we propose a defensive approach to protect IoT devices against DDoS attacks by dynamically altering their IP address pools. We validate the effectiveness of our strategy in mitigating security threats and enhancing IoT communication performance through simulations. We analyzed the impact of DDoS attacks on Quality of Service (QoS) for both Machine-to-Machine (M2M) and Human-to-Human (H2H) traffic, with and without our proposed Redundant IP Pool Strategy (RIPPS). The results demonstrate a significant improvement in packet delivery when our proposed RIPPS solution is used.

In the rest of the paper, we present a comprehensive review of the recent research related to IoT security, DDoS attacks, and various methods employed for detection and prevention. Additionally, our suggested method introduces the RIPPS for detecting and mitigating DDoS attacks in IoT networks. In the simulation part, we evaluate our proposed method using the Objective Modular Network Testbed (OMNeT++) simulation tool. This evaluation aims to assess the effectiveness of our approach in mitigating DDoS attacks. Finally, we conclude our paper by discussing the simulation results and highlighting the effectiveness of our strategy.

II. Related Work

The Internet of Things enables the collection of data from a large number of devices, the analysis and interpretation of that data, and the maintenance and enhancement of equipment across various fields [11].

The inherent diversity of IoT makes it susceptible to many security risks, such as challenges with trust, resource scarcity, and breaches of confidentiality and integrity [12].

As a result, various attacks can target IoT devices, with DDoS attacks becoming more prevalent. DDoS attacks are difficult to detect and aim to impair network servers and resource availability by flooding communication channels from multiple sources using various IoT devices. Consequently, research on mitigating DDoS attacks is rapidly growing [13]. Numerous studies have been conducted in this area, contributing significantly to the field. In this section, we highlight some of these studies.

Iqbal *et al.* [14] presented an overview of Software-Defined Networking (SDN) and thoroughly discussed SDN-based IoT deployment models. While it provided SDN-based IoT security solutions, it did not fully detail or analyze them.

Hamza *et al.* [15] focused on the threats, countermeasures, and requirements for the IoT ecosystem at the network level. They were able to determine and describe the security needs that apply to this level through their analysis. Nevertheless, they did not systematically derive these needs using a security requirement engineering technique. Furthermore, IoT gateways are not expressly covered by their standards.

Hameed *et al.* [16] conducted an extensive assessment of IoT security requirements and problems. They determined that safe routing, privacy, confidentiality, attack detection (including Denial of Service (DoS), DDoS, and insider threats), and robustness and resilience management are the five main security areas that require attention. However, they did not explain how these requirements were selected, nor did they consider the specific particularities of IoT gateways.

The present literature included topics such as threats, vulnerabilities, assaults, and countermeasures [17−21]. The definition and ranking of security needs, particularly for IoT gateways, remained lacking, nevertheless. The selection and configuration of IoT gateways may be supported by identifying pertinent security needs, enhancing the overall security of the IoT system. Furthermore, the existing literature mostly ignores IoT gateway security needs in favor of concentrating on the IoT system as a whole.

A different high-level strategy was proposed under [22]. This approach established a dedicated server in charge of generating and maintaining cryptographically secure keys. Only when this unique server has successfully verified each client's validity may they access the service. In order to obtain a secure key that can be used to process a particular scenario, the client must be successfully authorized on the specific server. Determining the legitimacy of the client is the goal of this scenario. These strategies use application-layer defensive techniques and do not affect the exchange of IP packets. Thus, a malefactor can assault the server via brute force.

Additionally, a variety of lower-level techniques were proposed by the research community. For example, one method divides the data stream between client and server into two successive TCP-level segments and compares the keys of consecutive segments to identify potential unauthorized sources. Data reception is halted if such segments are detected, thereby mitigating the potential impact of an attack [23].

Sengupta *et al.* [24] surveyed to classify attacks based on vulnerability elements, discussing countermeasures and actual attacks. The study also discussed problems with centralized IoT architecture and how blockchain technology can be used to solve these issues in IoT.

Salim *et al*. [25] analyzed server and network attacks with a particular focus on DDoS attacks and their defenses. They emphasized the importance of implementing essential first-line security measures for IoT devices, including password changes, firewalls, firmware updates, and adherence to security standards. In addition, the researchers highlighted the role of government laws as a mechanism to enforce these standards and strengthen overall IoT security, thereby ensuring a more resilient defense against botnet-based DDoS attacks targeting the cloud environment.

A DDoS defense mechanism based on dynamic server IP address changes was introduced in Ref. [26]. In this scheme, only authorized clients are aware of the pseudo-random rule that governs the server's IP transitions. While the method in Ref. [26] shares the same moving-target principle as Mirage—where end hosts periodically change IP addresses to confuse attackers—there are some notable distinctions.

- The victim's IP address changes only when the server is under active DDoS assault.
- The new IP address is assigned for all client sessions concurrently for a considerable amount of time (about five minutes).
- Since an external timestamp is used, precise time synchronization is necessary for the computation of each subsequent IP address.

Various approaches were proposed in Refs. [27−33] to detect and mitigate DDoS attacks in SDN environments using machine learning techniques. These approaches include adaptable architectures, ensemble machine learning methods, and modular frameworks that combine multiple ML algorithms. While achieving high accuracy and detection rates, these solutions often face challenges such as sensitivity to network conditions, limitations to predefined features, increased controller overhead, and high false positive rates for certain traffic flows.

A review of the literature [34−36] revealed that Fifth Generation (5G) Cloudization, particularly in IoT environments, improves DDoS avoidance through the use of various learning techniques. Radio access networks based on fog computing are made possible by cloudization, which increases processing power from the cloud to the access level. This enables IoT devices to employ more intelligent DDoS protection techniques as opposed to the outdated, basic defenses. DDoS prevention systems based on cloudization might be distributed or centralized.

Lounis and Zulkernine [37] surveyed resource-constrained areas of the Internet of Things, focusing on the most widely used short-range wireless communication technologies often targeted by attacks. They examined RFID, ZigBee, Bluetooth, and Wi-Fi technologies, classifying these attacks based on security services. The paper also discussed the limitations of current security measures and proposed countermeasures and procedures to mitigate specific types of attacks.

Wang *et al.* [38] presented an approach based on a two-layer IP, hopping-based Moving Target Defense (MTD) to enhance the security of Mobile Ad-hoc Networks (MANETs). The proposed approach entails continuously and randomly changing device IP addresses or virtual addresses in response to security situations and network requirements, thereby preventing the exposure of IP addresses over the wireless channel. Randomizing IP addresses is performed based on time or specified security events [38]. Although the experiments indicate that designing strategies for IP hopping based on threat levels or network conditions is challenging, on the other hand, static hopping patterns may become predictable for attackers. Our solution utilizes dynamic and adaptive hopping patterns that can be adjusted based on continuous monitoring and analysis of network behavior. This approach prevents predictability and enhances overall security by responding promptly to increasing threats and by adapting to network condition changes.

Several approaches have been proposed to mitigate DDoS attacks in SDN and IoT-Edge environments. One notable technique is IP Fast Hopping, which introduces dynamic address changes to conceal the server's actual IP and reduce the impact of brute-force DDoS attempts. In this method, the server's real IP address is hidden behind multiple "virtual" IPs, with mappings updated every millisecond. This dynamic shuffling separates botnet traffic from legitimate user traffic into different sub-streams, thereby decreasing stress on the network infrastructure during active attacks. Krylov and Kravtsov [39] emphasized that IP Fast Hopping not only conceals communication sessions but also provides a distributed mechanism that effectively distinguishes malicious traffic from legitimate users, thereby enhancing the resilience of network infrastructures against large-scale DDoS attacks.

In an attempt to address the shortcomings of traditional detection methods that rely on fixed signatures, Galadima *et al.* [40] examined the effectiveness of Moving Target Defense (MTD) strategies, specifically through the use of a Frequent Solve Mutation mechanism. The study, which focused on SDN networks, showed that regularly altering the targets' IP addresses is an effective preventative measure that relies on more than just spotting attack patterns. The study claims that because these mechanisms significantly reduce the validity period for targeting a single IP address, they provide a deceptive defense layer against DDoS attacks. This increases the complexity of the attack process and decreases the likelihood that edge devices will become botnets without severely depleting network resources [40].

TABLE I. YSIS OF IP HOPPING TECHNIQUES AND THE PROPOSED RIPPS FRAMEWORK.

Feature	IP Fast Hopping (IPFH)	Temporary Dynamic IP (TDIP)	Our Proposed Work: RIPPS
Primary Mechanism	High-frequency, pseudo-random IP rotation with cryptographic synchronization.	Low-frequency, periodic IP rotation using standard DHCP short lease times.	Temporary activation of redundant IP pools (Pre-assigned rotating addresses).
Operational Timing	Proactive / Always-on.	Proactive / Always-on.	Reactive/Hybrid: Activated specifically during an attack to thwart infiltration.
Resource Overhead	High (synchronization and crypto needed).	Low (Standard DHCP process).	Moderate: Low when idle; activated only when needed, leveraging predefined pools.
Novelty/Focus	Robust resistance for centralized servers.	Low-overhead protection for decentralized IoT endpoints.	Targeted Security: Introduces redundancy pools to limit an attacker's time for infiltration, specifically during DDoS events.

After reviewing the literature, we find that the existing IP hopping protocols establish MTD's effectiveness but

are limited by operational constraints. IPFH is computationally robust but impractical for resource-constrained IoT environments. Conversely, TDIP is low-overhead and practical but lacks the sophisticated, targeted defense capability against advanced attacks and remains always on. Therefore, in IoT environments, designing various approaches that balance solution efficiency with practical application is crucial. These methods should include robust defenses that can significantly mitigate the impact of DDoS attacks, ensuring more resilient and secure IoT networks.

As summarized in Table I, while existing methods provide certain levels of protection, they often suffer from high overhead or a lack of reactivity, which our proposed RIPPS aims to address.

As shown in the comparison, our approach shifts from a proactive synchronization model to a reactive redundancy model, which is a core contribution of this research. This transition leads us to the next section, where the detailed methodology of RIPPS is explained.

III. IP Address Management in IoT Environments

With the proliferation of the Internet and connected devices, connecting the growing number of IoT devices is very important [41]. In this context, the importance of managing IP addresses emerges as one of the key challenges facing IoT environments [42−45].

Each device in an IoT system requires an IP address to communicate over the Internet. With millions of connected devices [46, 47], the traditional IPv4 address system has become inadequate due to its limitation of around 4.3 billion unique addresses [48]. This constraint has accelerated the adoption of IPv6 [49], which provides approximately 3.4 undecillion, i.e., 1036, unique IP addresses, ensuring that each device has a unique identifier [50].

Managing IP addresses in an IoT environment poses many challenges, such as the ability to scale to support millions of devices without performance degradation, security against potential threats, and efficient address allocation in dynamic IoT environments [51]. To address these challenges, effective strategies include dynamic IP address allocation using Dynamic Host Configuration Protocol version 6 (DHCPv6) and the widespread adoption of IPv6 technology [52]. A key process in this dynamic allocation is the interaction between the IoT device and the DHCP server.

When an IoT device connects to a new network, it undergoes a four-message exchange process to obtain an IP address and other network configuration information [53], as shown in Fig. 1:

- Discover message: The IoT device multicasts a search message to all DHCP servers on the local network.
- Offer message: The DHCP server responds with an offer message containing the selected IP address and other network configuration information.
- Request Message: The IoT device accepts the offer and sends a request message to the DHCP server.
- Acknowledgment Message: Finally, the IP allocation process is completed when the server responds with a DHCP Acknowledgment.

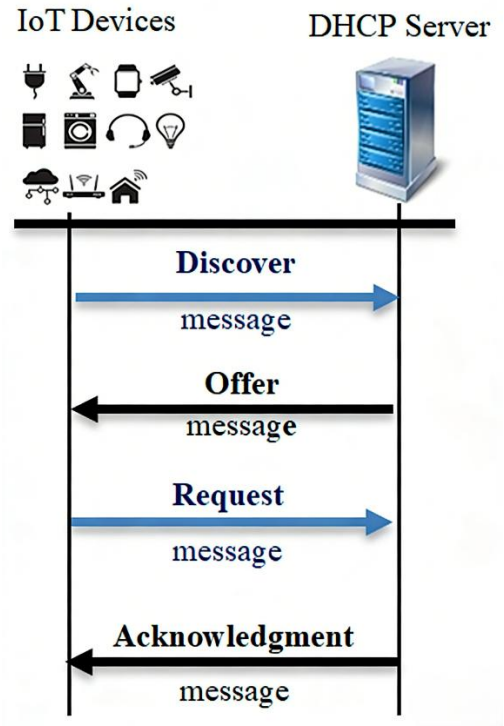


Fig. 1. Leasing an IP address to IoT devices.

Following this process, IP addresses can be effectively utilized as IoT devices join and leave the network, thanks to this dynamic allocation.

The key element of our approach involves configuring the DHCP with a traditional pool, which optimizes IP address allocation and streamlines network management. By dynamically allocating IP addresses from redundant pools during DDoS attacks, we aim to disrupt attackers' ability to gain access to the network and ensure enhanced security measures during critical periods.

IV. Redundant IP Pool Strategy (RIPPS)

A. Attack Detection and Triggering Model

In this work, RIPPS is designed as a specialized response and mitigation mechanism. Consequently, the focus is on the efficient handling of attack traffic rather than the internal design of a novel detection algorithm. We deploy an external DDoS detection module at the network edge, which continuously monitors traffic and raises an alarm when an attack is detected.

Leveraging established high-performance detectors—including entropy-based schemes, statistical detectors, and machine learning-based systems—we configure our system triggers based on proven benchmarks, operating under standard parameters where detection accuracy exceeds 97–99% and the False Positive Rate (FPR) is less than 1%. The detection latency is modeled to be within the sub-second to few-second range, ensuring compatibility with real-time operational constraints [54–60].

RIPPS is triggered immediately once the detector raises an alert. While detection delay dictates the initial reaction time, the low FPR ensures that even if RIPPS is activated for benign traffic, the impact is negligible due to the system's low operational overhead. This decoupling of detection and mitigation confirms the practical feasibility of the proposed system.

The Redundant IP Pool Strategy (RIPPS) is implemented as a complete end-to-end cycle that enhances IoT network resilience against DDoS attacks. In the normal phase, devices operate under a standard DHCP configuration with a stable IP pool and long lease times, supporting routine communication. Once an attack is detected, RIPPS automatically transitions to redundant IP pools with shortened lease times, thereby limiting the attacker's penetration window and reducing the risk of service disruption. During the monitoring phase, if the attack persists, the system dynamically rotates through successive redundant pools, ensuring continuous protection and maintaining network integrity. Finally, in the post-attack phase, RIPPS restores the DHCP server to its original configuration, resuming normal operations while preserving a secure environment. This end-to-end process—from normal operation, through attack mitigation, continuous monitoring, and recovery—provides a comprehensive and practical defense mechanism against DDoS threats in IoT networks. RIPPS is shown in the flowchart in Fig. 2.

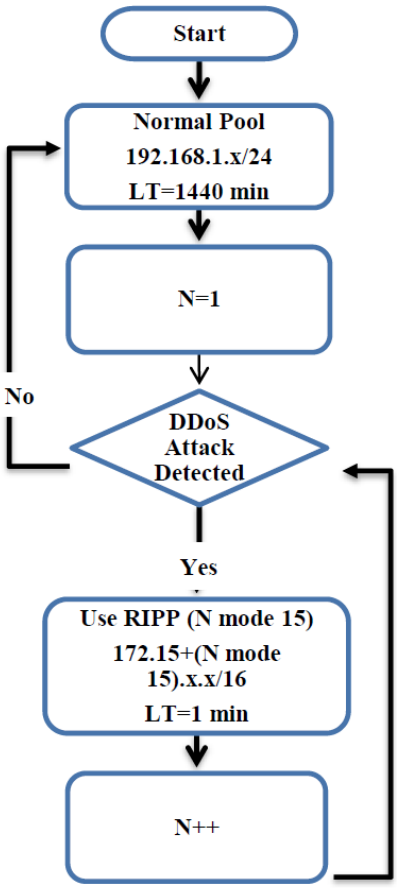


Fig. 2. RIPPS.

B. Normal Cycle Phase (prior DDoS attacks)

- During the normal cycle, we configure the DHCP server with a standard IP pool (192.168.1.x/24) and a Lease Time (LT) of one day (1440 minutes). The DHCP server assigns IP addresses to all network-connected devices.

C. Attack Phase

Upon detecting a DDoS attack, our system assumes a Penetration Time (PT) of 2 minutes. This assumption is based on typical attack scenarios reported in prior security analyses, in which adversaries require at least 120 seconds to penetrate a system under normal conditions successfully. By setting PT to 2 minutes, we establish a conservative threshold that allows us to evaluate RIPPS's responsiveness. The comparison between Latency Time (LT) and Penetration Time (PT) is critical:

- If LT < PT, the system can detect and respond before the attack succeeds.
- If LT ≥ PT, the system fails to prevent penetration. Thus, the assumption provides a clear benchmark to validate RIPPS's effectiveness. PT is the time for an attacker to gain access to the network resources. Consider an LT < PT (e.g., LT = 1 minute). Based on this assumption, the RIPPS automatically switches the system to the first redundant IP pool (RIPPS 1: 172.16.x.x/16) after one minute, allocating new IP addresses with a one-minute lease time to all connected devices. This strategy prevented the attacker from disrupting the network by utilizing temporary IP pools during an attack, thereby thwarting unauthorized access by limiting the time available for infiltration, ensuring enhanced security, and maintaining network integrity during the attack.

D. Monitoring Phase

If the attack continues beyond two minutes, our system activates the next line of defense using the second backup pool, RIPPS (2): 172.17.x.x/16. This involves dynamically allocating new IP addresses with a one-minute lease time to all connected devices. This measure ensures continuous protection against the ongoing attack.

It is essential to continuously monitor the network during the attack. If the attack continues, the RIPPS keeps changing the IP address range following the pools from RIPPS (3) to RIPPS (15); each pool utilizes an LT = 1 minute.

E. Post-attack Phase

Once the attack period ends, the RIPPS forces the DHCP server to resume its normal cycle operations with a pool (192.168.1.x/24) and a lease time of 1440 minutes. This ensures that the network resumes its standard operations and maintains a secure post-attack environment.

V. Simulation

We used the open-source simulator OMNeT++, integrated with the INET framework and Simu5G, to evaluate the proposed system. This powerful combination provides an ideal environment for developing, testing, and

investigating novel techniques under both conventional and unorthodox conditions [61–65].

We created three different scenarios for our simulation: an attack scenario, a defense/RIPPS scenario, and a normal cycle scenario. As shown in Table II, each scenario includes 20 devices producing different kinds of H2H and M2M traffic.

TABLE II. DISTRIBUTION OF DEVICES IN DIFFERENT CATEGORIES (VOICE OVER IP TRAFFIC "VOIP," FILE TRANSFER PROTOCOL "FTP")

Type	H2H		M2M	
	FTP	VoIP	Normal	Bots
Normal Cycle Scenario	4	4	12	0
Attack Scenario	4	4	6	6
Defense/RIPPS Scenario	4	4	11	1

In all scenarios, we used the following types of traffic for each category:

1) H2H

- File Transfer Protocol (FTP) traffic: Sends data to the server at a rate of 536 bytes every 53.6 ms.
- Voice over IP (VoIP) traffic: Sends data to the server at a rate of 40 bytes every 20 ms.

2) M2M

- Normal: Normal M2M traffic sends data to the server at a rate of 128 bytes every 1000 ms.
- Bots: Bots (controlled by the attacker) send data to the server at a rate of 256 bytes every 2 ms.

Other traffic parameters are summarized in Table III.

TABLE III. PACKET SIZE OF EACH TYPE OF TRAFFIC

Traffic Type		Packet Size
H2H	FTP	536 bytes every 53.6 ms
	VoIP	40 bytes every 20 ms
M2M	Normal	128 bytes every 1000 ms
	Bots	256 Bytes/every 2 ms

All scenarios have a simulation duration of 200 seconds and a channel bandwidth of 3 Mbps.

A. Normal Scenario

We used the results of this scenario as a benchmark to evaluate the performance of other cases. In this scenario, we used 4 FTP devices, 4 VoIP devices, and 12 normal M2M devices, as depicted in Fig. 3.

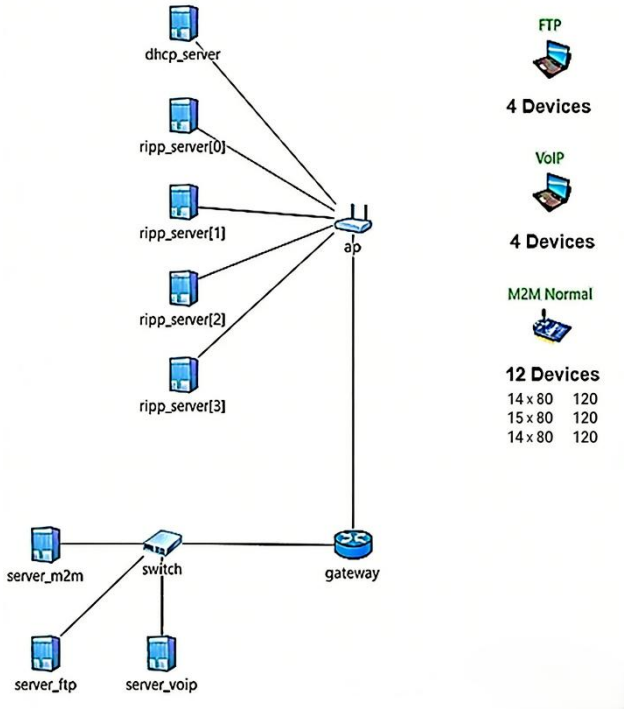


Fig. 3. Normal scenario.

B. Attack Scenario

Assumed in this scenario is that half of the accessible devices, 6 conventional M2M devices, are taken over by an attacker. Next, as illustrated in Fig. 4, the attacker deluges the network with Internet Control Message Protocol (ICMP) packets.

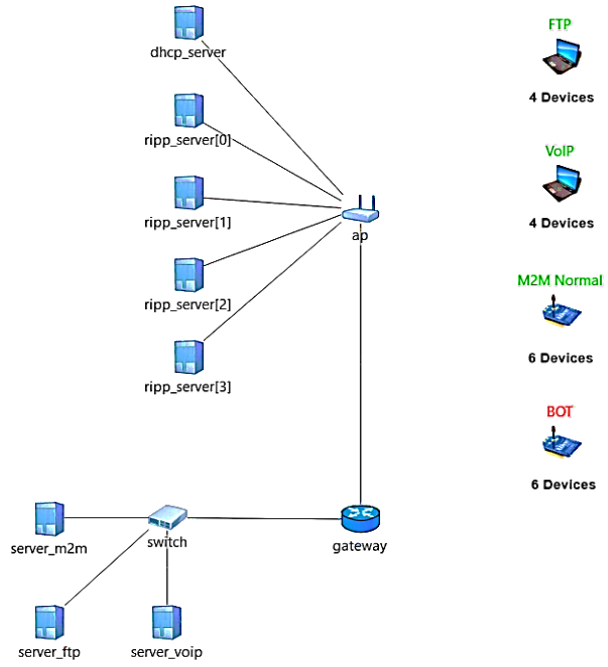


Fig. 4. Attack scenario.

C. Defense/RIPPS scenario

In this scenario, we implement our RIPPS defensive solutions strategy. Once in place, this strategy reduces the attacker's chance of successfully taking over one of our 12 normal M2M devices to just 8.3%, as shown in Fig. 5.

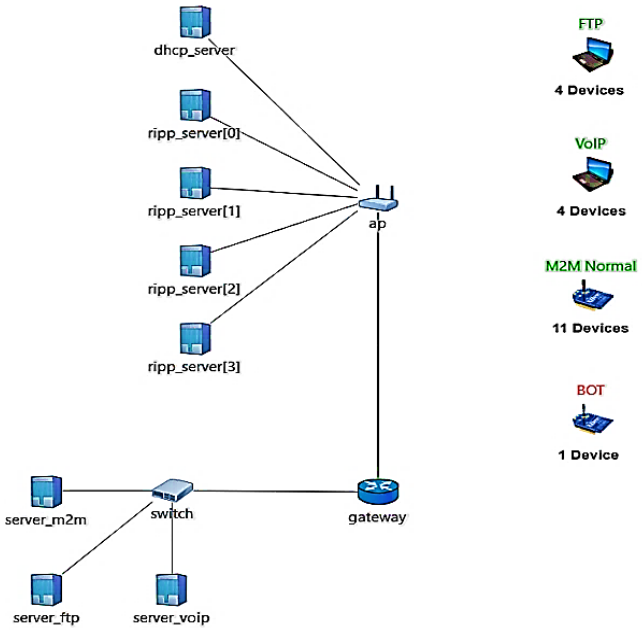


Fig. 5. Defense/RIPPS scenario.

To assess the efficacy of the RIPPS defense mechanism in thwarting network attacks and enhancing network performance, we specifically monitor three critical metrics: Round-Trip-Time (RTT), Packet Loss Rate (PLR), and Channel Utilization (CU).

VI. Results and Discussions

A. Channel Utilization (CU)

Fig. 6 displays the percentage of total channel utilization for various scenarios over a given time period.

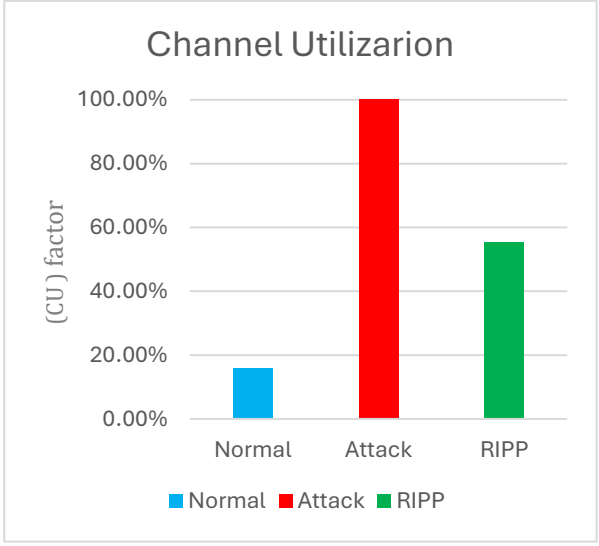


Fig. 6. CU in different scenarios.

- The CU is measured to be 15.73% in the normal scenario, indicating moderate network activity.
- The CU is observed to reach 100% during the attack scenario, which is taken as a definite sign of extreme congestion caused by the flooding attack.
- Compared to the Attack scenario, the CU decreases by 44.3% to 55.3% in the Defense/RIPPS scenario. This reduction underscores the effectiveness of the RIPPS protection strategy in mitigating the attack's impact, leading to enhanced network efficiency and reduced congestion.

B. Packet Loss Rate (PLR)

Analysis of the simulation metrics reveals the following results:

1) Packets sent

H2H and M2M devices transmit the same total number of packets under all conditions: Defense/RIPPS, Attack, and Normal:

- FTP traffic: 15124 packets.
- VoIP traffic: 25936 packets.
- M2M traffic: 2400 packets.

2) Packets Received

The total number of packets that the server has received varies greatly depending on the scenario:

a) Normal scenario

- FTP traffic: The server received all 15,124 transmitted packets.
- VoIP traffic: The server received all 25,936 transmitted packets.
- M2M traffic: The server received all 2,400 transmitted packets.

b) Attack scenario

- FTP Traffic: Of the 15,124 packets sent, the server received only 786.
- VoIP Traffic: Of the 25,936 packets sent, the server received only 2,807.
- M2M Traffic: Of the 2,400 packets sent, the server received only 36.

c) Defense/RIPPS scenario

- FTP traffic: The server received 14,459 of the 15,124 packets transmitted.
- VoIP traffic: The server received all 25,936 packets sent.
- M2M traffic: The server received 2,189 of the 2,400 packets transmitted.

3) PLR matrix

We compute the PLR by calculating the percentage of lost packets relative to the total number of packets delivered. We use the following Eq. (1) to calculate the PLR:

$$PLR = \frac{Packets\ Sent - Packets\ Received}{Packets\ Sent} \times 100\% \quad (1)$$

In Fig. 7, the findings demonstrate how network attacks affect packet delivery and how well the RIPPS protection mechanism reduces packet loss.

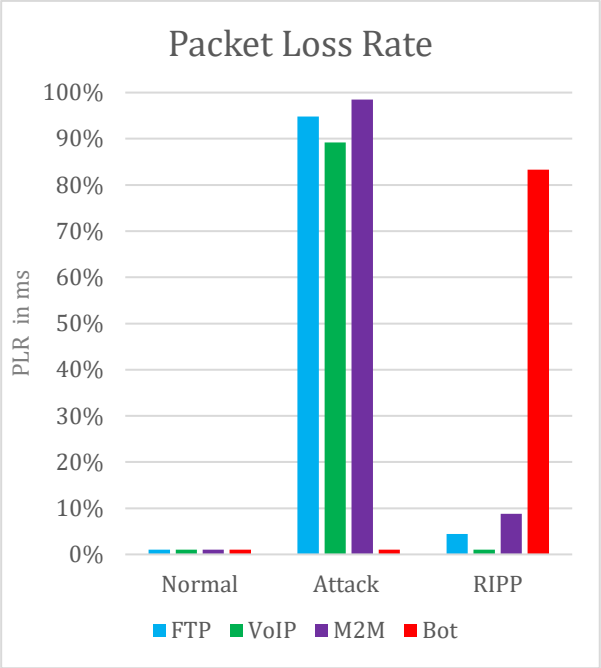
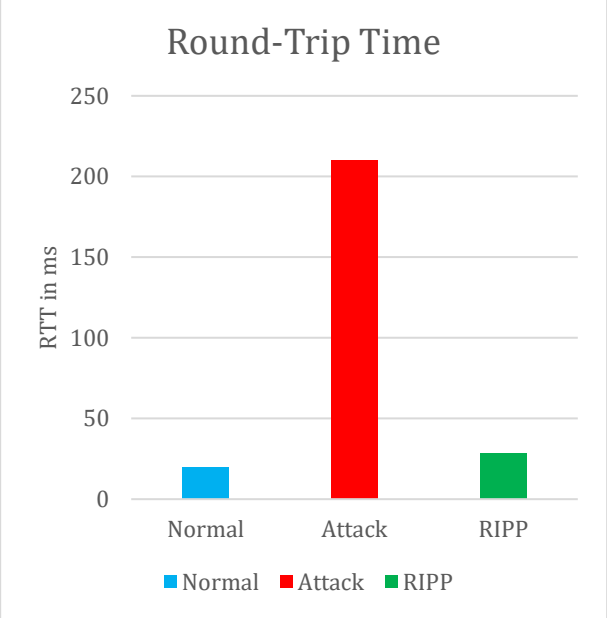


Fig. 7. PLR in different scenarios.



- Fig. 8. Round-trip time in different scenarios.

- In the normal scenario, all devices (FTP, VoIP, and M2M) have 0% PLR, indicating normal operation.
- In the attack scenario, FTP devices experience a 94.803% PLR, VoIP devices experience an 89.178% PLR, and M2M devices experience a 98.5% PLR. This shows that the flooding attack is causing a high rate of packet loss, with VoIP experiencing lower PLR because of its priority over other traffic.
- In the Defense/RIPPS scenario, the PLR for FTP devices drops to 4.397% (a reduction of 95.36%), for VoIP devices drops to 0% (a reduction of 100%), and for M2M devices drops to 8.792% (a reduction of 91.08%). This indicates a successful mitigation of the attack.

C. Round-Trip Time (RTT) in Different Scenarios

RTT measures the duration in seconds for a network packet to travel from a starting point to a destination and back to the starting point [66]. The mean RTT measurements, shown in Fig. 8, provide insights into the latency experienced by network packets during transmission.

TABLE IV. BENCHMARKING RIPPS RESULTS AGAINST EXISTING LITERATURE

Metric	IP Fast Hopping (IPFH) [39]	Temporary Dynamic IP (TDIP) [26]	Our Proposed RIPPS
Mitigation Strategy	High-frequency millisecond hopping	Periodic (5-min) lease updates	Reactive, targeted pool rotation
Operational Mode	Proactive / Always-on	Proactive / Always-on	Reactive (Activated during attack)
Latency (RTT)	Moderate impact due to sync overhead	Low impact	86.4% reduction from attack levels
Packet Loss (PLR)	Effective for server defense	Limited against rapid bursts	>90% reduction (0% for VoIP)
Resource Usage	High (Sync/Crypto needed)	Low (Standard DHCP)	Moderate (Low when idle)

- In a normal scenario, the mean RTT is 20 ms, indicating reliable and low-latency communication between devices and the server.
- In the attack scenario, the mean RTT increases to approximately 210 ms, reflecting significant delays in the packet arrival. This surge is indicative of how the network flooding attack affected network performance, leading to higher latency and hiccups in communication.
- We see a decrease in the RTT mean from 210 ms to 28.5 ms upon engaging the RIPPS defense mechanism, indicating an 86.4% reduction in RTT. This notable decrease illustrates how well the RIPPS defensive mechanism works to lower latency and return the network to regular operation.
- Reduction of the Channel Utilization (CU) factor in the RIPPS scenario (decreasing from 100% in the situation of an attack to around 55.3%) functions as the prime measure of successful mitigation of the DDoS attack. In the IoT setup, a value of 100% for the CU factor represents the condition ‘channel exhaustion,’ whereby the malicious M2M and H2H floods occupy the total available bandwidth.
- Through dynamic IP pool rotation, RIPPS works to ‘filter’ the traffic at the level of the network layer. The attacker still proceeds to transmit the flooded IP addresses that have expired from the previous pool to the system, which means that the harmful traffic is discarded at the gateway or server before it occupies

the internal resources of the communication channel. It is the lowering of CU that is not only an improvement in performance but is the technology that precludes total service blackout and the availability of communication.

- As shown in Table IV, RIPPS is contrasted with other Moving Target Defense (MTD) strategies, including IP Fast Hopping (IPFH) and Temporary Dynamic IP (TDIP).

VII. Conclusion

In this paper, we introduce the RIPPS as an innovative approach to enhancing network security and safeguarding IoT networks against threats from DDoS attacks. Our strategy involves dynamically leasing IP addresses from backup pools during DDoS attacks to counteract attackers and bolster security during critical periods. We provide a thorough analysis of various scenarios involving H2H and M2M communications within an IoT environment, utilizing the OMNeT++ Simulator with the INET framework and Simu5g to simulate Normal, Attack, and Defense/RIPPS scenarios.

The results of our simulations showed that the RIPPS defense mechanism is effective. In the normal scenario, the network metrics for packet loss rate, channel utilization, and round-trip time were stable and efficient. In contrast, the attack scenario caused severe network congestion, high packet loss, and increased latency. Applying the RIPPS defense mechanism greatly improved these conditions: Channel Utilization (CU) dropped by 44.3%, Packet Loss Rate (PLR) was reduced by over 90% for a different type of traffic, and Round-Trip Time (RTT) was reduced by 86.4%.

These findings underscore the importance of robust defense mechanisms in IoT networks to ensure reliable and secure communications.

In our future work, we will extend our research to check the effectiveness of RIPPS over IPv6 standards to secure them from potential DDoS attacks. Moreover, we will integrate our defense mechanism with other detection algorithms for improved accuracy and evaluate the impact of false alarms on network performance. Finally, we will explore the scalability of the RIPPS defense mechanism in larger IoT networks and investigate its integration with blockchains and artificial intelligence technologies to enhance security and efficiency.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Nuha A. Ismail conceived the research idea, conducted the literature review, designed the simulation scenarios, and performed the experimental evaluations; Ahmad Hani El Fawal contributed to the system modeling, methodology design, analysis and interpretation of the simulation results, and critically revised the manuscript for technical depth and clarity; Ali Mansour supervised the research work, provided technical guidance and validation of the proposed approach, and contributed to the overall structure and refinement of the manuscript; all authors had reviewed, approved, and agreed to the final version of the manuscript.

Acknowledgment

The authors express their gratitude to William Alameh of Modern University for Business and Science (MUBS) in Beirut, Lebanon, for his invaluable assistance in organizing, scripting, and executing several scenarios using OMNET++ and INET modelers.

References

[1] R. A. Mouha, "Internet of Things (IoT)," *Journal of Data* Analysis *and Information Processing*, vol. 9, no. 2, pp. 1–10, 2021. https://doi.org/10.4236/jdaip.2021.92006

[2] R. Hassan *et al*., "Internet of things and its applications: A comprehensive survey," *Symmetry*, vol. 12, no. 10, 1674, 2020. https://doi.org/0.3390/sym12101674

[3] K. Doshi *et al*., "Timely detection and mitigation of stealthy DDoS attacks via IoT networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2164–2176, 2021. https://doi.org/10.1109/TDSC.2021.3049942

[4] H. A. Abdulghan *et al*., "Analysis on security and privacy guidelines: RFID-based IoT applications," *IEEE Access*, vol. 10, pp. 131528–131554, 2022. https://doi.org/10.1109/ACCESS.2022.3227449

[5] I. U. Din *et al*., "Trust management techniques for the internet of things: A survey," *IEEE Access*, vol. 7, pp. 29763–29787, 2019.

[6] A. Agrahari *et al*., "Marvelous hand: An IoT-enabled artificial intelligence-based human-centric biosensor design for consumer personal security application," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 1063–1070, 2024.

[7] R. Yaegashi, D. Hisano, and Y. Nakayama, "Lightweight DDoS mitigation at network edge with limited resources," in *Proc. 2021, IEEE 18th Annual Consumer Communications and Networking Conference (CCNC)*, 2021, pp. 1–6.

[8] Y. Yilmaz and S. Buyrukoglu, "Development and evaluation of ensemble learning models for detection of DDoS attacks in IoT," *Hittite Journal of Science and Engineering*, vol. 9, no. 2, 2022.

[9] R. K. Yadav and K. Karamveer, "A survey on IoT botnets and their detection approaches," in *Proc. 2022 4th International Conference on Advances in Computing, Communication, Control and Networking,* 2022, pp. 1901–1906.

[10] S. Khanam *et al*., "A survey of security challenges, attacks taxonomy, and advanced countermeasures in the internet of things," *IEEE Access*, vol. 8, pp. 219709–219743, 2020. https://doi.org/10.1109/ACCESS.2020.3037359

[11] A. A. Y. Khan *et al*., "Internet of Things (IoT) security with blockchain technology: A state-of-the-art review," *IEEE Access*, vol. 10, pp. 122679–122695, 2020.

[12] L. Wei *et al*., "Trust management for internet of things: A comprehensive study," *IEEE Internet of Things Journal*, vol. 9, no. 10, pp. 7664–7679, 2022.

[13] P. Kumari and A. K. Jain, "A comprehensive study of DDoS attacks over IoT networks and their countermeasures," *Computers and Security*, vol. 127, 2023.

[14] W. Iqbal *et al*., "An in-depth analysis of IoT security requirements, challenges, and their countermeasures via software-defined security," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10250–10276, 2020. https://doi.org/10.1109/JIOT.2020.2997651

[15] A. Hamza, H. H. Gharakheili, and V. Sivaraman, "IoT network security: Requirements, threats, and countermeasures," arXiv preprint arXiv:2008.09339, 2020.

[16] S. Hameed, F. I. Khan, and B. Hameed, "Understanding security requirements and challenges in Internet of Things (IoT): A review," *Journal of Computer Networks and Communications*, vol. 1, 2019. https://doi.org/10.1155/2019/9629381

[17] P. Aufner, "The IoT security gap: A look down into the valley between threat models and their implementation," *International Journal of Information Security*, vol. 19, no. 4, pp. 1–18. https://doi.org/10.1007/s10207-019-00445-y

[18] I. Butun, P. Österberg, and H. Song, "Security of the internet of things: Vulnerabilities, attacks, and countermeasures," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 1, pp. 616–644, 2020. https://doi.org/10.1109/COMST.2019.2953364

[19] F. Mneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "IoT: Internet of threats? A survey of practical security vulnerabilities in real IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8182–8201, 2019. https://doi.org/10.1109/JIOT.2019.2935189

[20] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019. https://doi.org/10.1109/ACCESS.2019.2924045

[21] N. Neshenko *et al.*, "Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look at internet-scale IoT exploitations," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 3, pp. 2702–2733, 2019. https://doi.org/10.1109/COMST.2019.2910750

[22] A. K. Iyengar, M. Srivatsa, and J. Yin, "Protecting against denial-of-service attacks using trust, quality of service, personalization, and hiding port messages," U.S. Patent Application No. 20100235632 A1, U.S. Patent and Trademark Office, 2010.

[23] D. R. Marquardt, P. A. Paranjape, and P. S. Patil, "Securing a communication protocol against attacks," U.S. Patent Application No. 20110283367 A1, U.S. Patent and Trademark Office, 2011.

[24] J. Sengupta, S. Ruj, and S. D. Bit, "A comprehensive survey on attacks, security issues, and blockchain solutions for IoT and IioT," *Journal of Network and Computer Applications*, vol. 149, 2020. https://doi.org/10.1016/j.jnca.2019.102481

[25] M. M. Salim, S. Rathore, and J. H. Park, "Distributed denial of service attacks and their defenses in IoT: A survey," *The Journal of Supercomputing*, vol. 7, no. 7, pp. 5320–5363, 2020. https://doi.org/10.1007/s11227-019-02945-z

[26] P. Mittal, D. Kim, Y. C. Hu, and M. Caesar, "Mirage: Towards deployable DDoS defense for web applications," arXiv Preprint arXiv:1110.1060, 2011.

[27] A. A. Alashhab *et al*., "Enhancing DDoS attack detection and mitigation in SDN using an ensemble online machine learning model," *IEEE Access*, vol. 12, pp. 51630–51649, 2024. https://doi.org/10.1109/ACCESS.2024.3384398

[28] M. A. Ribeiro, M. S. P. Fonseca, and J. Santi, "Detecting and mitigating DDoS attacks with moving target defense approach based on automated flow classification in SDN networks," *Computers and Security*, vol. 134, 2023. https://doi.org/10.1016/j.cose.2023.103462

[29] Ö. Tonkal *et al*., "Machine learning approach equipped with neighbourhood component analysis for DDoS attack detection in software-defined networking," *Electronics*, vol. 10, no. 11, 2021. https://doi.org/10.3390/electronics10111227

[30] J. A. P. Díaz *et al*., "A flexible SDN-based architecture for identifying and mitigating low-rate DDoS attacks using machine learning," *IEEE Access*, vol. 8, pp. 155859–155872, 2020. https://doi.org/10.1109/ACCESS.2020.3019330

[31] R. Khamkar *et al*., "Low-rate DDoS attack identification and defense using SDN based on machine learning method," *International Research Journal of Engineering and Technology*, vol. 8, no. 3, pp. 423–429, 2021.

[32] K. M. Sudar and P. Deepalakshmi, "Flow-based detection and mitigation of low-rate DDoS attack in SDN environment using machine learning techniques," *IoT and Analytics for Sensor Networks*, pp. 193–205, 2022.

[33] A. Ali and M. M. Yousaf, "Novel three-tier intrusion detection and prevention system in software-defined network," *IEEE Access*, vol. 8, pp. 109662–109676, 2020. https://doi.org/10.1109/ACCESS.2020.3004644

[34] I. Ahmad *et al*., "Security for 5G and beyond," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3682–3722, 2019. https://doi.org/10.1109/COMST.2019.2916180

[35] N. Chaabouni *et al*., "Network intrusion detection for IoT security based on learning techniques," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019. https://doi.org/10.1109/COMST.2019.2896380

[36] N. N. Dao *et al*., "Mobile claudication storytelling: Current issues from an optimization perspective," *IEEE Internet Computing*, vol. 24, no. 1, pp. 39–47, 2020. https://doi.org/10.1109/MIC.2020.2965485

[37] K. Lounis and M. Zulkernine, "Attacks and defenses in short-range wireless technologies for IoT," *IEEE Access*, vol. 8, pp. 88892–88932, 2020. https://doi.org/10.1109/ACCESS.2020.2993553

[38] P. Wang, M. Zhou, and Z. Ding, "A two-layer IP hopping-based moving target defense approach to enhancing the security of mobile ad-hoc networks," *Sensors*, vol. 21, no. 7, pp. 1–18, 2021. https://doi.org/10.3390/s21072355.

[39] V. Krylov and K. Kravtsov, "DDoS attack and interception resistance, IP fast hopping-based protocol," arXiv preprint arXiv:1403.7371, 2014.

[40] H. Galadima, A. Seeam, and V. Ramsurrun, "Cyber deception against DDoS attacks using moving target defense framework in SDN IoT-EDGE networks," in *Proc. 2022 3rd International Conference on Next Generation Computing Applications (NextComp)*, 2022, pp. 1–6.

[41] C. Jayapal, P. Sultana, M. N. Saroja, and J. Senthil, "Security protocols for IoT," *Ubiquitous Computing and Computing Security of IoT*, pp. 1–28, 2018. https://doi.org/10.1007/978-3-030-01566-4_1

[42] L. Tightiz and H. Yang, A comprehensive review of IoT protocols' features in smart grid communication," *Energies*, vol. 13, no. 11, 2020. https://doi.org/10.3390/en13112762.

[43] A. M. Konsta, A. L. Lafuente, and N. Dragoni, "A survey of trust management for the internet of things," *IEEE Access*, vol. 11, pp. 122175–122204, 2023.

[44] Y. Liu *et al*., "A survey on blockchain-based trust management for the internet of things," *IEEE Internet of Things Journal*, vol. 10, no. 7, pp. 5898–5922, 2023.

[45] H. M. J. Almohri, L. T. Watson, and D. Evans, "Predictability of IP address allocations for cloud computing platforms," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 500–511, 2020. https://doi.org/10.1109/TIFS.2019.2924555

[46] S. Manimozhi and J. G. Jayanthi, "A literature survey on transition mechanisms in IPv4 and IPv6 networks," in *Proc. 2020, 4th International Conference on Intelligent Computing and Control Systems*, 2020, pp. 12–18.

[47] B. Kumar, "IPv6 addressing strategy for IoT network: A comprehensive review," in *Proc. 2023 International Conference on Sustainable Emerging Innovations in Engineering and Technology,* 2023, pp. 738–744. https://doi.org/10.1109/ICSEIET58677.2023.10303477

[48] M. T. Hossain and M. N. Rahman, "A review on IPv4 and IPv6: A comprehensive survey," Available at SSRN 5768923, 2023.

[49] G. Kumar and P. Tomar, "IPv6 addressing scheme with a secured duplicate address detection," *IETE Journal of Research*, vol. 68, no. 5, pp. 3371–3378, 2020.

[50] N. Galego, R. M. Pascoal, and P. R. Brandão, "IPv6 in IoT," in *Proc. International Conference on Management, Tourism and Technologies*, 2023, pp. 89–94. https://doi.org/10.1007/978-3-031-44131-8_9

[51] M. Tariq, S. Ahmad, and H. V. Poor, "Dynamic resource allocation in IoT enhanced by digital twins and intelligent reflecting surfaces," *IEEE Internet of Things Journal*, vol. 11, no. 16, 2024. https://doi.org/10.1109/JIOT.2024.3398413

[52] A. A. Ani *et al*., "Authentication and privacy approach for DHCPv6," *IEEE Access*, vol. 7, pp. 73144–73156, 2019. https://doi.org/10.1109/ACCESS.2019.2919966

[53] L. Li, G. Ren, Y. Liu, and J. Wu, "Secure DHCPv6 mechanism for DHCPv6 security and privacy protection," *Tsinghua Science and Technology*, vol. 23, no. 1, pp. 13–21, 2018. https://doi.org/10.26599/TST.2018.9010020

[54] L. Tsobdjou, S. Pierre, and A. Quintero, "An online entropy-based DDoS flooding attack detection system with dynamic threshold," *IEEE Transactions on Network and Service Management*, vol. 19, pp. 1679–1689, 2022. https://doi.org/10.1109/tnsm.2022.3142254

[55] N. Pandey and P. Mishra, "Performance analysis of entropy variation-based detection of DDoS attacks in IoT," *Internet Things*, vol. 23, 100812, 2023. https://doi.org/10.1016/j.iot.2023.100812

[56] M. Nooribakhsh and M. Mollamotalebi, "A review on statistical approaches for anomaly detection in DDoS attacks," *Information Security Journal: A Global Perspective*, vol. 29, 2022. https://doi.org/10.1080/19393555.2020.1717019

[57] M. S. Neto *et al*., "DDoS attack detection in SDN: Enhancing entropy − Based detection with machine learning," *Concurrency and Computation: Practice and Experience*, vol. 36, 2024. https://doi.org/10.1002/cpe.8021
[58] C. Liu and S. Zhong, "DDoS attack detection method based on machine learning," in *Proc. 2024 IEEE 15th International Conference on Software Engineering and Service Science*, 2024, pp. 83−87.
[59] T. Ali, Y. Chong, and S. Manickam, "Machine learning techniques to detect a DDoS attack in SDN: A systematic review," *Applied Sciences*, vol. 13, no. 5, 3183, 2023. https://doi.org/10.3390/app13053183
[60] Y. Liu *et al*., "IGED: Towards intelligent DDoS detection model using improved generalized entropy and DNN," *Computers, Materials and Continua*, vol. 80, no. 2, 2024.
[61] The OMNeT++ Community. (2024). OMNeT++: Discrete Event Simulation in C++ (Version 6.0)]. [Online]. Available: https://omnetpp.org/download/
[62] P. A. B. Bautista *et al*., "Large-scale simulations manager tool for OMNeT++: Expediting simulations and post-processing analysis," *IEEE Access*, vol. 8, pp. 159291–159306, 2020. https://doi.org/10.1109/ACCESS.2020.3020745
[63] OMNeT++ INET Framework. (2024). Download INET Framework. [Online]. Available: https://inet.omnetpp.org/index.html
[64] G. Nardini *et al*., "Simu5G—An OMNeT++ Library for end-to-end performance evaluation of 5G networks," *IEEE Access*, vol. 8, pp. 181176–181191, 2020.
[65] Simu5G: Simulator for 5G new radio networks. (2024). [Online]. Available: https://simu5g.org/
[66] New Mirai variant targeting network security devices. [Online]. Available: https://unit42.paloaltonetworks.com/mirai-variant-iot-vulnerabilities/