Enhanced Mechanism for Detecting RA Flooding DDoS Attacks in IPv6 Networks

Navaneethan C. Arjuman1, 2, a), Muhamad Syamil Imran Bin Mohd Mansor2, b), Zar Chi Hlaing2, c) and Mohamed Elnour Abdelhafez Fadul3, d)

*1Center for Cyber Security and Quantum Computing, CoE for Advanced Cloud, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Selangor*

*2Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia*

*3UST Global Malaysia, Level 2, Mini Circuit Building 3, Bayan Lepas Free Industrial Zone Phase 4, 11900 Bayan Lepas, Pulau Pinang*

*a)Corresponding author: navaneethan@mmu.edu.my*

*b)1221303708@student.mmu.edu.my*

*c) zar.chi.hlaing@student.mmu.edu.my*

*d) wdelnour@gmail.com*

**Abstract.** The adoption of Internet Protocol version 6 (IPv6) has become important with the growing number of Internet users. This transition has introduced challenges in network security. Distributed Denial-of-Service (DDoS) is one of the most widely performed attacks against IPv6 networks. The vulnerability lies in the Neighbor Discovery Protocol (NDP) mechanism. This protocol can be used by attackers to disrupt NDP packets,particularly Router Advertisement (RA) packets,which fill up system resources and causes severe interruptions in the network. Previously,several detection mechanisms have been developed to detect RA DDoS attacks. Although most of them have been shown to work, many of these detection mechanisms are complex and resource intensive.Machine learning and deep learning-based detection require large datasets and computational power. This can limit the implementation in real-life scenarios, especially narrowband networks such as in Internet of Things (IoT) networks. To overcome this challenge, this study has suggested a lightweight threshold-based traffic anomaly detection mechanism for the detection of RA flooding attacks. The mechanism monitors important traffic characteristics, which are RA packet rates and source Internet Protocol (IP) address behavioral conditions in which to recognize anomalous activities. By setting up a testbed network environment, the study would really test and refine the mechanism in a controlled space. This technique provides a realistic and cost-effective way of protecting IPv6 networks, particularly in a low-bandwidth IoT network. The results indicated high detection accuracy levels at high traffic rates. The average Precision, Recall, and F1 score were 0.89, 0.84, and 0.83, respectively. Enhanced mechanisms effectively demonstrate the system’s ability to detect RA flooding DDoS attacks.

# INTRODUCTION

The Internet developed rapidly over the past few decades, which in turn produced a substantial increase in the number of devices that access the Internet. Rapid growth in the Internet has depleted the supply of Internet Protocol version 4 (IPv4) addresses designed with 32-bit addressing [1]. With 128-bit addressing, IPv6 resolves the address limit by creating an essentially inexhaustible range of unique IP addresses [2, 3]. The benefit of larger address ranges within IPv6 includes more efficient routing capabilities along with automatic configuration abilities [2, 3].

Within IPv6 networks, NDP performs three vital functions by allowing device detection, router discovery, and peer availability testing [4]. NDP functions as the replacement for Address Resolution Protocol (ARP) in IPv4 networks while working as an integral part of ICMPv6 that manages device-to-device network status and error messages [4].

The convenience of communication between devices introduced by NDP produces new security threats that affect network protection [5]. There are no authentication features within the protocol, which allows Denial of Service (DoS) and DDoS service attacks on NDP implementations [5]. Attackers can carry out popular NDP attacks by generating various fake RA packets [6]. Such malicious alterations performed by attackers lead to exhausted system resources and degraded network performance, resulting in a complete system shutdown [6].

Different scholars have developed machine learning and deep learning detection techniques to identify these cyber attacks [7]. Identification techniques operate with precision, but require substantial processing capacity and large dataset capacities and need extended run times [8]. Detection methods are difficult to utilize in real-time operations when resources are limited. [7, 8].

In this research, a lightweight threshold-based detector is developed to detect RA flooding anomalies in IPv6 net- works. Compared to the ML-based approaches, this enhanced system minimizes power consumption, replacing com- plex computations with efficient threshold analysis, making it most suitable for IoT networks that depend on batteries. This system analyzes IPv6 traffic and incoming RA packets to determine anomalous behavior. Security alerts are generated when the RA packet rate goes beyond the preconfigured threshold values. The results show that this proposed enhanced mechanism is a suitable mechanism to detect RA flooding in resource-constrained IPv6 networks. The research question is how effective is the existing RA flooding detection mechanism in identifying attacks in IPv6 networks? Does a threshold-based solution offer better deployability in a resource-constrained environment than current solutions? The research objective is to analyze the current detection solutions to IPv6 RA flooding attacks, develop an enhanced threshold-based anomaly detection system and confirm and test its functionality in a controlled IPv6 network in terms of the appropriateness of low-resource environments.

# RELATED WORK

This section reviews recent research on NDP detection and prevention mechanisms, which other researchers developed over recent years.

The authors Najjar et al. [9] conducted research on how NDP-based attacks launched using Neighbor Solicitation (NS) and RA packets flooding and the impact on the network resources. The authors demonstrated through their testbed that these attacks made the network unusable because they consumed excessive Central Processing Unit (CPU) and memory resources. While revealing facts about assault conduct, the research study failed to present any method to either recognize or stop NDP.

Al-Ani et al. [10] introduced Neighbor Discovery Protocol Security (NDPsec) as a system that strengthens NDP message security through lightweight Ed25519 digital signature implementation. It blocks spoofing attempts against fake RAs. While the network operation and processing duration of NDPsec performed faster than traditional Secure Neighbor Discovery (SeND), the mechanism’s effectiveness requires pre-loaded public keys throughout networks. This distribution of key requirements introduces a single point of failure in large-scale deployment. In the actual IPv6 networks, preloading the public keys poses great challenges, especially for dynamic IoT environments. Although it is efficient against known attacks, it shows limited ability to find newly developing or emerging attacks.

Bahashwan et al. [5] developed a flow-based detection system that implements entropy measurement to trace irregular behavior in NDP traffic. This detection system examines specific patterns related to the IP address message frequency and validates the observations against predefined limits. Traffic with unpredictable behavior over multiple rounds will lead to suspicious status. The method proved effective for quick detection on rapid networks but showed unfavorable results because of its rigid threshold criteria, particularly when detecting flooding attacks.This rigidity results in great false alarm probabilities, restricting its use in the real-world networks with adaptive traffic loads.

According to Elejla et al. [11], deep learning serves as the detection framework for ICMPv6-based flooding at- tacks. The study evaluated three models starting from Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) through synthetic data analysis. LSTM gave the best results, with high accuracy and a low false positive rates. However, these deep learning models require significant computing power, which limits their use in real-time environments or systems with limited resources. As shown in Table 1, existing solutions for NDP attacks suffer from either high costs or ineffective detection capabilities.

**TABLE 1.** Critical review summary of mechanisms proposed in the literature and their limitations

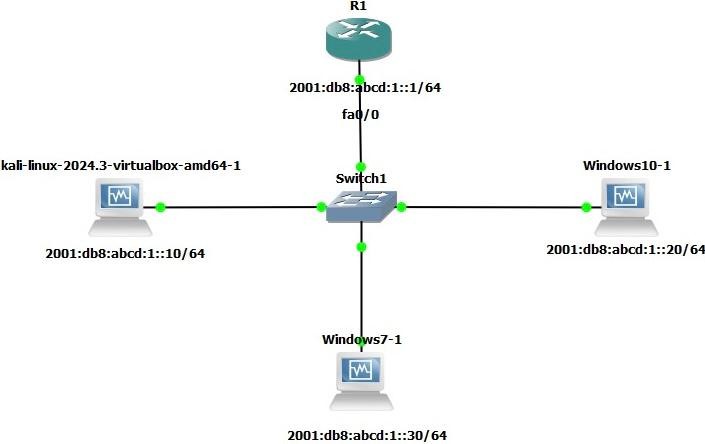
|  |  |  |
| --- | --- | --- |
| **Authors** | **Proposed Mechanisms** | **Limitations** |
| Najjar et al. [9] | Simulation-based analysis of NDP flooding attacks | No detection or prevention solution proposed |
| Al-Ani et al. [10] | NDPsec (uses Ed25519 for NDP authentication) | Relies on pre-distributed keys; only prevents, doesn’t detect evolving threats |
| Bahashwan et al. [5] | Entropy-based flow analysis for NDP anomalies | Fixed thresholds lead to false positives during high traffic |
| Elejla et al. [11] | Deep-learning IDS (RNN, LSTM, GRU) | Computationally expensive; requires large datasets |

# METHODOLOGY

This section presents the design and implementation of the proposed threshold-based anomaly detection mechanism to identify RA flooding attacks in IPv6 networks. It includes the system architecture, data collection and analysis strategy, threshold definition, and the detection logic using Python-based automation.

## System Architecture and Testbed

The system is deployed entirely at the host level, requiring no changes to the network infrastructure. This is a network testbed-based implementation that consists of an attacker node, two victim nodes, and a router. The attacker node uses the The Hacker’s Choice–IPv6 Attack (THC-IPv6) toolkit to launch RA flooding attacks, while the detection system operates on a victim node using Python script based on Scapy. Figure 1 shows a test environment that was used to perform RA flooding attacks on an IPv6 network. It consisted of a router, two victim computers (Windows 7 and Windows 10), and an attacker computer running Kali Linux and the THC-IPv6 tool. Normal traffic and attack traffic were transmitted, and the packets were captured using Wireshark on the victim computer in .pcap format to be analyzed. Table 2 presents all primary system elements with their specific functionalities.



**FIGURE 1.** Network topology of the IPv6 network

**TABLE 2.** System modules and their functionalities

|  |  |
| --- | --- |
| Module | Description |
| Traffic Capture Module | Captures live IPv6 traffic using Wireshark and exports .pcap files. |
| Threshold-Based Detection Unit | Extracts RA packet features and applies threshold comparison using Python based on Scapy. |
| Alert & Logging System | Flags anomalies, logs events, and optionally issues an alert. |

## Data Collection and Feature Extraction

To evaluate the detection mechanism, traffic data was collected from an IPv6 network under both normal and attack conditions. In the first phase, normal traffic was captured without any malicious activity to establish a baseline for typical RA behavior. In the second phase, RA flooding attacks were generated using the flood\_router6 tool from the THC-IPv6 toolkit on a Kali Linux attacker node. All network traffic was captured using Wireshark and exported in .pcap format. The captured data was then filtered to isolate RA packets by applying the filter for icmpv6.type=134, ensuring that only relevant RA packets were retained for further analysis.

From the filtered packet capture files, three key traffic features were extracted to support anomaly detection. These include: (i) RA packet rate — the number of RA packets received per second, which helps detect abnormal increases in RA traffic; (ii) source IP frequency — the number of RA packets sent by individual source IPs, useful for identifying potential spoofing or excessive use by a single host; and (iii) burst pattern behavior — detection of sudden spikes in traffic volume, indicative of flooding attempts. Table 3 shows that the critical features are obtained from RA traffic analytics, through which researchers identify anomalous behavior related to flooding attacks. These features formed the basis for threshold comparison in the detection algorithm, allowing the system to distinguish between normal and anomalous traffic patterns effectively. These metrics highlight anomalies indicative of flooding of rogue RA packets.

**TABLE 3.** Extracted features from RA traffic

|  |  |
| --- | --- |
| Feature | Description |
| Packet Rate | Number of RA packets per second. |
| Source IP Frequency | Number of RA packets from the same IP address. |
| Burst Patterns | Detection of sudden spikes in RA traffic volume. |

## Threshold Definition

The core detection logic is based on comparing observed values to a statistically defined threshold. The threshold is defined for every traffic feature as in Equation (1):

where, *µ*: the mean value of the feature under normal traffic conditions.

*C*: a constant margin introduced to accommodate natural variations in network behavior, determined empiri- cally during initial testing.

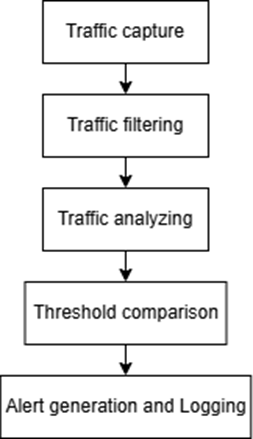
As an example, if the average RA packet rate during normal activity is 20 packets per second and the chosen constant margin *C* is 10, then the threshold becomes 30 packets per second. Any observed RA packet rate exceeding this value is flagged as anomalous. The value of *C* is fine-tuned through experimental evaluation to minimize both false positives and false negatives, ensuring accurate and reliable detection performance.

## Detection Logic and Automation

A Python script based on Scapy operates on Wireshark.pcap outputs and direct traffic monitoring through sniff() for automated detection. The logic consists of

1. Loading and parsing RA packets.
2. Observing RA packets by establishing specific time periods.
3. Calculating source IP frequency.
4. Executing threshold assessment procedures on different values.
5. Alert generation if thresholds reach their maximum levels.

Figure 2 presents the step-by-step procedure of the suggested detection system that involves collecting network traffic and then creating alerts before assigning them to log files. This modular deployment system reveals lightweight versions of the system that remain suitable for real-time applications.



**FIGURE 2.** Flow diagram for detection process of the suggested detection system

# TESTING & RESULTS DISCUSSION

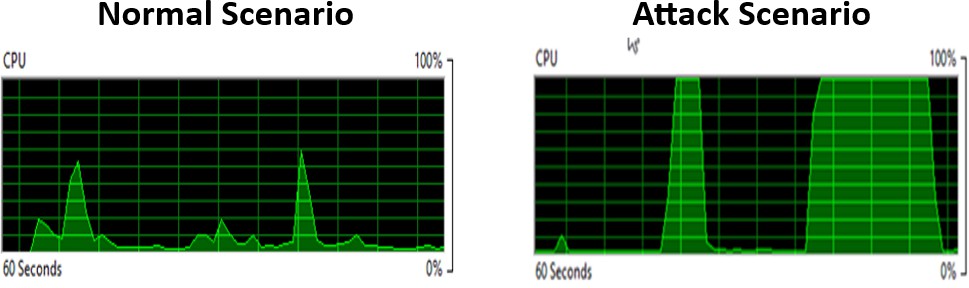
This section presents the results from testing the threshold-based anomaly detection mechanism for RA flooding attacks across an IPv6 environment would be presented here. Tests using both normal and attack traffic enabled evaluation of detection performance and the measures of Precision, Recall, and F1-Score.

## Experimental Setup

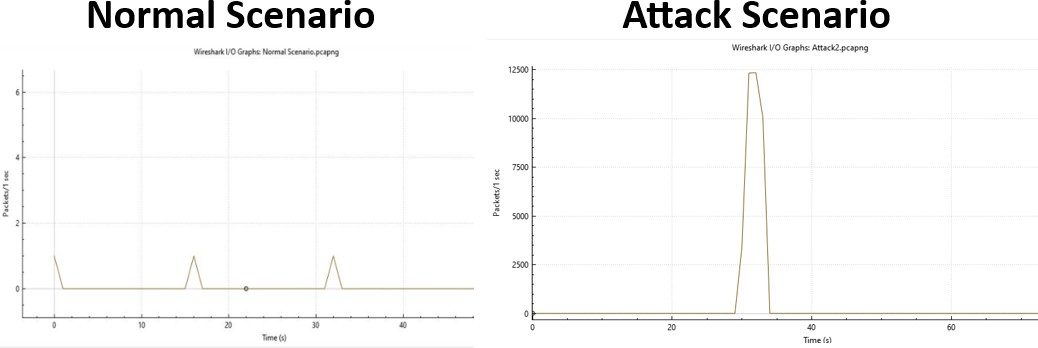
The detection tests were run under established isolated network environment. An IPv6 network topology included three main nodes: a Kali Linux attacker node and two victim nodes running Windows 7 and Windows 10, all connected through a router. The attacker executed RA flooding attacks using the flood\_router6 application from the THC- IPv6 toolkit. The detection process involved capturing network traffic using Wireshark and filtering it to obtain RA packets (ICMPv6 Type 134). These packets were processed through a custom Python script developed using the Scapy library. The script analyzed the packet rate and source IP frequency, then applied a statistically defined threshold to detect anomalous traffic patterns.

## Result Analysis

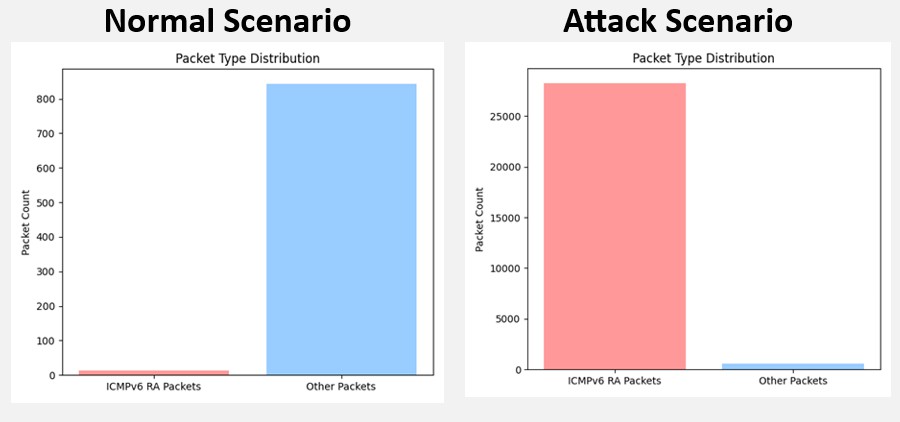
To further validate the detection mechanism, visual representations of key metrics were analyzed during normal and attack scenarios. Figure 3 shows the comparison of the CPU utilization during regular network activity and when under an RA flooding attack. Under a normal scenario, the CPU usage is constant with some moderate variations, whereas the RA flooding attack leads to sustained high usage. The comparison shows clearly the extreme effects of the attack on the system resources. Figure 4 demonstrates the I/O graph of RA packet rates (packets per second) in normal condition and during an RA flooding attack. The packet rate stays low in the normal scenario graph, whereas in the attack scenario graph, there is a sudden sharp increase to over 12000 packets per second. Such a dramatic rise proves the disruptive nature of the RA flooding attack by overloading the victim’s network interfaces and interfering with regular network operations. Figure 5 illustrates the bar graph that shows the distribution of packet types captured during both normal network operation and under RA flooding attack. In the normal scenario, there are very few ICMPv6 RA packets mixed with other traffic types, whereas in the attack scenario, more than 20,000 ICMPv6 RA packets are observed flooding other traffic. Such excessive concentration of RA packets proves the success of the flooding attack that leads to network congestion and service disruption.



**FIGURE 3.** CPU usage: attack vs. normal scenario



**FIGURE 4.** I/O traffic: attack vs. normal scenario



**FIGURE 5.** ICMPv6 packet types: attack vs. normal scenario

## Detection Accuracy

A total of twenty detection experiments were carried out to analyze precision under normal and RA flooding situations. The analytical data evaluation relied on the combination of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) evaluation systems. Table 4 has shown the formulas for computation of Precision, Recall and F1-Score.

**TABLE 4.** Formulas of performance metrics used in the detection evaluation

|  |  |  |
| --- | --- | --- |
| Metric | Formula | Description |
| Precision |  | Higher precision means fewer false positives. |
| Recall |  | Higher recall means fewer false negatives. |
| F1-Score |  | High F1-Score indicates a well-balanced performance. |

The developed system was tested both under normal traffic and RA flooding attack scenarios for various traffic levels. Table 5 shows all the results from the outcome of testing in both scenarios.

**TABLE 5.** Results summary across 20 test runs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Traffic Type** | **Level** | **TP** | **FP** | **FN** | **Precision** | **Recall** | **F1-Score** |
|  | Low | 8 | 0 | 12 | 1.00 | 0.40 | 0.57 |
| Normal Traffic | Medium | 13 | 0 | 7 | 1.00 | 0.65 | 0.79 |
|  | High | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
|  | Low | 20 | 12 | 0 | 0.625 | 1.00 | 0.77 |
| RA Flooding | Medium | 20 | 7 | 0 | 0.74 | 1.00 | 0.85 |
|  | High | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
|  |  | **Average** |  |  | **0.89** | **0.84** | **0.83** |

**TP** = True Positive, **FP** = False Positive, **FN** = False Negative

Under the normal traffic scenario, the high-level traffics the system shows that the results were perfect for Precision, Recall and F1-Score. But for low and medium traffic, there were several FNs detected. In this scenario, the system may struggle to distinguish normal traffic from anomalies when traffic volume is low or moderate, leading to missed detections (FN). This could be due to threshold tuning. If the detection threshold is set too high to avoid false positives (FP), it might ignore legitimate anomalies in lighter traffic. Also, it could be due to features used for detection (e.g., packet rate, request frequency) that may not be distinctive enough at lower traffic levels.

Under the RA flooding attack traffic scenario, the high-level traffic shows that the results were also perfect for Precision, Recall and F1-Score. But for low and medium traffic there were several FPs detected. This could be because the system might misinterpret benign bursts of traffic (e.g., temporary spikes) as RA flooding attacks when traffic is not consistently high. Without adaptive baselines for different traffic levels, the system could flag normal variability as malicious. Overall, the above system is effectively able to distinguish the malicious activities from legitimate activities in a reliable manner.

# CONCLUSION

In conclusion, a transition to IPv6 becomes security-wise challenging because of NDP vulnerabilities. The security risks associated with RA flooding network attacks remain high because these attacks diminish network availabil- ity while draining resources. Although current detection techniques using deep learning models and cryptography maintain security, implementation remains challenging due to the need for high processing power and stable network conditions.

This research developed automatic thresholding-based detection methods for identifying RA flooding attacks in IPv6 networks. This system uses traffic characteristics from RA packets along with source IP frequencies to determine abnormal activity and achieves outstanding results.

The proposed detection system uses lightweight, interpretable, adjustable detection logic that provides an opera- tional solution in comparison to resource-heavy machine learning models, especially when used in narrowband IoT networks.

# REFERENCES

1. L. E. Hughes, “The depletion of the IPv4 address space,” in *Third Generation Internet Revealed: Reinventing Computer Networks with IPv6* (Springer, 2022), pp. 119–146.
2. G. Ordabayeva, M. Othman, B. Kirgizbayeva, Z. D. Iztaev, and A. Bayegizova, “A systematic review of transition from IPv4 to IPv6,” in *Proceedings of the 6th International Conference on Engineering & MIS* (2020), pp. 1–15.
3. K. Igulu, F. Onuodu, and T. P. Singh, “IPv6: Strengths and limitations,” in *Communication Technologies and Security Challenges in IoT: Present and Future* (Springer, 2024), pp. 147–172.
4. R. M. Saad, M. Anbar, S. Manickam, S. I. Shaheen, and I. H. Hasbullah, “Neighbor discovery protocol anomaly-based detection system using neural network algorithm,” International Journal of Information Security **23**, 1787–1803 (2024).
5. A. A. Bahashwan, M. Anbar, and S. M. Hanshi, “Overview of IPv6-based DDoS and DoS attacks detection mechanisms,” in *International Conference on Advances in Cyber Security* (Springer, 2019), pp. 153–167.
6. A. A. Bahashwan, M. Anbar, I. H. Hasbullah, Z. R. Alashhab, and A. Bin-Salem, “Flow-based approach to detect abnormal behavior in neighbor discovery protocol (NDP),” IEEE Access **9**, 45512–45526 (2021).
7. A. Salih, S. T. Zeebaree, S. Ameen, A. Alkhyyat, and H. M. Shukur, “A survey on the role of artificial intelligence, machine learning and deep learning for cybersecurity attack detection,” in *Proceedings of the 2021 7th International Engineering Conference: Research & Innovation amid Global Pandemic (IEC)* (IEEE, 2021), pp. 61–66.
8. R. Geetha and T. Thilagam, “A review on the effectiveness of machine learning and deep learning algorithms for cyber security,” Archives of Computational Methods in Engineering **28**, 2861–2879 (2021).
9. F. Najjar, Q. Bsoul, and H. Al-Refai, “An analysis of neighbor discovery protocol attacks,” Computers **12**, 125 (2023).
10. A. Al-Ani, A. K. Al-Ani, S. A. Laghari, S. Manickam, K. W. Lai, and K. Hasikin, “NDPSec: Neighbor discovery protocol security mechanism,” IEEE Access **10**, 83650–83663 (2022).
11. O. E. Elejla, M. Anbar, S. Hamouda, S. Faisal, A. A. Bahashwan, and I. H. Hasbullah, “Deep-learning-based approach to detect ICMPv6 flooding DDoS attacks on IPv6 networks,” Applied Sciences **12**, 6150 (2022).
12. J. S. Nixon and M. Amenu, “Investigating security issues and preventive mechanisms in IPv6 deployment,” International Journal of Advanced Engineering and Nano Technology **9**, 1–20 (2022).