Improved Fragmentation Attacks Mitigation in IPv6 Networks

Navaneethan C. Arjuman,1, 2, a) Vimalrich Selvam Nadarajah,2, b) Shankar Karuppayah,3, c) and Selvakumar Manickam3, d)

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

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

*3Cybersecurity Research Centre (CYRES), Universiti Sains Malaysia, 11800 USM Penang, Malaysia*

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

*b) 1211104171@student.mmu.edu.my*

*c)kshankar@usm.my*

*d)selva@usm.my*

**Abstract.** The impending depletion of IPv4 addresses has left everyone worried about the future of networks worldwide, leading to the need for global adoption of IPv6 network infrastructure. Despite progressing research in the widespread adoption of IPv6 networks, an insufficiently addressed flaw is the opportunity for attackers to manipulate fragments for malicious purposes. This includes risks such as denial-of-service attacks and evasion of security mechanisms during reassembly. Existing security protocols, such as IPSec, are not specifically designed to mitigate these threats, leaving a gap in IPv6’s overall security framework. This study proposes the “Dynamic Fragmentation Policy Mechanism" (DFPM), as a solution to attacks imposed on fragmented packets in IPv6 networks. DFPM utilizes a machine learning algorithm that studies packet behavior to determine the likelihood of an attack. Upon confirmation of an attack, rules are dynamically added to the host’s firewall to drop malicious traffic from the attacker. The key benefit of this mechanism is that it is implemented at the host, causing minimal changes in terms of network latency and system performance, whilst also allowing for ease of integration without a need for a change of network infrastructure. The effectiveness of DFPM was evaluated against ICMPv6 Packet Too Big (PTB) error floods, the Reassembly Timeout attack, and Normal traffic under low, medium, and high traffic intensities. The results showed high detection accuracy across these traffic types, with average Precision, Recall, and F1-Scores reaching 0.96, 0.94, and 0.94, respectively. DFPM’s strong performance against attacks underscores its potential as a robust mitigation strategy for fragmentation-based threats in IPv6 networks.

# INTRODUCTION

With IPv4 slowly being phased out due to its outdated protocols and a diminishing pool of addresses, IPv6 has proven itself as the solution to IPv4 limitations by offering a larger address space, improved routing efficiency and improved end-to-end communication [1]. However, despite its numerous advantages, IPv6 remains vulnerable to its own set of security threats, particularly those associated with fragmentation and reassembly processes. Unlike IPv4, IPv6 inhibits intermediate routers from fragmenting packets. Instead, fragmentation is handled exclusively by the source node. When a packet exceeds the Maximum Transmission Unit (MTU) of the path, the source node generates smaller fragments using the fragmentation header, which are then reassembled at the destination [2]. Although innovative, this process introduces vulnerabilities during reassembly, such as resource exhaustion, ambiguous fragment handling, and evasion of security systems, which attackers can exploit to perform attacks such as Denial-of-Service (DoS), packet injection, and extension header manipulation.

The complexity that IPv6 extension headers add into the equation of this process further exacerbates the issue, as their flexible nature provides a fertile ground for exploitation. For example, attackers can create fragmented packets with improperly ordered or oversized extension headers to overwhelm intermediate devices and cause processing inefficiencies or security breaches [3]. Although IPsec’s Authentication Header (AH) and Encapsulating Security Payload (ESP) provide robust security features, they do not fully mitigate fragmentation-based attacks. As highlighted in RFC 4301, an attacker could send a non-initial fragment with a forged source address that, if bypassed, could overlap with IPsec-protected traffic from the same source, thereby compromising the integrity of the protected traffic [4].

Despite the existence of various mitigation mechanisms, many current defenses suffer from limitations in adapt- ability, efficiency, and practicality in the real world. Techniques relying on static rule sets or specialized hardware often fall short when faced with evolving threat landscapes or are impractical due to high costs and compatibility issues, particularly for organizations with legacy IT infrastructure. Moreover, there is a struggle to strike a balance between security and computational overhead imposed on systems. These gaps underscore the need for a lightweight, host-level detection and prevention system to mitigate fragmentation attacks in IPv6 environments.

This paper proposes the Dynamic Fragmentation Policy Mechanism (DFPM), a machine learning-powered defense system designed to monitor IPv6 network activity and mitigate fragmentation-related attacks. The scope of this re- search focuses on two denial-of-service attack methods, namely the ICMPv6 PTB flood and the Reassembly Timeout attack. In the first, attackers spoof the victim’s IP address and send forged ICMPv6 messages to routers or endpoints. These messages falsely signal that the victim’s packets exceed the Path MTU, leading the victim host to reduce their MTU settings. Over time, this disrupts legitimate communication by causing excessive fragmentation or completely stopping transmission. In the second, the attacker sends incomplete or intentionally delayed fragments to the victim [5]. The victim host retains these fragments in memory, waiting for the remaining pieces, which never arrive. The buildup of such incomplete fragment sets ultimately results in legitimate packets being dropped because fragment buffers are being filled with incomplete fragments. DFPM leverages a supervised decision tree model to detect such behaviors in real time, dynamically updating firewall rules to block malicious sources, ensuring high performance, and seamless integration with existing host systems. Although this study focuses on these two attack types, DFPM’s design is extensible and adaptable for broader IPv6 security use cases.

This paper presents a review of existing fragmentation attack mitigation techniques, details the methodology of the Dynamic Fragmentation Policy Mechanism, analyzes its effectiveness through experimental results, and concludes with key findings and future research directions.

# RELATED WORK

This section briefly discusses some of the related work in fragmentation attack mitigation solutions devised by other researchers in the past, including a critical review of their work.

Alyami et al. [6] discuss possible mitigation techniques for fragmentation attacks in 6LoWPAN networks.They emphasized on lightweight encryption with reference to Ayuso et al. [7], on their idea to use a modified RSA and elliptical curve cryptography, as this combination proves to be highly efficient. They also focused on the usage of Intrusion Detection Systems (IDS) to analyze and detect any signs of suspicious behavior, as well as reputation score systems to monitor and control traffic based on the amount of trust built from peer to peer within a network. However, the solutions were not streamlined into one tool. The usage of many tools in a single system may lead to higher computational overhead, as well as the increased need for supervision and monitoring of several tools at once. It is impractical to train people to use a multitude of tools to mitigate fragmentation attacks, let alone regular users at home. Nonetheless, their paper was highly insightful as they highlighted key methods to deter several variants of fragmentation attacks.

Lin et al. [8] proposed the Deterministic Automaton (ADM-DDA6); A Deterministic Finite Automata-based adap- tive rule matcher, aligned with RFCs, achieves full coverage of extension header threats with only 20 rules and low overhead, outperforming Snort and Suricata. To reduce the overhead associated with traditional detection methods, the model matches rules dynamically selected based on the current header type. If any header is deemed abnormal, detection is stopped, reducing computational costs. Although their model proved to be superior to existing IDS’ based on their tests, it is important to note that rule-based detection systems do not adapt to newer threats. The issue sur- rounding rule-based filtering is that rules are explicitly coded, meaning that only if a malicious packet matches the right criteria, it gets dropped. Newer vectors of attacks may carry different, unidentified signatures that may slip by ADM-DDA6. It is also worth acknowledging that ADM-DDA6 demonstrated some noticeable impact on network performance under conditions of higher throughput.

Naagas et al. [9] proposed DEH-DoSv6 to combat DoS attacks in IPv6 networks by using extension headers. Their model functions as a traffic classifier by filtering or limiting the rate of legitimate or illegitimate traffic. The traffic that is supposed to be switched from an incoming interface to the forwarding layer, and then up to the router control plane, before it reaches the destination gets filtered through a rule set. This method functions as a firewall to prevent unsolicited traffic from overwhelming router resources. They believe that virtual fragment reassembly (VFR) is the best approach for their deployment of DEH-DoSv6 as it is a solid method to mitigating fragmentation attacks. However, as discussed by Cisco [10] in their technical documentation, the process may impose heavy computational overhead on the host. This would be especially noticeable in high-bandwidth scenarios or when deployed on resource- constrained and legacy systems. This potentially reduces its reliability under stress or attack.

Li et al. [11] proposed the P4-NSAF, a programmable data plane scheme to defend IPv6 networks against ICMPv6-based DoS and DDoS by categorizing attacks into pure flooding and source-spoofing vectors. Although P4- NSAF that does not specifically focus on fragmentation-based attacks, it remains an insightful and effective method that could serve to mitigate them. It employs a bloom-filter–backed PortTable, SingleConnectTable, and MultiConnectTable to record (IPv6 srcAddr, ingressPort, MAC) bindings and drop any packet whose source mapping isn’t found. For flooding attacks (e.g. Echo Requests, Router Solicitations), it uses a Count-Min Sketch per time window to count destination IPv6 addresses and drops packets once the sketch estimate exceeds an administrator-configured threshold. However, their proposed method is designed to operate at the switch level, making the solution inacces- sible to basic device users at home. Regular PCs typically do not have the same kind of programmable data plane architecture as high-end switches, and an attempt to implement the solution’s logic on regular PCs would be highly complex and impractical.

Tseng et al. [12] present a security framework aimed at detecting IPv6 DoS attacks by integrating signature-based Intrusion Detection Systems (IDS) and machine learning classification technologies to accurately identify abnormal behaviour on a network. The machine learning model chosen for their research was a decision tree, known for its interpretability and effectiveness in classifying network traffic anomalies [13]. However, their scope of research comes with some notable limitations. Their focus is on detecting attacks on Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) environments, which makes it a solution that is not applicable to the global populace. A universal solution would be one that can be installed on the host, without modifications in network protocol or infrastructure.

A summary of the critical review of the previously discussed research is provided in Table 1.

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

Authors Proposed Mechanisms Limitations

Alyami et al. [6] Combination of cryptography

and IDS

The solution was not streamlined into one mechanism; requires further research.

Lin et al. [8] ADM-DDA6 (i) May not mitigate newer threats. (ii) Demonstrated some noticeable impact on network performance under conditions of higher throughput.

Naagas et al. [9] DEH-DoSv6 Works on the basis of a predefined rule set entirely; does not adapt to different forms of attacks.

Li et al. [11] P4-NSAF Deployment requires programmable switches with suf- ficient pipeline stages and register memory.

Tseng et al. [12] Combination of IDS & Ma-

chine Learning

Their research scope is focused on SDN/NFV environ- ments, it is not a universal solution.

# METHODOLOGY

This section describes the design and implementation of the proposed solution, beginning with an outline of the system architecture, followed by an explanation of the data collection and feature extraction process. Subsequently, the decision tree classification model is deliberated, and finally, the dynamic firewall update procedure for mitigating fragmentation-based attacks is discussed.

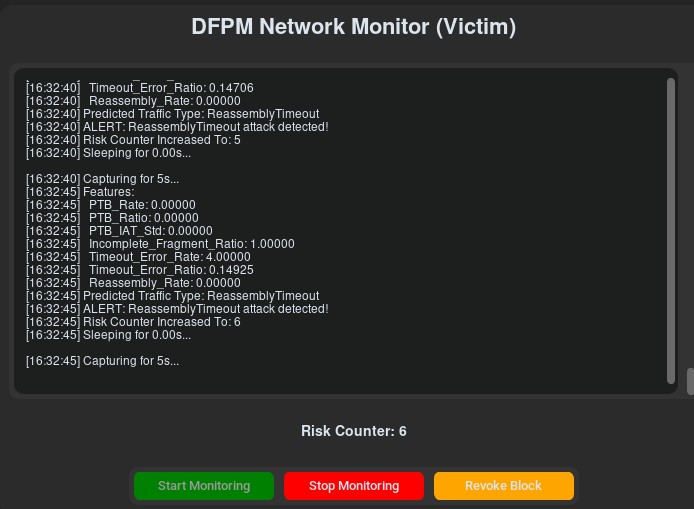
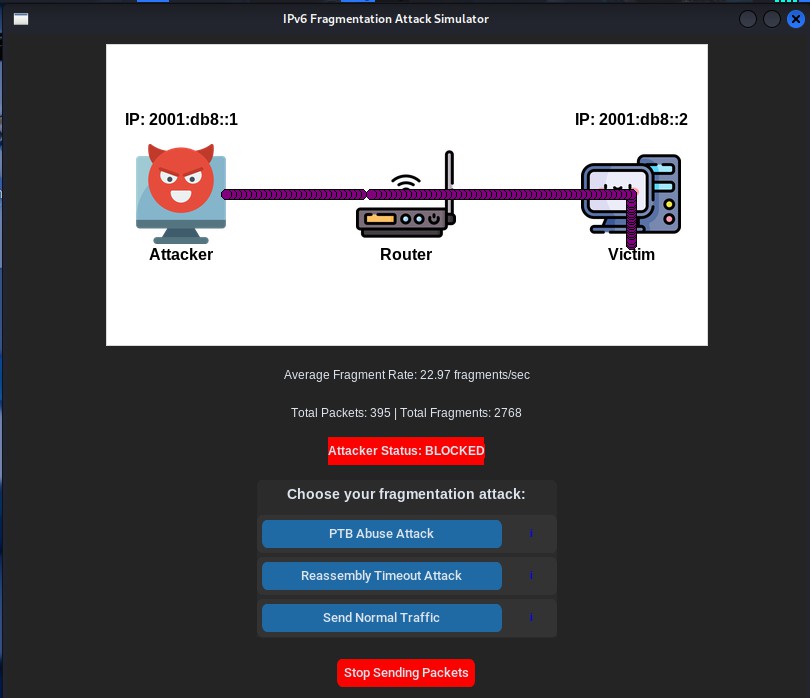
DFPM is implemented entirely at the host level, requiring no changes to upstream routers or network infrastructure. The victim host runs DFPM, which comprises of three primary components: the Packet Capture module, the Traffic Classification Engine, and the Policy Enforcement module. The process flow of DFPM is illustrated in Figure 1.



**FIGURE 1.** DFPM components process flow

To evaluate DFPM, a Mininet implementation was constructed comprising two end hosts (an attacker and a victim) and a router. To send malicious traffic to the victim host, an IPv6 Fragmentation Attack tool (see Figure 2) was created using Python with reference to Thc-Ipv6. Thc-Ipv6 is a toolkit designed using C for IPv6 security testing [14]. This toolkit was also utilized by Lin *et al.* [8] when testing their proposed mechanism.

Within each 5-second window, the Packet Capture Module on the DFPM Network Monitor (see Figure 3) captures all incoming IPv6 traffic on the host’s network interface. Subsequently, the Traffic Classification Engine parses the packet data to compute fragmentation attack indicators. Table 2 illustrates the features that are extracted and calculated from the captured IPv6 packets.



**FIGURE 2.** Topology visualization & the ipv6 fragmentation attack tool

**FIGURE 3.** The DFPM network monitor

**TABLE 2.** Extracted features from fragmented IPv6 traffic

|  |  |
| --- | --- |
| **Feature** | **Description** |
| PTB\_Rate | Rate of PTB messages per second |
| PTB\_Ratio | Ratio of PTB messages to total packets per capture |
| PTB\_IAT\_Std | Standard deviation of PTB inter-arrival times |
| Incomplete\_Ratio | Ratio of unassembled packets to total fragments per capture |
| Timeout\_Rate | Fragment timeout errors per second |
| Timeout\_Ratio | Ratio of timeout errors to total packets per capture |
| Reassembly\_Rate | Rate of successful reassemblies per second |
| Packet\_Rate | Rate of incoming packets per second |

The selection of these features was based on their strong indicative value in identifying fragmentation attacks. In the context of the ICMPv6 PTB flood, Elejla et al. [15] mention that excessive ICMPv6 error messages against hosts and routers can be sent to propagate their functionality. This is why the following metrics serve as attack indicators: the rate of incoming PTB messages (per second), the ratio of PTB messages to total regular packets, and the standard deviation of PTB error inter-arrival times.

For the Reassembly Timeout attack, it is mentioned in RFC 8900 that an attacker can construct a series of incom- plete fragment sets to repeatedly send to the victim [5]. This infers the indicators that should be considered are the ratio of incomplete fragment sets to the total number of captured fragments, the rate of incoming reassembly timeout errors, the ratio of timeout errors to the total number of captured packets, the rate of successful reassemblies per second, and the rate of incoming packets per second.

Upon completing computations, the Traffic Classification Engine begins its work to identify the traffic as malicious or benign. For the scope of this research, a supervised decision tree machine learning model was chosen as the Traffic Classification Engine for its interpretability [12] and low runtime overhead. This decision was heavily influenced by the number of attacks this research incorporates, dismissing the need for more powerful machine learning algorithms. The model was trained on three datasets comprising 27,000 samples (9000 each of Normal, ICMPv6 PTB Flood, Incomplete Reassembly). For the normal traffic, a dataset acquired from Mendeley Data [16] on IPv6 traffic was used to reproduce benign, realistic network traffic scenarios. As for the attack scenarios, the

datasets were produced using the IPv6 Fragmentation Attack Tool (as shown in Figure 2) to generate and capture malicious traffic for the machine learning model. The training was done using the DecisionTreeClassifier from scikit-learn, with Gini impurity as the split criterion. A grid search was used to tune the parameters, resulting in a tree with a maximum depth of 3 and a max of 1 feature to consider each time to make the split decision.

Each time a feature matches a rule in the tree, the risk score counter is increased by 2. If it does not match, the score stays the same. If the total risk score reaches 10 or more, it is considered an attack. This threshold was selected based on testing, where the model achieved a good balance between detecting actual attacks and avoiding false alarms. Figure 4 is a visual representative of the decision tree’s process flow.

A diagram of a system

AI-generated content may be incorrect.

**FIGURE 4.** DFPM traffic classification

Finally, the Policy Enforcement Module comes into action when the cumulative risk score ≥ 10. Table 3 maps each type of attack to a tailored firewall rule, designed to align with the unique characteristics and behavior of that specific threat to mitigate it.

**TABLE 3.** Tailored firewall rules for each attack

|  |  |
| --- | --- |
| **Attack Type** | **Host-Level Mitigation** |
| ICMPv6 PTB Flood | Apply rate-limiting rules [17] at the host’s firewall to control the rate of incoming "Packet Too Big" ICMPv6 messages. This reduces attack effectiveness without disabling fragmentation support. |
| Reassembly Timeout Attack | Block the attacker’s IP address via a host firewall rule to immediately stop further malicious fragments from being processed. |

Due to the nature of the ICMPv6 PTB flood attack where the attacker spoofs the IP address of the victim to induce the flurry of errors, it is not possible to block the IP address of the attacker. Doing so would result in disrupting the victim’s network. The best approach at the host level would be to perform rate-limiting, at the cost of potentially disrupting some legitimate traffic [17]. As for the reassembly timeout attack, detecting the IP address of the attacker is a straightforward process; therefore, blocking traffic from the malicious source should pose no significant challenge.

# RESULTS AND DISCUSSION

To analyze the effectiveness of DFPM in mitigating fragmentation attacks, the system was tested against 3 different packet transmission rates of each attack: Low (5 packets/second), Medium (10 packets/second), and High (20 pack- ets/second). The test was conducted across 20 consecutive packet captures for each transmission rate category. The mechanism’s effectiveness is measured using the metrics listed below in Table 4. These metrics were chosen because they are commonly used to assess the performance of a classification model.

**TABLE 4.** Computations used to determine the effectiveness of DFPM

|  |  |  |
| --- | --- | --- |
| **Metric** | **Formula** | **Description** |
| **Precision** | Precision = True Positives  True Positives+False Positives | Higher precision means fewer false positives. |
| **Recall** | Recall = True Positives  True Positives+False Negatives | Higher recall means fewer false negatives. |
| **F1-Score** | *F*1-*Score* = 2 *×* Precision*×*Recall  Precision+Recall | High F1 score indicates a well-balanced performance. |

Based on the findings of the tests (see Table 5), it is evident that DFPM demonstrated exceptional performance in detecting both the ICMPv6 PTB flood and the Reassembly Timeout attack across all three categories. However, there were a considerable number of false negatives for the Normal traffic test in the Low category. This indicates that the machine learning model requires further training for non-malicious, regular traffic. Overall, DFPM exhibited high accuracy with average scores of 0.96, 0.94, and 0.94 for Precision, Recall, and F1-Score, respectively. While such controlled traffic allows for clear evaluation, DFPM’s performance in live network environments may vary due to the unpredictable nature of real-world traffic. Nonetheless, the results indicate that the system has the potential to effectively mitigate fragmentation-based threats in IPv6 networks if given the proper data for training.

**TABLE 5.** DFPM detection accuracy by attack type and traffic intensity across 20 consecutive packet captures

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Traffic Type** | **Level** | **TP** | **FP** | **FN** | **Precision** | **Recall** | **F1-Score** |
| Normal | Low | 10 | 0 | 10 | 1.00 | 0.50 | 0.67 |
|  | Medium | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
|  | High | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
| Reassembly Timeout | Low | 20 | 10 | 0 | 0.67 | 1.00 | 0.80 |
|  | Medium | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
|  | High | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
| PTB Abuse | Low | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
|  | Medium | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
|  | High | 20 | 0 | 0 | 1.00 | 1.00 | 1.00 |
| **Average** | | | | | **0.96** | **0.94** | **0.94** |

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

# CONCLUSION

In conclusion, IPv6 is poised to become the backbone of our global network infrastructure, as its capabilities far extend beyond those of IPv4. With that in mind, this research is focused on solving security challenges during IPv6 packet fragmentation. DFPM’s test results indicate great potential for attack detection, but it struggles at times in discerning between malicious and regular traffic. While this may be a setback, DFPM’s base in machine learning allows for further training with more data to improve the accuracy of detection. Not only that, the system can be taught to mitigate other forms of fragmentation attacks. It is designed for host-level operation to avoid interfering with IPv6 network structures, ensuring wide compatibility. This is absolutely crucial to the mechanism’s function, as even the smallest change in network protocol can lead to network or performance degradation. With access to larger datasets and continuous refinement, DFPM could be a robust system in mitigating all forms of fragmentation-related attacks. Future research will explore several areas, including the evaluation of different machine learning models to effectively address more attack variants, an analysis of DFPM’s performance impact on the host, live-deployment testing, comparison with existing IDS tools, and strategies for optimizing the system to function efficiently in all environments.

# REFERENCES

1. O. Babatunde and O. Al-Debagy, “A comparative review of internet protocol version 4 (ipv4) and internet protocol version 6 (ipv6),” Interna- tional Journal of Computer Trends and Technology **13**, 10–13 (2014).
2. E. D. Paolo, E. Bassetti, and A. Spognardi, “A new model for testing ipv6 fragment handling,” (2023), arXiv:2309.03525 [cs.CR].
3. F. Gont, N. Hilliard, G. Döring, W. Kumari, G. Huston, and W. S. LIU, “Operational Implications of IPv6 Packets with Extension Headers,” RFC 9098 (2021).
4. K. Seo and S. Kent, “Security Architecture for the Internet Protocol,” RFC 4301 (2005).
5. R. Bonica, F. Baker, G. Huston, B. Hinden, O. Trøan, and F. Gont, “IP Fragmentation Considered Fragile,” RFC 8900 (2020).
6. S. Alyami, R. Alharbi, and F. Azzedin, “Fragmentation attacks and countermeasures on 6lowpan internet of things networks: Survey and simulation,” Sensors **22** (2022), 10.3390/s22249825.
7. J. Ayuso, L. Marin, A. Jara, and A. F. G. Skarmeta, “Optimization of public key cryptography (rsa and ecc) for 16-bits devices based on 6lowpan,” in *1st International Workshop on the Security of the Internet of Things, Tokyo, Japan* (2010) pp. 1–8.
8. B. Lin, L. Zhang, H. Zhang, Y. Guo, S. Ge, Y. Fang, and M. Ren, “An adaptive detection model for ipv6 extension header threats based on deterministic decision automaton,” Scientific Reports **14** (2024), 10.1038/s41598-024-59913-8.
9. M. Naagas, A. Malicdem, and T. Palaoag, “Deh-dosv6: A defendable security model against ipv6 extension headers denial of service attack,”, 274–282 (2021).
10. Cisco Systems, *Configuring Virtual Fragment Reassembly for CBAC*, Cisco (2012), cisco IOS 12.4T Security Configuration Guide.
11. Y. Li, W. Yang, Z. Zhou, Q. Liu, Z. Li, and S. Li, “P4-nsaf: defending ipv6 networks against icmpv6 dos and ddos attacks with p4,” in *ICC 2022 - IEEE International Conference on Communications* (2022) pp. 5005–5010.
12. C.-W. Tseng, L.-F. Wu, S.-C. Hsu, and S.-W. Yu, “Ipv6 dos attacks detection using machine learning enhanced ids in sdn/nfv environment,” in *2020 21st Asia-Pacific Network Operations and Management Symposium (APNOMS)* (2020) pp. 263–266.
13. N. K. Radke, S. S. Tomar, and A. Rajan, “Study on machine learning models for ipv6 address lookup in large block lists,” in *2023 National Conference on Communications (NCC)* (2023) pp. 1–6.
14. The Hacker’s Choice and Kali Linux, “THC-IPv6,” [https://www.kali.org/tools/thc-ipv6/](http://www.kali.org/tools/thc-ipv6/) (2024).
15. 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** (2022), 10.3390/app12126150.
16. O. Salamkayala, “ICMPv6\_DDOS- Dataset,” (2024).
17. M. Byerly, M. Hite, and J. Jaeggli, “Close Encounters of the ICMP Type 2 Kind (Near Misses with ICMPv6 Packet Too Big (PTB)),” RFC 7690 (2016).