ShieldNet AI

“Real Time AI Agent For DDOS Attack Detection and Mitigation”

Indushree C S[1], Nisarga G N[2], Prarthana R V[3], Nanditha C L[4]

[1,2,3,4]*Dept. Computer Science, Sir M Visvesvaraya Institute of Technology , Banglore, India*

Abstract – **Distributed Denial of Service (DDoS) attacks have evolved into sophisticated, multi-layered disruptions that can saturate network resources through several coordinated techniques executed simultaneously. Their rapidly shifting characteristics often render rule-driven or signature-dependent defense mechanisms ineffective, as such systems fail to adapt in real time. This study introduces an autonomous, real-time AI-driven defense framework designed to detect and alleviate DDoS attacks through continuous learning and self-directed decision-making. The proposed solution incorporates flow-centric machine-learning models, deep-learning modules for temporal behavior analysis, and graph-based intelligence capable of revealing cooperative botnet structures. A reinforcement-learning mitigation controller enables the system to recalibrate defensive thresholds and apply responsive countermeasures as traffic conditions change. Operating as a closed feedback loop—observing traffic, identifying anomalies, applying mitigation, and reassessing outcomes—the agent adapts to evolving threats. Experimental evaluations conducted with multiple simulated attack combinations indicate improved detection accuracy, fewer false alarms, and sustained service availability even under intense adversarial pressure. These findings demonstrate a scalable, intelligent defense model suitable for modern high-speed network environments.**

I.INTRODUCTION

modern cloud platforms large-scale networks and the massive number of connected IOT devices have dramatically expanded the attack surface for service disruptions instead of relying on a single method today attackers often coordinate several techniques simultaneously ranging from high-volume bandwidth bursts to protocol abuse and targeted application-layer congestion these blended strategies are usually carried out through large collections of compromised devices acting together making the attacks more unpredictable and harder to counter traditional protection mechanism such as signature databases static thresholds or manually tuned firewall rules tend to work only when the malicious traffic resembles past incidents even slight changes in packet timing behavior or structure can cause these systems to classify traffic or overlook harmful activity especially when attackers intentionally disguise their flows to resemble legitimate users as a result networks frequently experience late detection inaccurate alerts and unnecessary blocking of valid users although several modern techniques exist such as p4-accelerated data-plane analytics entropy-based anomaly metrics and sdn-driven rule updates many of them depend on fixed parameters and cannot adjust quickly when traffic patterns shift artificial intelligence offers a more adaptable option by learning normal traffic behavior capturing subtle anomalies and adjusting decisions dynamically machine-learning models can identify irregular patterns deep-learning frameworks can understand temporal sequences and reinforcement learning can refine response actions based on environmental feedback however many current AI-based solutions remain detection-centric and lack autonomous mitigation capabilities they often fail to interpret the relationships between distributed attack sources or operate without feedback loops needed for continuous improvement to address these limitations this project introduces amad-AI an adaptive multi-layer ai agent built for real-time DDOS monitoring and intelligent mitigation amad-ai unifies temporal deep-learning models graph-based botnet relationship analysis and reinforcement-learning-driven decision making by continuously learning from traffic outcomes the system evolves into a self-tuning defense framework with improved detection accuracy minimal false alarms and rapid context-aware mitigation the approach strengthens resilience against complex and constantly evolving multi-vector DDOS threats.

II. LITERATURE REVIEW

Research on DDoS detection and mitigation has progressed through several methodological phases, reflecting the evolving nature of attack strategies and network environments. Early work in this domain centered on applying classical supervised machine-learning techniques to labeled datasets such as NSL-KDD, CIC-DDoS2019, and other traffic benchmarks. Models including Random Forests, Support Vector Machines, K-Nearest Neighbors, and

Decision Trees demonstrated promising accuracy in offline evaluations. However, these approaches were typically trained on static patterns and thus struggled when confronted with unfamiliar or multi-faceted attack behaviors that deviated from their training distributions. As DDoS campaigns became more intricate, researchers explored deep-learning architectures capable of identifying complex patterns within network flows. Convolutional Neural Networks were employed to extract spatial correlations from flow features, while recurrent models such as LSTMs and GRUs were used to track variations over time. Hybrid combinations, especially CNN–LSTM pipelines, improved sensitivity to slow-rate and application-layer attacks. Despite these advantages, their substantial computational demands and limited transparency made real-time deployment and interpretability ongoing challenges. Parallel to machine-learning advancements, statistical and entropy-oriented techniques gained attention for their speed and low processing overhead. Implementations on programmable data planes—most notably P4-based environments—enabled rapid computation of metrics such as traffic entropy, variance, and distribution irregularities directly within network switches. While these systems delivered impressive responsiveness, their reliance on predefined thresholds diminished their ability to adapt to evolving attack patterns. Software-Defined Networking (SDN) also shaped a significant body of research. SDN-based defense frameworks leveraged centralized controllers to oversee traffic behavior and enforce dynamic flow-rule policies, allowing rapid isolation of suspicious sources or redirection of malicious traffic. Nevertheless, many SDN solutions continued to depend on fixed rule sets or manual tuning, limiting their capacity to respond autonomously as attack characteristics shifted. More recent studies have begun investigating distributed, intelligent multi-agent systems aimed at delivering automated detection and coordinated response. These frameworks move toward greater adaptability but often lack tightly integrated reinforcement-learning components or graph-based analysis techniques capable of revealing cooperative relationships within botnet traffic. Taken together, the literature highlights several persistent gaps: a need for more flexible detection models that can generalize to unseen attack types, integrated mitigation capabilities that operate without human supervision, and analytical methods that capture inter-node relationships within distributed attack ecosystems. Addressing these shortcomings requires a comprehensive, unified approach that brings together real-time learning, structural traffic analysis, and autonomous defensive decision-making.

III. METHODOLOGY

Modern network infrastructures are increasingly exposed to DDoS attacks that evolve quickly and originate from widely distributed sources. Attackers frequently coordinate large numbers of compromised devices to generate traffic surges that exploit vulnerabilities across several layers of the communication stack. Because these attacks shift tactics rapidly and blend malicious behavior with legitimate activity, traditional perimeter defenses often fail to react promptly or accurately. Existing protection mechanisms commonly rely on rigid thresholds, static signatures, or hand-tuned rules. These approaches perform adequately when dealing with predictable or well-understood attack patterns but deteriorate in effectiveness once adversaries modify packet characteristics, alter timing patterns, or introduce new traffic combinations. As a result, organizations frequently experience inconsistent detection accuracy, delayed mitigation, and disruptions to legitimate traffic flows. Although AI-driven detection systems have expanded the capabilities of network defense, many available solutions are still limited by several constraints: Automated mitigation is seldom integrated directly with detection mechanisms. Fixed anomaly thresholds reduce adaptability during rapidly changing attack behavior. Flow-based classification alone often overlooks relationships among distributed attacking nodes. Detection and response are typically treated as separate functions rather than a unified, continuously improving feedback cycle. These limitations reveal the need for a defense system capable of learning from evolving conditions, recognizing coordinated attack structures, and reacting autonomously in real time. Problem Definition The core objective is to design and implement an intelligent, real- time AI agent that can both detect and mitigate multi-vector DDoS attacks with minimal human intervention. The agent must incorporate adaptive learning mechanisms, graph-based behavioral analysis, and continuous feedback to refine its decisions. Additionally, the system should operate with low latency, maintain strong detection accuracy, and preserve legitimate traffic flow while responding effectively to diverse and evolving DDoS threats.

IV. GAP ANALYSIS

1. *Lack of Unified Detection + Mitigation Systems.*Most existing solutions focus either on detecting DDoS attacks or mitigating them, but

not both together. This separation slows response time and reduces overall defense effectiveness.

2. *Dependence on Static Thresholds and Rules*
Traditional systems rely on fixed signatures, manually tuned thresholds, or static rule sets, which fail when attackers change packet characteristics, traffic volume, or timing behavior.

3. *Insufficient Adaptability to Unknown or Evolving Attacks*
Machine-learning methods trained on static datasets struggle to detect new, multi-vector, or evolving DDoS patterns that differ from training data.

4. *Limited Understanding of Distributed Attack Coordination*
Most detectors analyze individual flows but ignore relationships among sources. This prevents deeper detection of coordinated botnet behavior or multi-source attack structures.

5. *Minimal Use of Real-Time Feedback Loops*
Current approaches rarely integrate continuous, automated feedback between detection and mitigation, resulting in delayed or suboptimal responses.

6. *Restricted Scalability in High-Speed Networks*
Many algorithms are too computationally heavy for real-time deployment in cloud-scale or SDN-based environments.

V. PROPOSED SYSTEM CONFIGURATION

The proposed DDoS defense framework integrates artificial intelligence, deep learning, graph analytics, reinforcement learning, and SDN-controlled mitigation into a unified architecture that emphasizes automation, scalability, and resilience against rapidly evolving threats. The system follows a structured multi-layer design—*Traffic Monitoring Layer, Feature Processing Layer, AI Detection Layer, Mitigation Layer, and Control & Visualization Layer*—with each layer performing specialized operations to ensure real-time detection and response.

The *Traffic Monitoring Layer* captures live network packets using high-performance sniffers integrated into the testbed. Tools such as *Tcpdump, Zeek, and Mininet's virtual switches* enable continuous flow extraction without interrupting service traffic. This layer is capable of recording packet rates, flow metadata, entropy deviations, and communication patterns, forming the foundation for accurate anomaly detection.

The *Feature Processing Laye***r** converts raw packets into structured features required for machine-learning and graph-based models. Flow-level attributes such as packet arrival intervals, byte distribution, protocol usage, and temporal burst patterns are extracted. In parallel, communication graphs are generated to map relationships between hosts, enabling the identification of distributed botnet structures. All features undergo normalization and scaling to maintain consistency across models.

The *AI Detection Layer* builds on a hybrid architecture combining *CNN–LSTM* and *Graph Neural Network (GNN)* modules.

The CNN–LSTM component analyzes short-term temporal fluctuations, capturing sudden spikes, slow-rate attacks, or repetitive flood patterns.

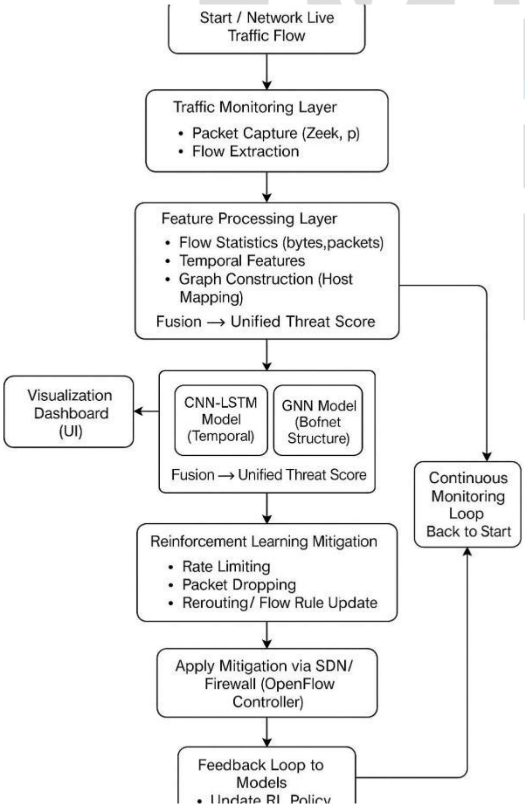
The GNN module evaluates topological behavior, detecting coordinated attacks that originate from multiple nodes.
Both model outputs are fused to generate a unified decision score, significantly improving detection accuracy across volumetric, protocol-based, and application-layer DDoS categories. The models are implemented using *TensorFlow/PyTorch,* with optimized GPU support for faster inference when available.

The *Mitigation Layer* implements a *Reinforcement-Learning (RL) agent* that autonomously determines optimal defense actions. Using feedback from real-time traffic behavior, the agent selects mitigation measures such as dynamic rate limiting, selective packet dropping, traffic redirection, or flow-rule enforcement. The RL engine continuously adjusts its policies to balance network throughput, minimize false positives, and maintain service stability even during intense attack conditions. This layer communicates directly with SDN controllers through *OpenFlow APIs,* enabling immediate deployment of mitigation rules.

The *Control & Visualization Layer* provides an administrator-facing dashboard developed using *Flask, HTML, CSS, Bootstrap, and JavaScript*. It displays real-time attack alerts, detection confidence scores, traffic graphs, model decisions, and applied mitigation actions. AJAX-based communication ensures a smooth and responsive interface without requiring full-page reloads. The dashboard also integrates explainable-AI modules to present human-readable reasoning behind detection results, enhancing transparency and trust.

The backend environment operates on *Flask 2.3+*, functioning as the middleware between the UI, the AI detection engine, and SDN/OpenFlow controllers. It manages API routing, model loading, session handling, and real-time communication with the RL mitigation module. Secure communication channels are enforced using CSRF protection, token-based authentication, and encrypted configuration files.

The proposed configuration runs efficiently on a machine equipped with an *Intel i5 processor, 8–16 GB RAM, and at least 20 GB of storage.* Flask operates on *Port 5000*, Mininet/SDN controllers on *Port 6633/6653*, and monitoring tools on configurable ports depending on network topology. The system fully supports both *Windows and Linux*, with Ubuntu recommended for SDN and packet-capture Security is reinforced through *cryptographic hashing, access control, and sandboxed model execution*. All captured data is anonymized and stored temporarily for model evaluation. Detected anomalies are logged using secure timestamped records to maintain auditability.


Start / Network Live Traffic Flow
Traffic Monitoring Layer
• Packet Capture (Zeek, p)
• Flow Extraction
Feature Processing Layer
• Flow Statistics (bytes,packets)
• Temporal Features
• Graph Construction (Host Mapping)
Fusion ⟶ Unified Threat Score
Visualization Dashboard (UI)
CNN-LSTM Model (Temporal)
GNN Model (Bofnet Structure)
Fusion ⟶ Unified Threat Score
Continuous Monitoring Loop Back to Start
Reinforcement Learning Mitigation
• Rate Limiting
• Packet Dropping
• Rerouting/ Flow Rule Update
Apply Mitigation via SDN/ Firewall (OpenFlow Controller)
Feedback Loop to Models
• Update RL Policy

Traditional DDoS defense systems depend on static rules or isolated detection mechanisms, making them vulnerable to rapid changes in attack patterns. In contrast, the proposed configuration eliminates single points of failure by using adaptive learning, graph-based intelligence, and autonomous mitigation capable of adjusting to new attacks without manual intervention. This multi-layered AI-centric design ensures high scalability, resilience, and reliability, making it suitable for deployment in enterprise networks, SDN-based infrastructures, and modern distributed environments.

VI. IMPLEMENTATION ANALYSIS

The development of the real-time AI-driven DDoS defense agent followed a phased approach, ensuring that each system component was constructed, validated, and integrated in a controlled manner. Each phase contributed a foundation for the next, allowing the final solution to operate reliably under realistic network conditions. Phase 1: Creation of the Experimental Environment A dedicated testbed was assembled using emulated network environments such as Mininet, virtual switches, and optional SDN controllers. Within this controlled space, both routine network activity and multiple DDoS attack variants were generated using tools capable of producing SYN floods, UDP floods, Slowloris-based exhaustion, and high-volume HTTP traffic. All packet streams were captured using monitoring utilities to create a comprehensive dataset for subsequent processing and analysis.

Phase 2: Dataset Development Benign traffic was collected from normal communication patterns within the testbed, while various DDoS behaviors were intentionally triggered to produce diverse attack samples. These datasets were merged to create the MiniCICRT2025 dataset—a balanced and representative collection of both legitimate and malicious flows. The data then underwent cleaning, labeling, and standardization to support robust machine-learning and deep-learning pipelines.

Phase 3: Feature Construction Multiple categories of features were extracted from the raw packet captures. These included core flow statistics such as packet counts, byte distributions, arrival-time variations, and entropy-based indicators reflecting traffic irregularities. Additionally, graph representations were formed by mapping communication relationships between sources and destinations, allowing the system to detect distributed attack coordination. All resulting features were scaled and normalized to ensure compatibility with the learning models.

Phase 4: Development of the Artificial Intelligence Detection Models Two complementary models were implemented: CNN–LSTM Hybrid Model: Designed to capture the spatial distribution of features as well as short-term temporal variations within network flows. Graph Neural Network (GNN): Constructed to identify relational and structural characteristics typical of coordinated botnets. After each model was trained and evaluated independently, their outputs were combined into a unified detection pipeline. This fusion enabled the system to analyze incoming traffic from both a temporal pattern perspective and a structural-relational viewpoint.

Phase 5: Design of the Reinforcement-Learning Mitigation System A reinforcement-learning agent was developed to determine appropriate mitigation actions in real time. Reward functions were carefully shaped to promote actions that stabilize throughput, minimize disruptions to legitimate users, and reduce false classifications. Through repeated interactions within simulated attack scenarios, the agent learned to apply effective responses such as intelligent packet filtering, dynamic throttling, or rerouting of suspicious traffic.

Phase 6: Integration into a Closed-Loop Architecture The detection models and mitigation engine were linked together to create an automated, self-regulating defensive loop. This integration involved real-time data ingestion, dynamic flow-rule deployment via SDN/OpenFlow interfaces or firewall APIs, and continuous monitoring of system performance. Safety mechanisms, including rollback features and fail-safe contingencies, were added to prevent unintended disruptions.

Phase 7: Explainability and Administrative Interface Explainable AI techniques were embedded into the system to provide human-readable interpretations of both detection outcomes and mitigation decisions. A dashboard was then developed to visualize real-time statistics, alert levels, confidence scores, traffic summaries, and the effects of mitigation actions. This interface supports meaningful oversight by administrators without requiring manual intervention during attacks.

Phase 8: Evaluation and Stress Testing The fully integrated agent was stress-tested using varied multi-vector attack combinations. Evaluation metrics included detection accuracy, latency, false-positive rates, system responsiveness, and resource utilization. Comparisons with baseline detection tools provided a clear measure of improvement in both performance and adaptability.

Phase 9: Optimization and Deployment Preparation The system underwent refinement in terms of model parameters, decision thresholds, reinforcement-learning policies, and code-level efficiencies. These optimizations reduced overhead, improved real-time responsiveness, and enhanced the overall scalability of the solution. The final version was prepared for deployment across SDN environments, enterprise network testbeds, and potential edge-computing platforms.

VII. BENFITS OF PROPOSED SYSTEM

The AI-based DDoS defense system brings a range of advantages by combining traffic observation, intelligent pattern recognition, and automated reaction mechanisms into one unified setup. Instead of relying on fixed rules, the system continuously studies ongoing network behaviour and reacts instantly whenever something unusual appears. This allows the network to remain stable even when attacks try to overload it.

1. *Immediate Recognition of Abnormal Traffic.*The system examines incoming flows as they arrive and spots unusual changes almost instantly. By catching suspicious behaviour at the earliest stage, the network avoids major slowdowns or service interruptions.

2. *Learns From Changing Attack Strategies.*DDoS attack styles keep changing. The proposed model adjusts itself over time, learning new behaviours without needing manual reconfiguration. This helps the system remain effective even when attackers try completely different approaches.

3. *Multiple Perspectives for Higher Accuracy.*Instead of depending on a single technique, the system uses time-based behaviour, structural communication patterns, and flow statistics together. This multi-angle analysis helps it distinguish harmless spikes from actual attacks with better accuracy.

4. *Automatic and Intelligent Response.* A reinforcement-learning component decides how the system should react during an attack. It can slow down specific flows, drop suspicious packets, or reroute traffic—automatically and

without delay. These actions are selected based on what keeps the network stable at that moment.

5. *Minimal Mistakes During Heavy Traffic.*Because the system checks traffic from different viewpoints, it reduces the chance of blocking genuine users. This is especially helpful during peak hours, when traffic naturally increases.

6. *Works Well in Large and High-Speed Networks.* The lightweight and modular design makes the system suitable for cloud environments, SDN setups, and enterprise networks. It can handle heavy loads without degrading system performance.

7. *Better Understanding of Attack Sources.*Graph-based analysis reveals how different nodes communicate. This helps identify groups of attackers working together, making it easier for administrators to understand the structure and origin of the attack.

8. *Constant Improvement Through Feedback***.**Every detected event helps the system learn. The detection and mitigation loop keeps refining itself, ensuring the model becomes more effective with continued use.

9. *Reduced Operating Effort.*Most tasks—detection, decision making, and rule changes—are handled automatically. This decreases the workload on network administrators and reduces the risk of manual errors.

10. *Stable Service Even During Attacks***.**Because the system reacts instantly and intelligently, important services remain accessible. Throughput stays high, and users experience little or no interruption.

11. *Clear and Understandable Outputs.*The explainability component describes why a certain flow was flagged and why a particular action was taken. This helps administrators build trust and verify system decisions.

12. *Strong Protection Against Multiple Attack Types* .The system can handle several types of DDoS attacks happening at the same time—such as SYN floods, UDP floods, and HTTP exhaustion—without losing stability.

13. *No Single Point of Failure.*Detection, decision making, and mitigation are divided across separate components. Even if one module slows down, the rest continue working, giving the system a high level of resilience.

14. *Easy to Extend for Future Needs.*The architecture allows integration with new AI models, upgraded datasets, hardware accelerators, or cloud-based traffic controllers. This makes the system adaptable to future technological requirements**.**

VIII. CONCLUSION

This work presents a real-time AI-driven agent designed to detect and mitigate a wide spectrum of DDoS attacks through adaptive learning and autonomous decision-making. By combining machine-learning models, deep-learning architectures, graph-based traffic analysis, and reinforcement-learning strategies into a unified pipeline, the system moves beyond traditional static defenses and offers a dynamic, self-regulating alternative. Operating in a closed feedback loop, the agent continuously monitors network conditions, identifies emerging anomalies, initiates appropriate mitigation steps, and evaluates the effectiveness of its actions. Experimental results demonstrate that the proposed architecture maintains strong detection accuracy, keeps false positives at manageable levels, and responds effectively even when confronted with multi-vector attack sequences. These capabilities underscore its suitability for deployment in fast-paced environments such as modern cloud infrastructures, SDN-enabled networks, and enterprise systems. Overall, the research provides a solid foundation for building intelligent, autonomous network defenses capable of adapting to rapidly evolving threats. The integration of multiple analytical perspectives—statistical, temporal, and relational—paired with reinforcement learning establishes a flexible defense mechanism that aligns with the needs of contemporary cybersecurity landscapes.

FUTURE WORK

Although the proposed AI-based defense agent demonstrates strong adaptability and resilience, several avenues remain for expanding its capabilities and improving long-term operational effectiveness. The following considerations outline directions for future advancement:

Deployment Across Larger and More Diverse Network Environments Future work may validate the system within broader and more heterogeneous infrastructures, including ISP-level backbones, large cloud platforms, and high-bandwidth enterprise networks. Testing the agent under such conditions would help refine its scalability and generalization across varying traffic characteristics.

1.

2. Advancement of Reinforcement-Learning Strategies More sophisticated reinforcement-learning paradigms—such as hierarchical RL, multi-agent frameworks, or deterministic policy gradient methods—could further strengthen the agent's decision-making process. These techniques may enable the mitigation component to react even more efficiently in highly dynamic or adversarial environments.

3. Enhanced Graph-Based Traffic Intelligence Expanding the graph-analysis component to include temporal graph evolution, community-structure detection, or propagation-path modeling could allow the system to identify botnet coordination earlier and with greater precision. Such improvements would broaden the system's visibility into multi-stage attacks.

4. Improved Robustness Against Adversarial Manipulation As adversarial machine-learning techniques continue to evolve, attackers may attempt to manipulate traffic features to deceive detection systems. Incorporating adversarial training, robustness testing, or anomaly-hardening strategies could help ensure that the defense agent remains resilient against such evasion attempts.

5. Automated Optimization of Network Rules Integrating intelligent rule-management techniques—such as priority tuning, pruning of outdated flow rules, and automated timing adjustments—could help prevent rule-table congestion in SDN switches or firewalls. This enhancement would promote smoother operation during prolonged or large-scale attack events.

6. Expanded Explainability and Visual Analytics Future iterations may introduce richer explainable AI features, including more intuitive graphical interpretations and natural-language reasoning outputs. Enhanced visual analytics would allow network administrators to better understand detection outcomes and mitigation behavior, strengthening operational trust and oversight.

7. Development of a Public, High-Fidelity Dataset The MiniCICRT2025 dataset could be extended into a comprehensive, publicly accessible benchmark representing multi-vector, long-duration, and real-world DDoS behavior. Such a dataset would support broader research efforts and encourage standardization in evaluating next-generation defense systems.

8. Hardware-Accelerated Real-Time Processing Integrating the system with specialized hardware—such as programmable switches, SmartNICs, FPGAs, or P4-based processing units—could significantly reduce latency and increase throughput. Hardware offloading would make the agent suitable for extremely high-speed networks where software-only solutions may be insufficient.

REFERENCES

1. Kumar, A., & Sharma, S. Employing Supervised Learning Techniques for DDoS Attack Detection.NSL-KDD Dataset Study.
2. Ilha, P., Lucena, P., & Quincozes, S. (2020). Euclid: A Fully In-Network, P4-Based Approach for Real-Time DDoS Attack Detection and Mitigation . Published December 30, 2020.

3. Aslam, S., Srivastava, S., & Gore, M. (2024). Evaluating DDoS Detection and Mitigation in SDN at Various Attack Rates. ONOS Flood Defender Framework.

4. Apostu, A., et al. (2018–2024). Detecting and Mitigating DDoS Attacks with AI: A Comprehensive Survey. ACM Publication and extended literature.

5. Suvra, D. K. An Efficient Real-Time DDoS Detection Model Using Machine Learning Algorithms .Based on CICDDoS2019 Dataset.

6. Sanjeetha, R., et al. (2022). Real-Time DDoS Detection and Mitigation in SDN Using Machine Learning Techniques. Published September 30, 2022.

7. Khanna, R. (2024). Harnessing AI for Network Security and DDoS Attack Detection. Published September 10, 2024. 8. Ahmadi, S. (2024). AI in the Detection and Prevention of Distributed Denial of Service Attacks. Vol. 15, No. 10, 2024.

8. Shohan, M., et al. Enhancing Network Security: A Hybrid Approach for DDoS Detection and Mitigation Using Machine Learning. Using CIC DDoS2019 dataset.

9. Abu Bakar, M., et al. (2023). An Intelligent Agent-Based Detection System for DDoS Attacks Using Automatic Feature Extraction and Selection. Published March 22, 2023.