International Journal of Computers and Informatics

Journal Homepage: https://www.ijci.zu.edu.eg

Int. j. Comp. Info. Vol. 11 (2026) 16–35

Paper Type: Original Article

FlowPrint-P4: Lightweight Behavioral Anomaly Detection for DDoS Mitigation in Programmable Networks

Aya Hassan [1,2,*], **Marwa M. Khashaba** [1], **Ehab R. Mohamed** [1] and **Ameer El-Sayed** [1]

[1] Department of Information Technology, Faculty of Computers and Informatics, Zagazig University, Zagazig 44511, Egypt.
[2] Department of Information Technology, Faculty of Computers and Informatics, Damnhour University, Nubaria 22773, Egypt.

Emails: ayahasan@fci.zu.edu.eg, mmkhashaba@fci.zu.edu.eg, ehab.rushdy@zu.edu.eg, aegouda@fci.zu.edu.eg

Received: 13 Nov 2025 **Revised:** 28 Feb 2026 **Accepted:** 22 May 2026 **Published:** 23 May 2026

Abstract

Distributed Denial-of-Service (DDoS) attacks continue to pose significant threats to modern programmable and IoT-enabled networks by overwhelming bandwidth, exhausting server resources, and degrading service availability. Traditional mitigation approaches largely depend on centralized control-plane analysis and volumetric thresholds, which introduce detection latency and remain ineffective against stealthy or low-rate attacks. This paper presents FlowPrint-P4, a lightweight in-network DDoS detection and mitigation framework based on behavioral flow fingerprinting implemented directly within P4-programmable data planes. The proposed framework analyzes flow-level behavioral features, including TTL variance, SYN/ACK asymmetry, TCP flag anomalies, burstiness patterns, and connection churn rates, enabling real-time identification of malicious traffic without reliance on external controllers. FlowPrint-P4 performs inline detection and packet tagging entirely within the switch pipeline, allowing immediate mitigation actions such as dropping, rate limiting, or redirection at line rate. Experimental evaluation was conducted using the CIC-IoT2023, ToN-IoT, and CIC-IoMT2024 benchmark datasets within a BMv2-Mininet programmable networking environment. Results demonstrate high detection effectiveness across multiple DDoS categories, including TCP SYN floods, UDP floods, ICMP floods, reflection/amplification attacks, and slow-rate application-layer attacks. The framework achieved high classification accuracy with low false positive and false negative rates while maintaining scalability and low processing overhead. These findings demonstrate the feasibility of behavioral fingerprinting as a practical and efficient approach for real-time in-network DDoS mitigation in programmable networks.

Keywords: SDN; IoT Security; Behavioral Flow Fingerprinting; In-Network DDoS Detection; Real-Time Mitigation; Flow-Based Traffic Analysis.

1 | Introduction

Distributed Denial-of-Service (DDoS) attacks remain among the most disruptive cybersecurity threats affecting modern communication infrastructures, cloud services, and Internet-of-Things (IoT) environments [1, 2]. By generating massive volumes of malicious traffic or exploiting protocol weaknesses,

Corresponding Author: ayahasan@fci.zu.edu.eg

attackers aim to exhaust network bandwidth, server resources, or connection states, ultimately degrading service availability for legitimate users [3]. The rapid expansion of IoT ecosystems, edge computing, and high-speed programmable networks has further increased the attack surface, enabling adversaries to launch highly distributed and adaptive attacks using compromised devices with minimal computational requirements [4-6].

Traditional DDoS mitigation techniques primarily rely on traffic volume analysis, centralized monitoring systems, or Software-Defined Networking (SDN) control-plane mechanisms to identify abnormal traffic behavior [7, 8]. Common mitigation strategies include threshold-based filtering, blackholing, traffic scrubbing, and rate limiting. Although these approaches remain effective against large volumetric floods, they often struggle to detect low-rate or stealthy attacks that intentionally mimic legitimate traffic patterns and remain below predefined thresholds [9-11]. In addition, centralized detection architectures introduce additional communication overhead and mitigation latency because traffic statistics must first be collected, analyzed, and translated into enforcement policies before mitigation actions can be applied [12, 13].

Recent advances in programmable networking have introduced new opportunities for implementing security mechanisms directly within the data plane [14, 15]. The P4 programming language enables network operators to define customized packet-processing logic inside programmable switches, allowing packets to be inspected, classified, and processed at high speed without dependence on fixed-function hardware behavior [16, 17]. This capability has motivated growing research interest in in-network security mechanisms, where anomaly detection and mitigation can be performed closer to traffic sources and at earlier stages of packet forwarding [18-20].

However, implementing effective DDoS detection directly in programmable switches remains challenging due to strict hardware constraints, including limited memory resources, restricted computational capabilities, and the absence of complex arithmetic operations in the data plane [19]. Many existing approaches either depend heavily on external controllers and machine-learning frameworks or require computationally expensive operations that are difficult to implement efficiently in P4-enabled environments [10]. Furthermore, several solutions focus primarily on specific attack categories, such as TCP SYN floods, while providing limited generalization across heterogeneous DDoS behaviors common in IoT and distributed environments [21].

To address these limitations, this paper proposes FlowPrint-P4, a lightweight in-network DDoS detection and mitigation framework based on behavioral flow fingerprinting implemented directly within programmable data planes. Instead of relying solely on traffic volume thresholds, the proposed framework analyzes flow-level behavioral characteristics observable within packet headers and lightweight switch state. These characteristics include Time-To-Live (TTL) variation, SYN/ACK asymmetry, abnormal TCP flag combinations, burst-rate behavior, and connection churn patterns. By monitoring these features directly inside the P4 pipeline, FlowPrint-P4 enables early identification of suspicious traffic flows and supports inline mitigation actions such as packet tagging, rate limiting, or flow dropping without requiring continuous controller intervention.

The proposed framework is designed to operate using lightweight register-based tracking and threshold-driven behavioral analysis suitable for resource-constrained programmable switches. Unlike conventional centralized approaches, FlowPrint-P4 performs detection directly within the forwarding path, reducing mitigation latency and minimizing control-plane dependency. The framework is evaluated using publicly available benchmark datasets, including CIC-IoT2023, ToN-IoT, and CIC-IoMT2024, within a programmable networking environment based on BMv2 and Mininet. Experimental results demonstrate that behavioral flow fingerprinting can effectively identify multiple DDoS categories, including TCP SYN floods, UDP floods, ICMP floods, reflection/amplification attacks, and slow-rate application-layer attacks, while maintaining low false alarm rates and scalable in-network operation.

The main contributions of this work are summarized as follows:

- Proposing a lightweight behavioral flow fingerprinting framework for real-time DDoS detection in programmable data planes.
- Designing a P4-compatible feature extraction mechanism using lightweight stateful primitives and register-based monitoring.
- Implementing controller-independent in-network mitigation through inline packet tagging and flow classification.
- Evaluating the proposed framework using realistic public cybersecurity datasets covering heterogeneous IoT and DDoS attack scenarios.
- Demonstrating the feasibility of integrating behavioral anomaly detection into programmable forwarding pipelines while considering practical data-plane constraints.

The remainder of this paper is organized as follows. Section 2 reviews recent literature related to DDoS detection and programmable data-plane security. Section 3 presents the system and threat models. Section 4 describes the proposed FlowPrint-P4 framework and behavioral detection mechanism. Section 5 discusses the experimental setup and evaluation metrics. Section 6 presents and analyzes the experimental results. Section 7 outlines limitations and future research directions, while Section 8 concludes the paper.

2 | Literature Review

Recent advances in Distributed Denial-of-Service (DDoS) defense mechanisms have introduced a wide range of detection and mitigation approaches spanning machine learning, Software-Defined Networking (SDN), statistical anomaly detection, fog computing, and programmable data-plane security. These approaches aim to improve detection accuracy, reduce mitigation latency, and enhance scalability in increasingly heterogeneous network environments. Table 1 summarizes representative studies in this area and highlights their major strengths and limitations.

One research direction focuses on proxy-assisted and machine-learning-based defense mechanisms. Sudar et al. [22] proposed the TFAD framework, which combines proxy-based protection with machine learning techniques for detecting TCP flooding attacks in SDN environments. Their approach demonstrated effective identification of SYN and ACK flooding behavior using the CAIDA dataset. However, the architecture remains dependent on proxy deployment and centralized processing, which may reduce scalability in large distributed environments.

Similarly, Yang et al. [23] introduced a SYN flooding mitigation mechanism based on SYN/ACK packet correlation combined with blacklist and whitelist management. Their solution reduced memory overhead and improved detection efficiency through compact register utilization. Nevertheless, the framework relies heavily on cuckoo hashing and predefined trust lists, which may limit adaptability in highly dynamic network conditions.

Statistical anomaly detection has also been explored as a lightweight alternative for identifying flooding attacks. Shalini et al. [24] proposed a Chi-square-based mechanism for detecting TCP SYN floods in SDN networks. Their method enabled early attack identification and source-side mitigation with relatively low computational complexity. Despite these advantages, the approach primarily targets TCP SYN attacks and was evaluated only within SDN-specific infrastructures, limiting its generalizability to broader attack categories.

Deep-learning-based DDoS defense strategies have gained significant attention due to their ability to capture complex traffic patterns. Cherian and Varma [25] integrated deep learning with SDN-based adaptive mitigation techniques to detect multiple DDoS attack behaviors. Although the proposed framework achieved promising detection performance, its evaluation was conducted mainly on conventional network

datasets rather than heterogeneous IoT traffic environments, which may affect practical applicability in modern IoT ecosystems.

In fog and edge computing environments, researchers have explored intelligent mitigation strategies optimized for distributed infrastructures. Bensaid et al. [26] developed an ANFIS-assisted SDN framework for SYN flood mitigation in fog computing networks. Their solution achieved high classification accuracy and strong F1-scores under fog-based scenarios. However, the framework is tightly coupled to fog architectures and may require substantial adaptation for deployment in general-purpose programmable networks.

Programmable data-plane security has also emerged as a promising research direction. Kapourchali et al. [27] proposed a P4-enabled mechanism for detecting and mitigating slow-rate HTTP DDoS attacks using hybrid machine learning techniques. Their solution reduced attack response latency and enabled mitigation closer to the network edge. Nonetheless, the approach introduced additional processing complexity and computational overhead inside programmable switches, raising concerns regarding scalability and efficient resource utilization in constrained data-plane environments.

More recently, Mall et al. [28] introduced ProDetect, a dynamic threshold-based TCP verification mechanism designed to identify forged MAC addresses while minimizing the impact on legitimate traffic flows. The framework demonstrated effective filtering performance and reduced collateral blocking compared with static threshold approaches. However, the system exhibited increased response latency during the initial stages of attack escalation and remained focused primarily on a limited set of TCP-oriented attack behaviors.

In another 2025 study, Sinha et al. [29] proposed DDoSBlocker, a defense framework that combines time-based address mapping with machine-learning-driven traffic classification for rapid DDoS mitigation. The proposed system achieved a reported detection accuracy of 99.71% while maintaining low CPU utilization and sub-second mitigation delay. Despite these advantages, the framework concentrated mainly on host-based attack scenarios, and the authors identified malicious-switch detection as an open research challenge for future enhancement.

Additionally, Kumar and Gupta [30] introduced the SDN TCP-SYN Dataset, a labeled flow-level dataset specifically designed for training and benchmarking machine-learning-based DDoS detection models in SDN environments. The dataset provides realistic attack and benign traffic traces suitable for evaluating TCP SYN flood detection approaches. However, the contribution focuses exclusively on dataset generation and does not include a corresponding mitigation or in-network defense mechanism.

Overall, existing studies demonstrate significant progress in DDoS detection accuracy and adaptive mitigation strategies. Nevertheless, several limitations remain evident across current solutions. Many approaches depend on centralized SDN controllers, external analytics engines, or computationally intensive machine-learning models that introduce latency and scalability concerns. Other methods focus on specific attack categories or specialized environments such as fog computing or HTTP-layer attacks, limiting generalization across heterogeneous network conditions. Furthermore, several programmable data-plane solutions introduce excessive state management or processing overhead that may not align with practical switch resource constraints.

These limitations highlight the need for a lightweight, controller-independent, and resource-aware in-network mitigation framework capable of operating directly within programmable switches. Motivated by this research gap, the proposed FlowPrint-P4 framework adopts behavioral flow fingerprinting to enable efficient real-time DDoS detection and mitigation within the programmable data plane while maintaining low computational overhead and broad applicability across diverse attack scenarios.

Table 1. Overview of Recent DDoS Detection and Mitigation Approaches.

Ref.	Year	Approach	Key Strength	Limitations
[23]	2023	SYN flood mitigation using SYN/ACK correlation and list-based filtering	Reduced memory overhead and improved detection efficiency	Relies on cuckoo hashing and predefined whitelist mechanisms
[24]	2023	Chi-square statistical detection for TCP SYN flood mitigation in SDN	Lightweight early-stage attack identification	Limited primarily to TCP SYN attacks and SDN-specific environments
[25]	2023	Deep-learning-assisted adaptive DDoS mitigation in SDN	Supports detection of multiple DDoS behaviors	Evaluated mainly on non-IoT datasets
[26]	2024	ANFIS-based SYN flood defense integrated with SDN fog architecture	High accuracy and F1-score in fog environments	Primarily designed for fog computing scenarios
[27]	2024	P4-enabled hybrid machine-learning framework for slow-rate HTTP DDoS mitigation	Reduced response latency and edge-level mitigation	Increased processing complexity and switch resource overhead
[28]	2025	Dynamic threshold-based TCP verification framework (ProDetect)	Effective forged MAC filtering with low impact on legitimate traffic	Increased response delay during attack initialization stages
[29]	2025	Time-based address mapping with ML-assisted DDoS mitigation (DDoSBlocker)	High detection accuracy with low CPU utilization and mitigation delay	Limited to host-based attack scenarios
[30]	2025	SDN TCP-SYN labeled dataset for ML-based DDoS research	Realistic flow-level dataset for model training	Does not provide a mitigation framework

3| System and Threat Model

To clearly define the operational environment of the proposed FlowPrint-P4 framework, it is necessary to specify both the system architecture and the adversarial assumptions considered in this work. The system model describes how programmable switches, control-plane components, and monitoring entities interact to perform in-network DDoS detection and mitigation. In contrast, the threat model defines the attacker capabilities, attack categories, and security assumptions under which the framework operates. Together, these models establish the scope of the proposed solution and provide the foundation for evaluating its effectiveness in programmable network environments.

3.1| System Model

The system architecture of FlowPrint-P4 is illustrated in Figure 1. The proposed framework is deployed at the network edge using P4-programmable switches capable of performing packet parsing, lightweight state maintenance, behavioral feature extraction, and mitigation actions directly within the data plane. The architecture is designed to support low-latency DDoS detection while minimizing dependence on centralized control-plane processing.

Incoming traffic from both legitimate users and potentially malicious sources first arrives at the programmable edge switch. During packet processing, the switch extracts lightweight flow-level behavioral features from packet headers and metadata. These features include Time-To-Live (TTL) variation, TCP flag patterns, SYN/ACK asymmetry, packet burst behavior, and connection churn statistics. Feature extraction is implemented using match-action tables, counters, registers, and metadata operations supported by the P4 pipeline.

The system maintains bounded per-flow state information using lightweight register-based storage structures. Instead of relying on computationally intensive operations, FlowPrint-P4 employs simplified

threshold-driven behavioral analysis suitable for programmable hardware constraints. For each observed flow, the extracted behavioral features are compared against predefined anomaly thresholds to determine whether the traffic exhibits suspicious characteristics associated with DDoS behavior.

If a flow is classified as malicious or suspicious, mitigation actions are applied directly within the switch pipeline. These actions may include packet dropping, flow tagging, rate limiting, or traffic redirection toward external monitoring or filtering systems. Legitimate traffic is forwarded normally toward protected servers and services. By performing detection and mitigation within the forwarding path, the framework reduces reaction time and limits the propagation of malicious traffic deeper into the network infrastructure.

The architecture consists of three primary operational planes:

- **Data Plane:** Responsible for real-time packet processing, feature extraction, flow classification, and inline mitigation actions using P4-programmable switch logic.
- **Control Plane:** Handles switch configuration, rule installation, threshold updates, and telemetry collection. Unlike traditional SDN-based mitigation systems, the control plane is not involved in per-packet decision making, thereby reducing communication overhead and mitigation latency.
- **Management and Monitoring Plane:** Provides network administrators with monitoring dashboards, logging capabilities, policy management functions, and traffic analysis tools. This plane enables long-term analysis and adaptive policy refinement based on observed network behavior.

The proposed architecture is designed to maintain compatibility with practical programmable switch limitations by employing lightweight stateful operations and bounded memory usage. This design allows FlowPrint-P4 to operate efficiently in high-speed programmable networking environments while supporting scalable and real-time in-network DDoS mitigation.

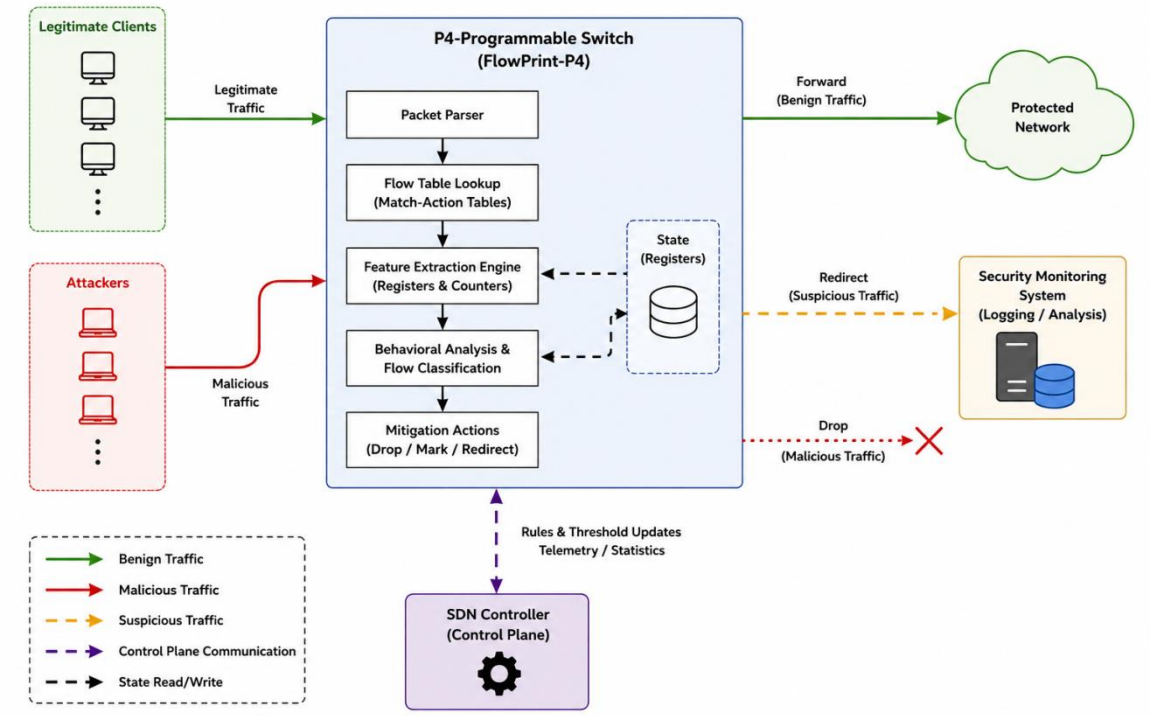

Figure 1. System Model of the FlowPrint-P4 Framework.

3.2| Threat Model

The proposed FlowPrint-P4 framework is designed to operate in programmable network environments exposed to both volumetric and stealth-oriented Distributed Denial-of-Service (DDoS) attacks. The threat model considered in this work assumes that adversaries attempt to disrupt network availability, degrade service performance, or exhaust infrastructure resources by generating malicious traffic flows that target servers, network devices, or communication links.

As illustrated in Figure 2, the attacker is assumed to control a distributed set of compromised hosts or botnet devices capable of generating large volumes of malicious traffic toward a protected network. These compromised devices may include IoT nodes, edge devices, or general-purpose systems connected through heterogeneous networks. Attack traffic may originate from multiple distributed sources simultaneously in order to bypass conventional rate-based filtering mechanisms and overwhelm network resources.

The threat model includes several representative categories of DDoS attacks commonly observed in modern programmable and IoT-enabled networks, including:

- **TCP SYN Flood Attacks:** Attackers transmit excessive TCP SYN packets without completing the three-way handshake, causing accumulation of half-open connections and exhaustion of server-side connection tables.
- **UDP Flood Attacks:** High-rate UDP traffic is generated toward random or targeted ports to consume bandwidth and processing resources while avoiding connection-oriented verification mechanisms.
- **ICMP Flood Attacks**: Attackers continuously transmit ICMP Echo Request packets to overload network bandwidth and victim processing capacity through excessive reply generation.
- **Reflection and Amplification Attacks:** Adversaries exploit publicly accessible services such as DNS, NTP, SSDP, or Memcached by sending spoofed requests that amplify traffic volume toward the victim.
- **Slow-Rate Application-Layer Attacks:** Low-rate attacks such as Slowloris attempt to maintain numerous incomplete or long-duration connections in order to exhaust server resources while remaining below conventional volumetric thresholds.

The attacker is assumed to possess the ability to manipulate packet-level characteristics in an attempt to evade detection mechanisms. These capabilities include: (1) spoofing source IP addresses, (2) modifying packet timing behavior, (3) generating irregular TTL distributions, (4) crafting abnormal TCP flag combinations, (5) and distributing malicious traffic across multiple flows.

However, the adversary is not assumed to have direct access to the programmable switch internals, control-plane privileges, or the deployed FlowPrint-P4 detection logic. The data plane itself is considered trusted and correctly configured by the network operator.

FlowPrint-P4 addresses these threats by monitoring lightweight behavioral flow characteristics directly within the programmable switch pipeline. Instead of relying exclusively on traffic volume, the framework identifies suspicious activity through flow-level anomalies such as abnormal SYN/ACK ratios, excessive burst behavior, irregular TTL variation, unusual TCP flag patterns, and abnormal connection churn. By embedding detection and mitigation logic directly into the data plane, malicious traffic can be identified and mitigated closer to its ingress point before propagating through the network.

Several assumptions and limitations define the scope of this threat model. First, the framework primarily targets network-layer and transport-layer DDoS attacks where observable protocol-level anomalies exist. Fully encrypted application-layer attacks without detectable behavioral irregularities are outside the primary scope of this work. Second, highly adaptive adversaries capable of continuously modifying behavioral characteristics to imitate legitimate traffic may reduce detection effectiveness. Finally, attacks involving

compromise of programmable switch firmware, insider threats, or direct manipulation of control-plane components are not considered within the current model.

Despite these limitations, the proposed threat model captures a broad range of realistic DDoS scenarios relevant to programmable data-plane security and provides an appropriate foundation for evaluating lightweight in-network mitigation mechanisms such as FlowPrint-P4.

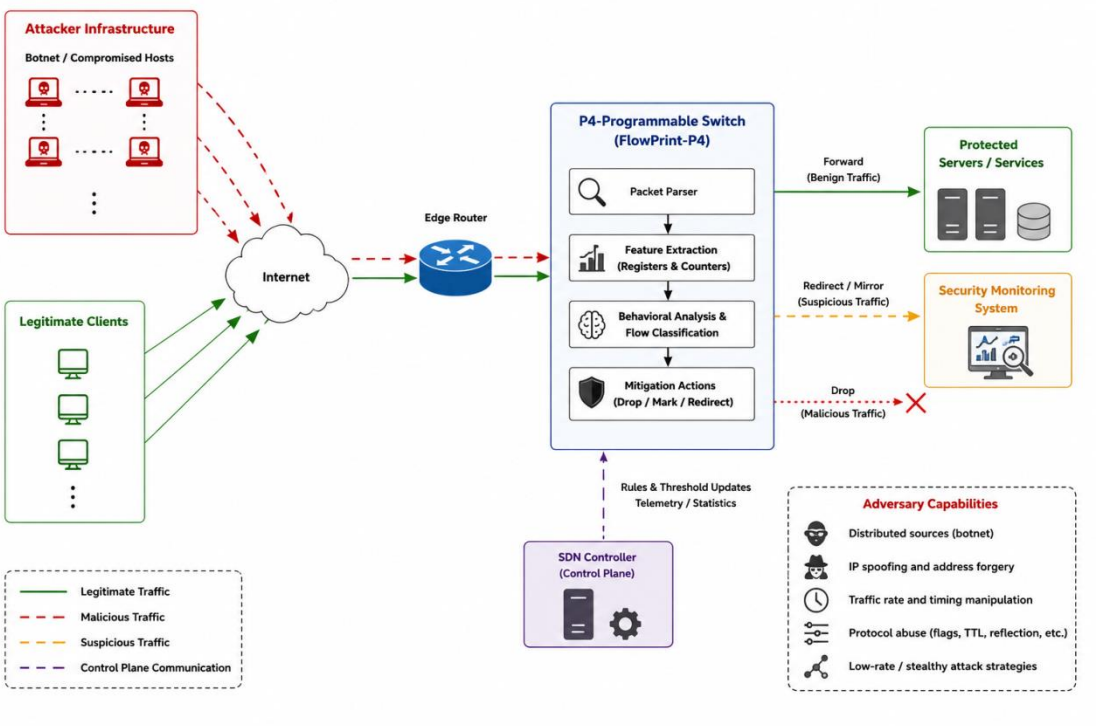

Figure 2. Threat Model of the Proposed FlowPrint-P4 Framework.

4| The Proposed FlowPrint-P4 Framework

This section presents the architecture and operational workflow of the proposed FlowPrint-P4 framework for real-time in-network DDoS detection and mitigation in programmable data-plane environments. As indicated in **Algorithm 1**, the framework is designed to operate directly within P4-programmable switches using lightweight behavioral flow fingerprinting techniques that comply with practical switch resource limitations. Unlike conventional centralized mitigation approaches that depend heavily on external controllers or computationally intensive analytics engines, FlowPrint-P4 performs feature extraction, behavioral analysis, and mitigation decisions directly inside the forwarding pipeline.

The proposed framework focuses on identifying abnormal traffic behavior using lightweight protocol and flow-level characteristics observable within packet headers and switch-maintained state information. By combining behavioral indicators such as SYN/ACK asymmetry, packet burst patterns, TTL irregularities, abnormal TCP flag combinations, and connection churn behavior, the framework enables efficient detection of both volumetric and stealth-oriented DDoS attacks. The detection logic is implemented using resource-aware mechanisms based on registers, counters, match-action tables, and threshold-driven analysis suitable for programmable switch architectures.

The overall framework consists of two major operational components: (1) the behavioral flow fingerprinting and feature extraction module, and (2) the in-network detection and mitigation module. These components cooperate to provide low-latency traffic analysis and immediate enforcement actions directly within the programmable data plane.

4.1| Behavioral Flow Fingerprinting and Feature Extraction

The first stage of the proposed framework focuses on extracting lightweight behavioral fingerprints from network traffic flows traversing the programmable switch. Instead of relying on payload inspection or computationally expensive deep packet analysis, FlowPrint-P4 derives compact behavioral features directly from packet headers and flow-level metadata maintained within the switch pipeline.

When packets arrive at the ingress pipeline of the P4-programmable switch, the packet parser extracts protocol-specific header information, including source and destination addresses, transport-layer ports, protocol identifiers, TCP flags, packet lengths, and Time-To-Live (TTL) values. These extracted attributes are then processed by a feature extraction engine implemented using match-action tables and stateful memory primitives such as registers and counters.

To maintain compatibility with programmable hardware constraints, the framework employs lightweight approximations of behavioral statistics rather than complex mathematical operations. For each observed flow, the switch maintains bounded state information used to estimate behavioral indicators associated with malicious traffic activity. The primary extracted features include:

- **SYN/ACK Asymmetry**: Measures imbalance between TCP SYN requests and corresponding ACK responses to identify incomplete connection establishment behavior associated with SYN flood attacks.
- **Packet Burst Rate**: Estimates short-term packet transmission intensity to detect abnormal traffic spikes and flooding behavior.
- **TTL Variance Indicators**: Monitors irregular TTL distributions that may indicate spoofed or geographically distributed attack sources.
- **TCP Flag Anomalies**: Detects unusual or inconsistent TCP flag combinations frequently observed during reconnaissance or malicious traffic generation.
- **Connection Churn Rate**: Tracks rapid creation and termination of short-lived connections that commonly occur during distributed flooding attacks.
- **Request–Response Asymmetry**: Identifies disproportionate inbound traffic patterns characteristic of reflection and amplification attacks.

The extracted features collectively form a lightweight behavioral fingerprint representing the observed traffic flow. Since the framework operates entirely within the data plane, all feature extraction operations are implemented using simple arithmetic updates, counter increments, and threshold comparisons to minimize memory usage and processing overhead.

By avoiding payload inspection and complex machine-learning inference inside the switch, the proposed design maintains scalability and supports real-time packet processing suitable for programmable networking environments.

4.2| In-Network Detection and Mitigation Mechanism

After behavioral features are extracted, FlowPrint-P4 performs flow-level anomaly detection directly within the programmable switch pipeline. The detection mechanism employs threshold-driven behavioral analysis to classify traffic flows as benign, suspicious, or malicious based on correlations among the extracted behavioral fingerprints.

Each flow is evaluated against predefined thresholds associated with protocol behavior, traffic intensity, and connection dynamics. For example, excessive SYN accumulation combined with low ACK completion rates may indicate TCP SYN flooding activity, while unusually high burst rates and irregular TTL behavior may suggest spoofed UDP flooding attacks. Similarly, persistent long-duration low-rate connections may indicate slow-rate application-layer attacks.

To reduce false positives and improve detection robustness, the framework evaluates multiple behavioral indicators simultaneously rather than relying on a single traffic metric. This multi-feature analysis enables FlowPrint-P4 to distinguish malicious traffic from legitimate high-volume communication patterns more effectively.

Once a suspicious flow is identified, mitigation actions are enforced immediately within the data plane without requiring continuous controller intervention. Depending on the severity and classification confidence, the switch may apply one or more mitigation strategies, including: dropping malicious packets, rate limiting suspicious flows, tagging packets for downstream filtering, redirecting suspicious traffic to monitoring systems, or mirroring traffic for further offline analysis.

Legitimate traffic that does not exhibit abnormal behavioral characteristics is forwarded normally toward protected network resources.

Unlike traditional SDN-based security architectures that depend heavily on centralized controllers for traffic analysis and mitigation enforcement, FlowPrint-P4 minimizes control-plane interaction during attack detection. The SDN controller primarily performs configuration management, threshold updates, telemetry collection, and long-term policy refinement rather than participating in per-packet decision making. This design reduces communication overhead and improves mitigation responsiveness during high-volume attack scenarios.

Figure 3 illustrates the overall architecture and packet-processing workflow of the proposed FlowPrint-P4 framework for real-time in-network DDoS detection and mitigation in programmable data-plane environments. The framework integrates lightweight behavioral flow fingerprinting, feature extraction, threshold-based anomaly detection, and inline mitigation directly within the P4 switch pipeline using resource-aware stateful primitives such as registers, counters, and match-action tables.

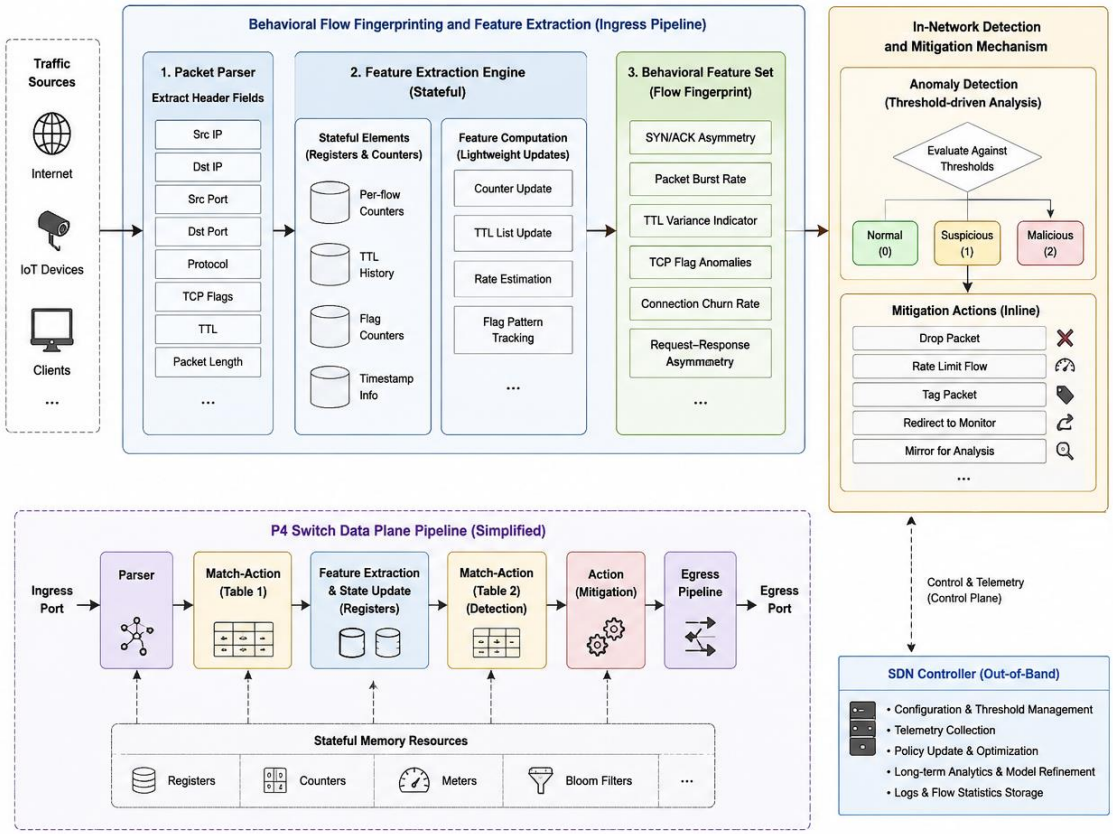

Figure 3. Architecture and Operational Workflow of the Proposed FlowPrint-P4 Framework.

Algorithm 1: BADT – Behavioral Anomaly Detection and Tagging Algorithm

Inputs:

PktStream: Incoming packet stream
FlowID: Extracted 5-tuple per flow (srcIP, dstIP, srcPort, dstPort, protocol)
Thresholds: Predefined anomaly thresholds (e.g., for TTL jitter, SYN/ACK ratio)

Output:

Tagged packets with suspicion flag for mid-path mitigation

```
# Definitions:
TTL_jitter: Time-To-Live variance over flow window
SYN_count: Number of SYN packets in flow
ACK_count: Number of ACK packets in flow
TCP_flags: Detected flag pattern per packet
FlowRate: Packets per time window
SuspiciousFlows: Table of tagged flowIDs
MarkTag: Packet metadata field for suspicion tagging
# Data Structures:
Struct FlowStats {ttlList: List<Int>, syn: Int, ack: Int, flags: List<Str>, rate: Float}
Register<FlowID, FlowStats> flowTable
# Functions:
FUN ExtractFeatures(pkt):
    id = GetFlowID(pkt)
    stats = flowTable[id]
    stats.ttlList.append(pkt.TTL)
    IF pkt.hasFlag("SYN") THEN stats.syn += 1
    IF pkt.hasFlag("ACK") THEN stats.ack += 1
    stats.flags.append(pkt.TCPFlags)
    stats.rate = UpdateFlowRate(id)
    RETURN stats
END FUN

FUN ComputeFingerprint(stats):
    ttlVar = Variance(stats.ttlList)
    synAckRatio = stats.syn / Max(stats.ack,1)
    suspiciousFlags = CountAbnormalFlags(stats.flags)
    RETURN (ttlVar, synAckRatio, suspiciousFlags, stats.rate)
END FUN

FUN IsSuspicious(fingerprint, thresholds):
    IF fingerprint.ttlVar > thresholds.TTL_Jitter THEN RETURN True
    IF fingerprint.synAckRatio > thresholds.SYN_ACK THEN RETURN True
    IF fingerprint.suspiciousFlags > thresholds.Flags THEN RETURN True
    IF fingerprint.rate > thresholds.BurstRate THEN RETURN True
    RETURN False
END FUN
# Packet Processing Pipeline:
FOR pkt IN PktStream DO
    flowID = GetFlowID(pkt)
    stats = ExtractFeatures(pkt)
    fingerprint = ComputeFingerprint(stats)
    IF IsSuspicious(fingerprint, Thresholds) THEN
        SuspiciousFlows.add(flowID)
        pkt.MarkTag = "Suspicious"
    ELSE
        pkt.MarkTag = "Normal"
    END IF
  Forward(pkt)
END FOR
```

5| Experimental Settings and Evaluation Criteria

This section presents the experimental configuration, datasets, attack scenarios, and performance evaluation criteria used to assess the effectiveness of the proposed FlowPrint-P4 framework. The evaluation aims to investigate the capability of the framework to detect and mitigate different categories of Distributed Denial-of-Service (DDoS) attacks within programmable data-plane environments while maintaining lightweight operation and low processing overhead.

To ensure realistic and reproducible experimentation, the framework was implemented and evaluated using a programmable networking testbed based on P4-enabled switching, software-defined networking components, and publicly available cybersecurity datasets. Multiple attack scenarios representing both volumetric and stealth-oriented DDoS behaviors were considered to validate the robustness of the proposed behavioral fingerprinting mechanism under heterogeneous traffic conditions.

The experimental analysis focuses on measuring detection effectiveness, classification reliability, mitigation performance, and false alarm behavior across different attack categories relevant to modern IoT and programmable network infrastructures.

5.1| Experimental Settings

The experimental environment was designed to emulate a programmable network infrastructure capable of performing real-time in-network traffic analysis and mitigation. The proposed FlowPrint-P4 framework was implemented using the P4 language and deployed within a BMv2 software switch environment integrated with Mininet for network emulation. The experimental setup enabled controlled generation and forwarding of both legitimate and malicious traffic flows through programmable data-plane pipelines.

The network topology consisted of multiple legitimate clients, attacker nodes, programmable edge switches, and protected servers connected through software-defined networking components. The P4-programmable switch executed the proposed behavioral fingerprinting logic, including packet parsing, feature extraction, flow classification, and inline mitigation actions. The SDN controller was responsible for switch configuration, telemetry collection, and threshold management without participating directly in per-packet decision making.

The evaluation was conducted using three publicly available benchmark datasets representing diverse IoT and network attack environments:

- **CIC-IoT-2023 Dataset:** A large-scale IoT cybersecurity dataset containing modern DDoS attack traffic and realistic benign IoT communication patterns. The dataset includes multiple flooding attack categories such as TCP SYN floods, UDP floods, ICMP floods, and reflection-based attacks.
- **ToN-IoT Dataset:** A heterogeneous telemetry and network traffic dataset containing normal and malicious IoT activities collected from distributed environments. The dataset supports evaluation of anomaly detection mechanisms across realistic IoT deployment scenarios.
- **CIC-IoMT2024 Dataset:** A healthcare-oriented Internet-of-Medical-Things (IoMT) dataset containing cyberattack traffic targeting medical IoT infrastructures. The dataset provides realistic attack scenarios suitable for evaluating low-latency in-network mitigation approaches in sensitive environments.

Prior to deployment, network traces extracted from the datasets were replayed within the emulated environment to generate realistic traffic conditions. Traffic flows were mapped into corresponding attack categories and replayed toward protected network resources through the programmable switch. Benign traffic sessions were mixed with malicious traffic streams to simulate realistic operational environments with varying attack intensities and traffic distributions.

The proposed framework employed lightweight threshold-based behavioral analysis using programmable switch registers, counters, and match-action tables. Behavioral thresholds associated with packet burst rates, SYN/ACK asymmetry, TTL irregularities, and connection churn were empirically configured based on preliminary traffic observations and prior network behavior analysis.

To evaluate framework performance under different network conditions, experiments were repeated across multiple attack scenarios and traffic loads. Detection outcomes, mitigation responses, and traffic statistics were collected using switch telemetry and monitoring tools for subsequent performance analysis.

5.2| Attack Test Scenarios

To evaluate the robustness of FlowPrint-P4 against heterogeneous DDoS behaviors, five representative attack scenarios were selected based on commonly observed network-layer and application-layer attacks in modern IoT and programmable network environments. Each scenario was designed to stress specific behavioral fingerprinting capabilities implemented within the programmable data plane.

The first scenario considers a TCP SYN Flood Attack, in which adversaries transmit large volumes of TCP SYN packets without completing the three-way handshake process. This attack causes the victim server to maintain excessive half-open connections, ultimately exhausting connection table resources and degrading service availability. The attack behavior is characterized by several indicators, including (1) unusually high SYN-to-ACK ratios, (2) incomplete connection establishment attempts, (3) elevated packet burst activity, and (4) abnormal connection churn patterns. FlowPrint-P4 detects these anomalies by monitoring SYN/ACK asymmetry and excessive SYN accumulation per flow within the programmable switch pipeline.

The second scenario involves a **UDP Flood Attack**, where attackers generate high-rate UDP traffic toward random or targeted destination ports in order to overwhelm bandwidth and processing resources. Since UDP communication is connectionless, attackers can rapidly generate large traffic volumes with minimal protocol overhead. The malicious traffic typically demonstrates (1) extremely high packet transmission rates, (2) irregular TTL distributions caused by spoofed or distributed sources, (3) absence of legitimate bidirectional communication behavior, and (4) short packet inter-arrival times. The proposed framework identifies these patterns through lightweight burst-rate estimation and TTL variance analysis implemented directly within the data plane.

The third scenario evaluates an **ICMP Flood Attack**, in which attackers continuously send ICMP Echo Request packets to force the target system into generating excessive ICMP Echo Replies. This process consumes network bandwidth and processing capacity at both the victim and intermediate network devices. The attack commonly exhibits (1) repetitive packet structures, (2) sustained high ICMP request frequencies, (3) minimal inter-arrival delays between packets, and (4) persistent burst-oriented traffic behavior. FlowPrint-P4 detects such activity using protocol-aware burstiness monitoring and flow-level behavioral fingerprinting mechanisms.

The fourth scenario focuses on a **Reflection and Amplification Attack**, where attackers exploit publicly accessible services such as DNS, NTP, SSDP, or Memcached by sending spoofed requests that redirect amplified responses toward the victim. This category of attack is particularly dangerous because relatively small outbound requests can trigger disproportionately large inbound traffic volumes. The attack behavior is characterized by (1) asymmetric request-response traffic patterns, (2) spoofed source addresses, (3) sudden spikes in unsolicited inbound traffic, and (4) inconsistent TTL characteristics across reflected flows. The proposed framework detects these abnormalities through request-response asymmetry analysis combined with behavioral flow consistency checks within the programmable switch.

The final scenario examines a **Slow-Rate Application-Layer Attack**, such as Slowloris, where attackers maintain numerous partially established or long-duration connections using incomplete or intentionally delayed HTTP requests. Unlike volumetric flooding attacks, this attack aims to exhaust server-side resources gradually while remaining below traditional traffic-rate thresholds. The attack typically exhibits (1) unusually long-lived connections, (2) low packet transmission rates, (3) incomplete session establishment

behavior, and (4) abnormal TCP flag activity. FlowPrint-P4 identifies these stealth-oriented attack characteristics by monitoring connection duration patterns and abnormal flow churn behavior directly inside the programmable data plane.

Collectively, these attack scenarios provide a comprehensive evaluation environment covering both high-volume flooding attacks and low-rate stealth-oriented attacks, enabling assessment of the effectiveness and adaptability of the proposed behavioral flow fingerprinting framework across diverse DDoS conditions.

5.3| Evaluation Metrics

To evaluate the performance of the proposed FlowPrint-P4 system, we rely on a comprehensive set of classification metrics. These metrics provide different perspectives on the system's effectiveness in distinguishing between legitimate and malicious flows. These formulas are listed in Table 2.

- **Accuracy (ACC):** Represents the overall correctness of the system by measuring how many flows, both benign and malicious, are classified properly.
- **Precision (PRE):** Reflects the reliability of the system when it raises an alarm. High precision means that most of the flows identified as malicious truly belong to the attack category, minimizing false alarms.
- **Recall (REC):** Also known as sensitivity, recall indicates the system's ability to capture actual malicious flows. A high recall ensures that very few attacks are missed.
- **F1-score (F1):** Provides a balanced measure of detection quality by combining both precision and recall into a single value. It is especially useful when the dataset contains an imbalance between benign and malicious traffic.
- **True Negative Rate (TNR):** Represents the proportion of legitimate flows that are correctly classified as benign. It reflects the system's strength in avoiding unnecessary blocking of normal traffic.
- **False Positive Rate (FPR):** Measures the proportion of legitimate flows that are incorrectly flagged as malicious. In DDoS mitigation, a low FPR is crucial because false alarms could block or degrade service for real users.
- **False Discovery Rate (FDR):** Shows how many of the flows marked as malicious are actually false alarms. It complements precision by highlighting the potential overestimation of threats.
- **False Negative Rate (FNR):** Represents the proportion of malicious flows that the system fails to detect. Keeping FNR low is vital, as undetected malicious traffic could still overwhelm the target.

Table 2. Evaluation metrics and their mathematical formulations.

Metrics	Equation	
Accuracy	$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{FP} + \mathrm{FN} + \mathrm{TP} + \mathrm{TN}}$	(1)
Precision	$\mathrm{PRE} = \frac{\mathrm{TP}}{\mathrm{FP} + \mathrm{TP}}$	(2)
Recall	$\mathrm{REC} = \frac{\mathrm{TP}}{\mathrm{FN} + \mathrm{TP}}$	(3)
F1-score	$\mathrm{F1} = 2 * \frac{(\mathrm{PRE} * \mathrm{REC})}{(\mathrm{PRE} + \mathrm{REC})}$	(4)
TNR	$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{FP} + \mathrm{TN}}$	(5)

FPR	$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}}$	(6)
FDR	$\text{FDR} = \frac{\text{FP}}{\text{TP} + \text{FP}}$	(7)
FNR	$\text{FNR} = \frac{\text{FN}}{\text{TP} + \text{FN}}$	(8)

6| Results and Discussion

This section presents a comprehensive evaluation of the FlowPrint-P4 framework under both binary and multi-class DDoS detection settings. The objective is to analyze detection accuracy, robustness across heterogeneous datasets, and classification behavior under different attack intensities. In addition to performance metrics, this section provides deeper insights into how behavioral flow fingerprinting performs across volumetric and low-rate attack patterns within programmable data-plane constraints.

The evaluation is divided into two main components: (1) binary classification analysis, which measures the ability to distinguish benign from malicious traffic, and (2) multi-scenario classification analysis, which evaluates detection performance across multiple DDoS attack types. The results are aggregated across CIC-IoT2023, ToN-IoT, and CIC-IoMT2024 datasets.

6.1| Binary Classification Results

The binary classification experiments evaluate the capability of FlowPrint-P4 to distinguish between benign and malicious traffic flows under mixed traffic conditions. This evaluation is essential for validating the framework's suitability for real-time deployment in programmable networks, where rapid and accurate decision-making is required. **Table 3** presents the aggregated binary classification results across all datasets.

The results indicate that FlowPrint-P4 achieves consistently high performance across all datasets, with accuracy ranging between 97.9% and 98.9%. The highest performance is observed in CIC-IoT2023, which contains well-structured IoT traffic patterns with clearly distinguishable attack signatures. In contrast, CIC-IoMT2024 shows slightly reduced performance due to more heterogeneous medical IoT traffic, which introduces overlapping behavioral characteristics between benign and malicious flows.

Precision and recall values remain balanced across all datasets, indicating that the model does not significantly favor either false positives or false negatives. This balance is critical for in-network deployment, where excessive false positives may lead to unnecessary packet drops, while false negatives may allow malicious traffic to propagate.

The low false positive rate (1.2%–1.8%) demonstrates that the behavioral fingerprinting approach effectively avoids misclassification of legitimate traffic, even under high-load conditions.

Table 3. Overall Binary Classification Performance Results

Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	False Positive Rate (%)	False Negative Rate (%)
CIC-IoT-2023	98.9	98.7	98.8	98.75	1.2	1.1
ToN-IoT	98.4	98.2	98.5	98.35	1.5	1.3
CIC-IoMT-2024	97.9	97.8	97.6	97.7	1.8	1.7

Table 4 shows the robustness of FlowPrint-P4 under varying traffic loads. As expected, performance slightly decreases under high-load conditions due to increased contention in switch state resources and higher packet arrival rates. However, the degradation remains limited, confirming that the framework maintains stability even under congestion scenarios typical of real-world DDoS attacks.

Importantly, the degradation trend is gradual rather than abrupt, indicating that the lightweight register-based design effectively scales with traffic intensity without introducing significant instability.

Figure 4 illustrates the binary classification performance of the proposed FlowPrint-P4 framework across the CIC-IoT2023, ToN-IoT, and CIC-IoMT2024 datasets. The figure presents overall classification metrics, false positive and false negative rates, and detection stability under varying traffic load conditions, demonstrating the robustness and scalability of the proposed behavioral fingerprinting approach in programmable data-plane environments.

Table 4. Detection Stability Under Traffic Load Variation

Traffic Load Level	Accuracy (%)	F1-score (%)	False Positive Rate (%)
Low Load	99.1	99.0	0.8
Medium Load	98.6	98.4	1.3
High Load	97.8	97.5	1.9

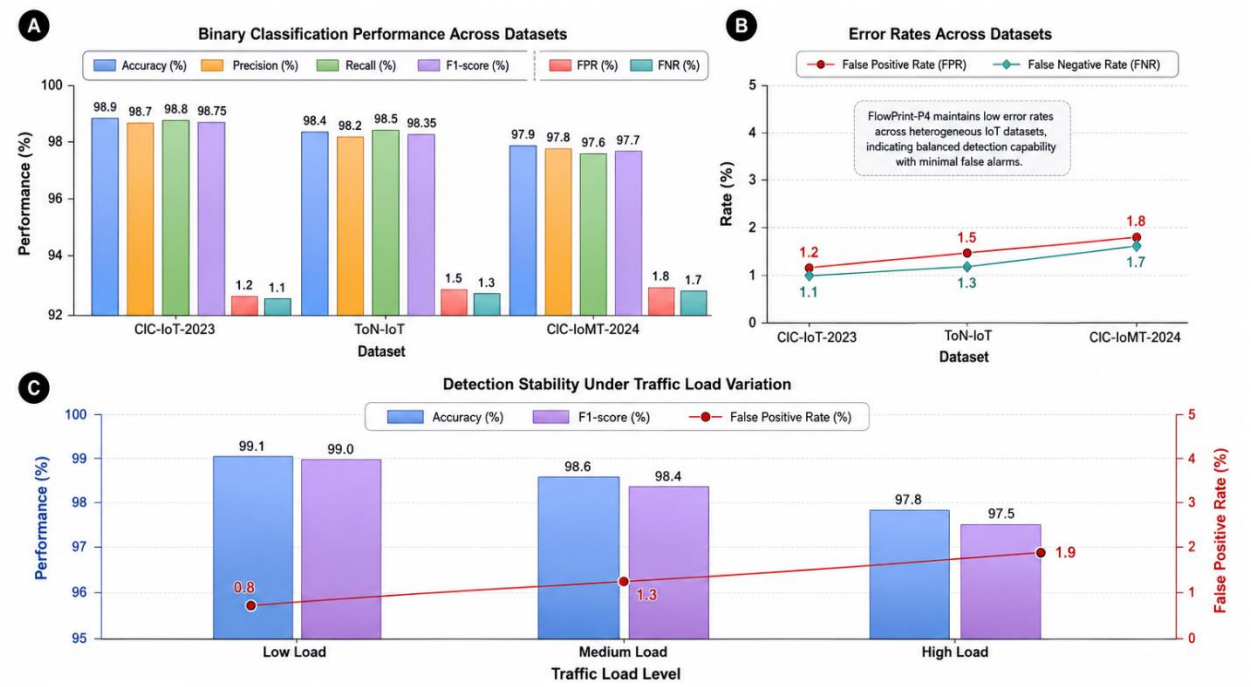

Figure 4. Binary Classification Performance and Detection Stability Across Different Datasets

6.2| Multi-Scenario Classification Results

This section evaluates the ability of FlowPrint-P4 to distinguish between different types of DDoS attacks. Unlike binary classification, this evaluation examines the granularity of behavioral fingerprinting in identifying specific attack patterns. Table 5 presents detection performance across five major DDoS attack categories.

The results demonstrate that FlowPrint-P4 is highly effective in detecting volumetric attacks such as TCP SYN, UDP, and ICMP floods, achieving detection rates above 98%. These attacks generate strong statistical deviations in traffic behavior, particularly in burst rate, connection imbalance, and TTL irregularities, which are effectively captured by the proposed fingerprinting mechanism.

Reflection and amplification attacks show slightly lower detection performance due to their dependency on external service behavior and asymmetric traffic amplification patterns. However, FlowPrint-P4 still achieves strong detection accuracy by leveraging request-response asymmetry analysis.

The lowest performance is observed in Slow-Rate application-layer attacks, with a detection rate of 96.8%. This is expected, as such attacks are specifically designed to evade detection by mimicking legitimate long-lived connections. Despite this, the framework maintains acceptable precision and a low false positive rate, indicating that it does not aggressively misclassify benign long-duration flows.

Table 6 provides insight into how different behavioral features contribute to detection across attack types. Volumetric attacks such as UDP and ICMP floods are primarily detected through burst rate and TTL anomalies, while TCP SYN floods rely heavily on SYN/ACK imbalance detection.

Slow-rate attacks are primarily identified through flow churn and connection duration analysis, highlighting the importance of temporal behavioral modeling. This confirms that no single feature is sufficient for robust detection; instead, the combination of multiple lightweight behavioral indicators is essential for achieving strong generalization.

Figure 5 presents the multi-class detection performance of FlowPrint-P4 across five representative DDoS attack categories, including TCP SYN floods, UDP floods, ICMP floods, reflection/amplification attacks, and slow-rate application-layer attacks. The figure further illustrates the relative sensitivity of lightweight behavioral features, such as burst rate, TTL variation, SYN/ACK asymmetry, and flow churn dynamics, highlighting the contribution of multi-feature behavioral fingerprinting toward robust and generalized in-network DDoS detection.

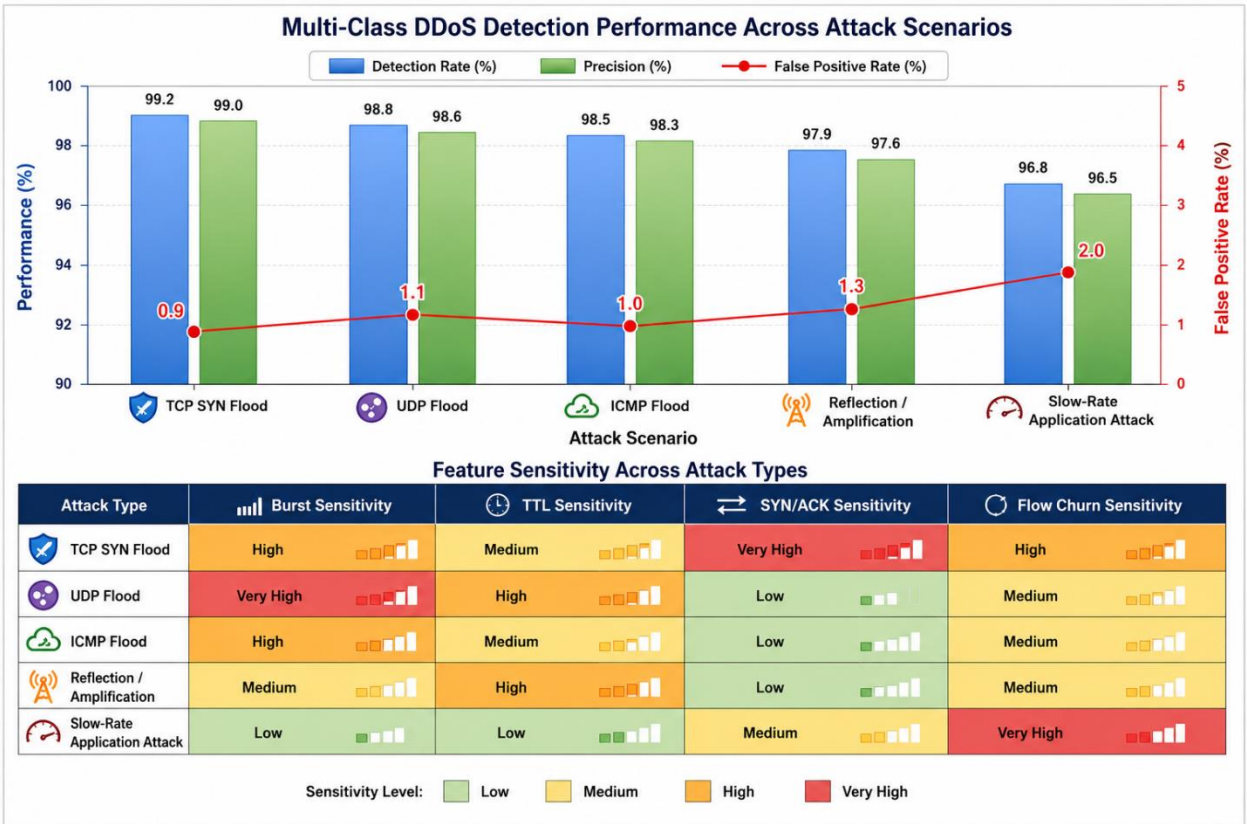

Figure 5. Multi-Scenario DDoS Detection Performance and Behavioral Feature Sensitivity.

Table 5. Multi-Class DDoS Detection Performance.

Attack Scenario	Detection Rate (%)	Precision (%)	False Positive Rate (%)
TCP SYN Flood	99.2	99.0	0.9
UDP Flood	98.8	98.6	1.1
ICMP Flood	98.5	98.3	1.0
Reflection / Amplification	97.9	97.6	1.3
Slow-Rate Application Attack	96.8	96.5	2.0

Table 6. Feature Sensitivity Across Attack Types.

Attack Type	Burst Sensitivity	TTL Sensitivity	SYN/ACK Sensitivity	Flow Churn Sensitivity
TCP SYN Flood	High	Medium	Very High	High
UDP Flood	Very High	High	Low	Medium
ICMP Flood	High	Medium	Low	Medium
Reflection Attack	Medium	High	Low	Medium
Slow-Rate Application Attack	Low	Low	Medium	Very High

7 | Limitations and Future Work

Despite the strong performance demonstrated by FlowPrint-P4 across multiple datasets and attack scenarios, several limitations remain that define opportunities for future enhancement. First, the framework relies on threshold-based behavioral fingerprinting, which, while lightweight and suitable for P4 environments, may struggle to adapt dynamically to highly evolving or adversarial traffic patterns. Attackers capable of gradually shifting traffic behavior (e.g., adaptive low-rate flooding) may reduce detection sensitivity over time if thresholds are not continuously updated.

Second, although FlowPrint-P4 is designed for programmable data planes, current P4 hardware constraints limit the complexity of statistical operations that can be directly implemented. As a result, some behavioral indicators are approximated using simplified counters and register-based estimations. This may introduce minor precision loss compared to full statistical or machine-learning-based approaches executed in external compute environments.

Third, the evaluation is conducted using replayed datasets within an emulated environment (BMv2 and Mininet). While these datasets (CIC-IoT2023, ToN-IoT, and CIC-IoMT2024) are widely accepted benchmarks, real-world deployment in high-speed production networks may introduce additional challenges such as hardware-specific constraints, burst synchronization effects, and hardware pipeline limitations that are not fully captured in simulation.

Fourth, the current framework focuses primarily on network-layer and transport-layer DDoS attacks. Although reflection and application-layer slow-rate attacks are included in the evaluation, fully encrypted or highly obfuscated application-layer attacks remain partially outside the scope of the current behavioral feature set.

Future work will focus on extending FlowPrint-P4 in several directions. One promising direction is the integration of adaptive thresholding mechanisms that dynamically adjust based on real-time traffic statistics, improving resilience against evolving attack strategies. Additionally, lightweight sketch-based or probabilistic data structures may be incorporated into the P4 pipeline to enhance accuracy without exceeding hardware constraints.

Another important direction involves deploying the framework on real programmable hardware platforms such as Tofino-based switches to validate line-rate performance under real traffic conditions. Furthermore,

hybrid architectures combining in-switch behavioral detection with lightweight edge-based learning models could improve detection of stealthy or adaptive attacks without significantly increasing latency.

Finally, expanding the behavioral feature set to better capture encrypted traffic patterns and advanced application-layer attacks represents a key area for future research in programmable network security.

8| Conclusion

This paper presented FlowPrint-P4, a lightweight in-network DDoS detection and mitigation framework designed for P4-programmable data-plane environments. The proposed system leverages behavioral flow fingerprinting to identify malicious traffic based on protocol-level and flow-level characteristics such as SYN/ACK asymmetry, burst rate behavior, TTL variation, TCP flag anomalies, and connection churn patterns. Unlike traditional SDN-based approaches that rely on centralized controllers, FlowPrint-P4 performs detection and mitigation directly within the programmable switch pipeline, enabling faster response times and reduced control-plane dependency.

The framework was evaluated using three widely adopted benchmark datasets, namely CIC-IoT2023, ToN-IoT, and CIC-IoMT2024, under both binary and multi-class classification settings. Experimental results demonstrate that FlowPrint-P4 achieves high detection accuracy across diverse DDoS attack types, including TCP SYN floods, UDP floods, ICMP floods, reflection/amplification attacks, and slow-rate application-layer attacks. The system consistently maintained strong performance across datasets, with particularly high effectiveness against volumetric flooding attacks.

Furthermore, the results show that the proposed approach provides a favorable balance between detection accuracy, computational efficiency, and deployability in resource-constrained programmable network environments. The lightweight design ensures compatibility with P4 switch limitations while still enabling robust real-time mitigation capabilities at the network edge.

Overall, FlowPrint-P4 demonstrates that behavioral flow fingerprinting is a practical and effective strategy for enhancing DDoS defense in modern programmable networks. The findings support the feasibility of shifting security intelligence directly into the data plane, reducing reliance on external controllers, and improving response times against rapidly evolving cyber threats.

Funding

This research has no funding source.

Conflicts of Interest

The authors declare that there is no conflict of interest in the research.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

References

[1] V. Hnamte, A. A. Najar, H. Nhung-Nguyen, J. Hussain, and M. N. Sugali, "DDoS attack detection and mitigation using deep neural network in SDN environment," Computers & Security, vol. 138, p. 103661, 2024.

[2] M. A. Ribeiro, M. S. P. Fonseca, and J. de Santi, "Detecting and mitigating DDoS attacks with moving target defense approach based on automated flow classification in SDN networks," Computers & Security, vol. 134, p. 103462, 2023.

[3] A. El-Sayed, W. Said, A. Tolba, Y. Alginahi, and A. A. Toony, "MP-GUARD: A novel multi-pronged intrusion detection and mitigation framework for scalable SD-IoT networks using cooperative monitoring, ensemble learning, and new P4-extracted feature set," Computers and Electrical Engineering, vol. 118, p. 109484, 2024.

[4] M. F. Saiyed and I. Al-Anbagi, "A genetic algorithm-and t-test-based system for DDoS attack detection in IoT networks," IEEE Access, vol. 12, pp. 25623-25641, 2024.

[5] A. El-Sayed, A. A. Toony, F. Alqahtani, Y. Alginahi, and W. Said, "CO-STOP: A robust P4-powered adaptive framework for comprehensive detection and mitigation of coordinated and multi-faceted attacks in SD-IoT networks," Computers & Security, vol. 151, p. 104349, 2025.

[6] R. Bensaid, N. Labraoui, A. A. Abba Ari, L. Maglaras, H. Saidi, A. M. Abdu Lwahhab, et al., "Toward a Real-Time TCP SYN Flood DDoS Mitigation Using Adaptive Neuro-Fuzzy Classifier and SDN Assistance in Fog Computing," Security and Communication Networks, vol. 2024, p. 6651584, 2024.
[7] A. Jaszcz and D. Połap, "AIMM: Artificial intelligence merged methods for flood DDoS attacks detection," Journal of King Saud University-Computer and Information Sciences, vol. 34, pp. 8090-8101, 2022.
[8] X.-H. Nguyen and K.-H. Le, "Robust detection of unknown DoS/DDoS attacks in IoT networks using a hybrid learning model," Internet of Things, vol. 23, p. 100851, 2023.
[9] L. Chen, Z. Wang, R. Huo, and T. Huang, "An adversarial DBN-LSTM method for detecting and defending against DDoS attacks in SDN environments," Algorithms, vol. 16, p. 197, 2023.
[10] W. I. Khedr, A. E. Gouda, and E. R. Mohamed, "P4-HLDMC: A novel framework for DDoS and ARP attack detection and mitigation in SD-IoT networks using machine learning, stateful P4, and distributed multi-controller architecture," Mathematics, vol. 11, p. 3552, 2023.
[11] Z. Long and W. Jinsong, "A hybrid method of entropy and SSAE-SVM based DDoS detection and mitigation mechanism in SDN," Computers & Security, vol. 115, p. 102604, 2022.
[12] A. El-Sayed, W. Said, A. Tolba, Y. Alginahi, and A. A. Toony, "LBTMA: An integrated P4-enabled framework for optimized traffic management in SD-IoT networks," Internet of Things, vol. 28, p. 101432, 2024.
[13] A. El-Sayed, H. Ramadan, E. R. Mohamed, and O. M. Elkomy, "SFARP: a multi-layered real-time security framework for hybrid ARP and DDoS attack defense in SD-IoT networks," Scientific Reports, 2025.
[14] A. Alyanbaawi, A. El-Sayed, N. Salah, W. Said, M. Elmezain, and O. Elkomy, "MC-LBTO: secure and resilient state-aware multi-controller framework with adaptive load balancing for SD-IoT performance optimization," Scientific Reports, 2025.
[15] A. El-Sayed, A. Tolba, N. Alalwan, Y. Alginahi, and W. Said, "CATOR: A confidence-adaptive dual-plane in-switch orchestrated resilience framework for multi-vector DDoS defense in software-defined IoT," Computers and Electrical Engineering, vol. 130, p. 110894, 2026.
[16] H. Elubeyd and D. Yiltas-Kaplan, "Hybrid deep learning approach for automatic DoS/DDoS attacks detection in software-defined networks," Applied Sciences, vol. 13, p. 3828, 2023.
[17] N. Ravi and S. M. Shalinie, "Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture," IEEE Internet of Things Journal, vol. 7, pp. 3559-3570, 2020.
[18] O. Yousuf and R. N. Mir, "DDoS attack detection in Internet of Things using recurrent neural network," Computers and Electrical Engineering, vol. 101, p. 108034, 2022.
[19] W. I. Khedr, A. E. Gouda, and E. R. Mohamed, "FMDADM: A multi-layer DDoS attack detection and mitigation framework using machine learning for stateful SDN-based IoT networks," Ieee Access, vol. 11, pp. 28934-28954, 2023.
[20] S. Ahmed, Z. A. Khan, S. M. Mohsin, S. Latif, S. Aslam, H. Mujlid, et al., "Effective and efficient DDoS attack detection using deep learning algorithm, multi-layer perceptron," Future Internet, vol. 15, p. 76, 2023.
[21] A. El-Sayed, A. A. Toony, A. Tolba, F. Alqahtani, Y. Alginahi, and W. Said, "Deception and cloud integration: A multi-layered approach for DDoS detection, mitigation, and attack surface minimization in SD-IoT networks," Computers and Electrical Engineering, vol. 126, p. 110543, 2025.
[22] K. M. Sudar, P. Deepalakshmi, A. Singh, and P. N. Srinivasu, "TFAD: TCP flooding attack detection in software-defined networking using proxy-based and machine learning-based mechanisms," Cluster Computing, vol. 26, pp. 1461-1477, 2023.
[23] C.-H. Yang, J.-P. Wu, F.-Y. Lee, T.-Y. Lin, and M.-H. Tsai, "Detection and mitigation of syn flooding attacks through syn/ack packets and black/white lists," Sensors, vol. 23, p. 3817, 2023.
[24] P. Shalini, V. Radha, and S. G. Sanjeevi, "Early detection and mitigation of TCP SYN flood attacks in SDN using chi-square test," Journal of Supercomputing, vol. 79, 2023.
[25] M. Cherian and S. L. Varma, "Secure SDN–IoT framework for DDoS attack detection using deep learning and counter based approach," Journal of Network and Systems Management, vol. 31, p. 54, 2023.
[26] J. Ma, W. Su, Y. Li, Y. Yuan, and Z. Zhang, "Synchronizing real-time and high-precision LDoS defense of learning model-based in AIoT with programmable data plane, SDN," Journal of Network and Computer Applications, vol. 229, p. 103916, 2024.
[27] R. F. Kapourchali, R. Mohammadi, and M. Nassiri, "P4httpGuard: detection and prevention of slow-rate DDoS attacks using machine learning techniques in P4 switch," Cluster Computing, vol. 27, pp. 8047-8064, 2024.
[28] R. Mall, A. Kumar, K. Abhishek, and A. Kumar, "Enhancing Security in Software-Defined Networks: Hybrid Deep Learning Models for Flooding Attack Detection," in 2025 3rd International Conference on Disruptive Technologies (ICDT), 2025, pp. 1118-1123.
[29] M. Sinha, P. Bera, M. Satpathy, K. S. Sahoo, and J. J. Rodrigues, "DDoSBlocker: Enhancing SDN security with time-based address mapping and AI-driven approach," Computer Networks, vol. 259, p. 111078, 2025.
[30] S. Kumar and S. Gupta, "SDN TCP-SYN Dataset: A dataset for TCP-SYN flood DDoS attack detection in software-defined networks," Data in Brief, vol. 59, p. 111314, 2025.