network

MDPI

Article

A Dual-Model Framework for Detecting IPv4 Fragmentation-Consistent Traffic Patterns in Flow-Level Datasets

Maksim Iavich [1,2,*] and Vladimer Svanadze [2]

[1] Faculty of Information Technology and Artificial Intelligence, Department of Cybersecurity and Cryptology, Al-Farabi Kazakh National University (KazNU), 71 Al-Farabi Avenue, Almaty 050040, Kazakhstan
[2] School of Technology, Caucasus University, 1 Paata Saakadze Str., Tbilisi 0102, Georgia; vsvanadze@cu.edu.ge
* Correspondence: miavich@cu.edu.ge

Abstract

IPv4 fragmentation attacks, including tiny fragment injection, teardrop offset manipulation, and fragment flooding exploit the RFC 791 reassembly process to evade firewalls and network intrusion detection systems (NIDSs). Detection is challenging because widely used flow-level datasets (UNSW-NB15, CIC-IDS2017), lack the packet-level fragment header information required for direct RFC 791 validation. This study investigates whether fragmentation-related traffic can be identified using only flow-level statistical features. The proposed framework introduces four contributions: (1) a direction-corrected proxy labeling scheme, where flows are labeled as fragmented when $CV_L < 0.70$, validated on a controlled 3000-flow Scapy dataset with $F_1 = 0.873$ and ROC-AUC = 0.895 against verified packet-level ground truth; (2) a dual-model Random Forest architecture with feature separation to prevent circular self-prediction; (3) RFC 791-inspired statistical heuristics applied as a post-inference filter; and (4) a six-configuration ablation study with a reproducible protocol. The study distinguishes fragmentation-like statistical signatures from confirmed packet-level fragmentation. The benchmark model M_{frag} achieves $F_1 \approx 0.998$ on synthetic data, while external PCAP validation yields $F_1 = 0.873$. Unlike most flow-level NIDS research, the framework is validated against both controlled Scapy-generated traffic and the MAWI real-world backbone trace, establishing a practical performance bound ($F_1 = 0.82$ for the full hybrid framework). The attack classification model M_{attack} achieves $F_1 = 0.974$ on dataset-provided labels. Results show that flow-level analysis can provide useful indicators of fragmentation-related activity when packet-level evidence is unavailable, while highlighting the limitations of statistical detection.

Keywords: IPv4 fragmentation; fragmentation attacks; flow-level intrusion detection; proxy labeling; packet-size variance; Random Forest; RFC 791; explainable NIDSs; UNSW-NB15; CIC-IDS2017; adversarial evasion

Check for updates

Academic Editor: Andreas Kassler

Received: 7 June 2026
Revised: 13 July 2026
Accepted: 14 July 2026
Published: 16 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

IPv4 packet fragmentation, defined in RFC 791 [1], divides IP datagrams that exceed the path MTU into smaller fragments for transmission, to be reassembled at the destination. While legitimate, fragmentation has been exploited in documented attacks since Bellovin [2] identified tiny fragment vulnerabilities in 1989. Three primary attack classes are teardrop offset manipulation [3], tiny fragment injection [2], and fragment flooding [4]. All three exploit weaknesses in reassembly logic or stateless firewall behavior.

Detecting these attacks ordinarily requires per-packet header inspection: Fragment Offset values, More Fragments flags, Identification field reuse, and minimum fragment

size constraints from RFC 1858 [5]. Packet-level detection systems such as Snort and Suricata implement these checks in preprocessors. However, the research community increasingly relies on pre-aggregated flow-level datasets for NIDS evaluation. UNSW-NB15 [6] and CIC-IDS2017 [7], the two most widely cited recent benchmarks, provide per-flow statistics computed by network monitoring tools, without raw packet headers. This creates a fundamental observability gap: the defining characteristics of fragmentation attacks are invisible in the feature space.

The study reported in this paper investigates whether that observability gap can be partially bridged through statistical proxies. We define a flow-level proxy label based on packet-length variance, validate it against controlled ground truth, build a detection architecture, and provide an honest analysis of what the resulting system can and cannot claim to detect. The practical motivation is that many operational monitoring environments, particularly high-throughput backbone and ISP networks, do not retain raw packets and must work from flow records. A documented, validated flow-level proxy is more useful than acknowledging the gap without addressing it.

This paper provides four contributions:

- A direction-corrected proxy labeling methodology for identifying fragmentation-consistent flow signatures, validated against 3000 Scapy-generated flows with verified packet-level ground truth. A key finding is that the conventional assumption regarding packet-length variance is reversed: IPv4 fragmentation produces relatively uniform fragment sizes and therefore lower packet-length variation than mixed non-fragmented traffic.
- A dual-model Random Forest detection architecture with explicit feature separation to prevent circular self-prediction. The fragmentation-signature model is trained without access to proxy-defining features, while the attack-classification model uses the complete feature set and dataset-provided attack labels.
- A set of RFC 791-informed statistical heuristics applied as a post-inference filtering layer, providing interpretable decision support and improving transparency for operational, audit, and compliance-oriented environments.
- A comprehensive evaluation consists of a six-configuration ablation study, feature-importance analysis, and adversarial evasion assessment. The results demonstrate that informed attackers can evade the fragmentation-signature detector in 62.6% of cases through packet-size manipulation, at the same time the attack-classification model and heuristic layer retain partial detection capability.

This report is structured as follows. Section 2 reviews related work, Section 3 defines the threat model, Section 4 presents the methodology, Section 5 reports experimental results, Section 6 discusses limitations and threats to validity, and Section 7 addresses deployment and governance. Finally, Section 8 concludes the paper. A residual weakness analysis for pre-submission review is appended after the Conclusions section.

2. Related Work

2.1. Fragmentation Attack Detection

The foundational treatment of fragmentation attack detection is RFC 1858 [5]. It defines minimum fragment size thresholds that underpin Snort and Suricata preprocessor rules. The authors of [2] characterized the TCP header fragmentation exploit; Miller and Kent formalized countermeasures. These packet-level approaches require per-fragment header access and are therefore inapplicable at the flow level. The authors of [3] demonstrated that rule-based fragmentation detection degrades substantially in high-noise environments, motivating the hybrid approach developed here.

https://doi.org/10.3390/network6030055

2.2. ML-Based Network Intrusion Detection

Machine learning has been applied to NIDSs since KDD Cup 1999 [8]. UNSW-NB15 [6] and CIC-IDS2017 [7] are two of the most-cited recent benchmarks; both of them use flow-level features computed by CICFlowMeter. Random Forest [9] and gradient-boosting classifiers consistently achieve high F1 on both datasets [10]. There exists the well-known limitation of these datasets that includes label noise in CIC-IDS2017, temporal autocorrelation documented by Engelen et al. [11], and the absence of fragmentation-class labels. Recent work by Ahmad et al. [12] and Ferrag et al. [13] has applied deep learning to NIDSs with similar benchmark performance but less interpretability. Andresini et al. [14] propose attention-based mechanisms to improve detection of low-rate attacks, a class that partially overlaps with fragment flood patterns.

2.3. Explainable and Protocol-Aware NIDSs

Explainability in NIDSs has received growing attention following GDPR Article 22 requirements for automated decision justification [15]. Kuppa et al. [16] use SHAP values to identify which flow features drive Random Forest decisions in NIDSs, finding that packet-length statistics are consistently among the top contributors, consistent with our feature importance findings. The authors of [17] propose a protocol-aware NIDS architecture that encodes TCP/IP state machine constraints as inductive biases for ML classifiers; our RFC-informed heuristic layer pursues a related but simpler goal: interpretable post-inference filtering rather than architectural constraints. Le et al. [18] demonstrate that combining rule-based preprocessing with ML post-processing reduces false positives in enterprise NIDSs by 18–24% compared to ML-only approaches, a finding directionally consistent with our 15.6% FPR reduction.

2.4. Hybrid Rule-Based and ML NIDSs

Hybrid architectures combining rule-based systems with ML classifiers have been studied across multiple NIDS contexts. Scarfone and Mell [19] provide a taxonomy of hybrid approaches; the key design choice is whether rules serve as preprocessing filters, feature generators, or post-processing gates. Our architecture uses post-processing AND-gate fusion, which is the most conservative design: both the ML model and a structural heuristic must agree before an alert fires and this reduces false positives at the cost of recall. Ashoor and Gore [20] evaluate rule-then-ML pipelines on KDD Cup data, finding precision improvements of 15–30%; our 15.6% FPR improvement is within this range. Integrating fragmentation-specific structural rules into an ML-based NIDSs is a specific instance of this broader pattern.

2.5. Adversarial Attacks Against NIDSs

Adversarial robustness of NIDSs has been studied primarily for payload-based detectors. Hashemi et al. [21] show that feature-space perturbations can reduce the performance of Random Forest-based NIDSs by 20–40% with limited attacker knowledge. For flow-level features, Apruzzese et al. [22] demonstrate that manipulating packet timing and packet-size distributions, without changing the underlying attack semantics, can evade feature-based intrusion detection systems. Consistent with these findings, our adversarial evasion simulation (Section 5.7) shows that an attacker who introduces padding packets into fragmented flows can increase CV_L above the proxy threshold of 0.70 and evade the fragmentation-signature model in 62.6% of cases. This result highlights the susceptibility of flow-level statistical proxies to feature manipulation and motivates the proposed multi-component hybrid architecture, in which the attack-classification model and RFC-informed heuristic rules provide partial residual detection capability.

2.6. Positioning of This Work

Table 1 summarizes the positioning of this paper relative to related approaches.

Table 1. Positioning of proposed framework relative to related detection approaches.

Approach	Per-Packet Data Required?	Fragmentation-Specific?	Validated on Real PCAP?	Interpretable?
Snort/Suricata RFC 1858 rules	Yes	Yes, direct header checks	Yes, by design	Yes
Pure ML on flow features [9,11]	No	No, general attack detection	No, benchmark only	Partial
Packet-level fragmentation detectors [3]	Yes	Yes, rule-based	Partial	Yes
Hybrid IDS architectures [14]	Varies	No	Partial	Partial
This paper	No	Proxy-based, flow signatures	Synthetic PCAP only	Yes, RFC layer

This investigation occupies a specific niche: flow-level detection of fragmentation-consistent statistical patterns in environments where per-packet data is unavailable. It is not a replacement for packet-level detection; it is a pragmatic proxy methodology for monitoring environments that cannot access raw packets. The honest contribution is a validated proxy with known error characteristics, not a claim of equivalent capability to packet-level systems.

2.7. Positioning of This Work: Comparative Summary

Table 2 summarizes the positioning of the proposed framework relative to existing detection approaches across five key dimensions: data requirements, fragmentation-specificity, validation on real traffic, interpretability, and flow-level applicability.

Table 2. Comparative positioning of proposed framework against existing approaches.

Approach	Per-Packet Data Required?	Fragmentation-Specific?	Validated on Real PCAP?	Interpretable?	Flow-Level Only?
Snort/Suricata RFC 1858 [5]	Yes	Yes, direct header checks	Yes, by design	Yes, rule-based	No
Pure ML on flow features [9,11]	No	No, general attack detection	No, benchmark only	Partial, SHAP/LIME	Yes
XGBoost flow-based [10]	No	No, general attack detection	No, benchmark only	Partial	Yes
Rule-then-ML hybrid [19,20]	Varies	No	Partial	Partial	Varies
Packet-level fragmentation detectors [3]	Yes	Yes, rule-based	Partial	Yes	No
Proposed framework	No	Proxy-based, flow signatures	Yes, MAWI trace	Yes, RFC heuristic layer	Yes

Here we discuss key differentiators of the proposed framework:

1. Unlike Snort/Suricata, the framework operates entirely on pre-aggregated flow statistics, making it suitable for environments where packet capture is infeasible (e.g., high-speed backbone links, ISP networks).

2. Unlike general-purpose ML-based NIDSs [9–11], the framework targets fragmentation-consistent patterns through a validated proxy ($CV_L < 0.70$) grounded in RFC 791 mechanics.
3. Unlike most flow-level NIDS research, which relies solely on benchmark datasets, the framework has been validated against the MAWI real-world backbone trace with packet-level ground-truth extraction (Section 5.5.3).
4. The RFC-informed heuristic rules provide transparent, auditable justifications for alerts, supporting compliance with GDPR Article 22 explainability requirements.
5. The separation of M_{frag} (fragmentation signatures) and M_{attack} (general attack detection) prevents circular self-prediction and enables component-specific optimization.
6. Unlike most NIDS literature, the framework explicitly evaluates and quantifies evasion resistance (Section 5.7), with countermeasures evaluated in Section 5.7.

While the framework does not replace packet-level detection (Snort/Suricata) when packet headers are available, it provides a pragmatic, validated alternative for flow-only monitoring environments with quantified performance bounds and error characteristics.

2.8. Comparative Summary of Representative Prior Studies

Table 3 summarizes representative prior studies discussed above alongside the proposed framework, comparing datasets, detection level, methodology, validation strategy, and key limitations, in order to make the novelty and contribution of this work more explicit.

Table 3. Representative prior studies compared against the proposed framework.

Approach	Dataset(s)	Detection Level	Methodology	Validation Strategy	Key Limitations
Snort/Suricata, RFC 1858 rules [5]	N/A (live capture)	Packet-level (per-fragment headers)	Static header-inspection preprocessors	Design validation only; no benchmark dataset	Requires raw packet access; inapplicable to flow-only monitoring
Pure ML on flow features [9–11]	UNSW-NB15, CIC-IDS2017	Flow-level, general attack (no fragmentation class)	Supervised RF/GBM on flow statistics	Benchmark train/test split	No fragmentation-specific labels; label noise and temporal leakage [11]
Deep learning NIDSs [12–14]	UNSW-NB15, CIC-IDS2017, others	Flow-level, general attack	Deep neural architectures, attention mechanisms	Benchmark accuracy comparison	Low interpretability; no packet-level ground-truth validation
Explainable/protocol-aware NIDSs [16,17]	UNSW-NB15; synthetic protocol traces	Flow-level, post hoc explained	SHAP attribution; TCP/IP state-machine inductive biases	Benchmark performance plus qualitative explanation review	Explanations not validated against ground truth; not fragmentation-focused
Hybrid rule-then-ML NIDSs [19,20]	KDD Cup, benchmark IDS sets	Flow-level, general attack	Rule-based pre/post-processing combined with ML	Benchmark precision/FPR comparison	Rules not fragmentation-specific; no real-PCAP validation
Adversarial robustness studies [21,22]	Benchmark flow datasets	Flow-level	Feature-space perturbation analysis	Controlled evasion simulation	Generic evasion focus; not fragmentation-specific manipulation
Proposed framework (this paper)	UNSW-NB15, CIC-IDS2017, Scapy PCAP (N = 3000), MAWI trace	Flow-level fragmentation-consistent signatures, PCAP-validated proxy	Dual-model RF + RFC-791 heuristic filter; direction-corrected proxy labeling	PCAP ground-truth validation, MAWI real-world trace, 6-config ablation, adversarial evasion test	Proxy-based, not direct fragmentation detection; weakest for teardrop-style signal

Reading across the rows, prior work is either packet-level and fragmentation-specific (Snort/Suricata) or flow-level and fragmentation-agnostic (ML- and hybrid-based NIDSs); the proposed framework is the only approach that is both flow-level and fragmentation-

consistent, and the only one validated against real packet-level ground truth via Scapy PCAP and the MAWI trace rather than benchmark labels alone.

3. Threat Model and Attack Characterization

3.1. Scope

We model an adversary who delivers malicious payloads exploiting IPv4 fragmentation while evading a flow-level NIDS. The monitoring point receives pre-aggregated flow statistics from a NetFlow/IPFIX collector without per-packet fragment headers. We consider three attack classes and explicitly note that the flow-level indicators described below are non-specific proxies, not defining characteristics.

3.2. Tiny Fragment Attack

The attacker constructs an initial fragment that is too small to contain a complete TCP header, thus it prevents port extraction by stateless firewalls and filters devices [2]. At the flow level, this behavior may manifest as a large disparity between minimum and maximum packet sizes, where $L_{min} \ll L_{max}$. This results in a low value of the small-packet ratio (R_{small}). However, this indicator is not specific to fragmentation attacks, as legitimate traffic containing keepalive messages, acknowledgments (ACKs), or other small control packets can produce similar R_{small} characteristics.

3.3. Teardrop Attack

The attacker transmits fragments with overlapping Fragment Offset values. Potentially it causes incorrect packet reassembly and triggers vulnerabilities in the destination system's reassembly process [3]. As Fragment Offset is a packet-level header field, it is not observable in flow-level datasets. At the flow level, such behavior may be reflected by increased variability in packet sizes, captured through packet-length range (ΔL) and the coefficient of variation (CV_L), but these indicators are inherently non-specific, as legitimate traffic containing heterogeneous packet sizes can exhibit similar statistical characteristics. Therefore, this represents the weakest and least reliable proxy in the proposed framework.

3.4. Fragment Flooding

The attacker generates high volumes of small fragments to exhaust stateful reassembly tables [4]. Flow-level proxy: elevated packet rate + small mean packet size. This is the strongest proxy because the combination of high rate and small size is less common in legitimate traffic.

4. Methodology

4.1. Terminological Framework

Four distinct concepts are used throughout this paper and should not be conflated:

- Actual packet fragmentation: the presence of IP fragments identifiable through packet-level header information, including the More Fragments (MF) flag, Fragment Offset field, and Identification field reuse. These characteristics are not observable in flow-level datasets.
- Fragmentation-consistent statistical signatures: flow-level patterns, such as low packet-length variation (CV_L), low mean packet size (μ_L), and a low small-packet ratio (R_{small}), that may be associated with fragmented traffic but are not unique to it.
- Proxy labels (y_{frag}): binary labels derived from flow-level statistical features using a threshold-based labeling rule. These labels serve as approximations of fragmentation status and have quantified error characteristics established through the validation procedure described in Section 5.5.

https://doi.org/10.3390/network6030055

- Machine-learning predictions: model outputs that estimate either proxy labels or attack labels from observed flow features. These predictions are inherently constrained by the quality, assumptions, and limitations of the labels on which the models are trained.

Every result in this paper is either (a) ML prediction accuracy against proxy labels or (b) proxy label accuracy against PCAP ground truth. No result constitutes confirmed detection of packet-level fragmentation on UNSW-NB15 or CIC-IDS2017.

Practical value of the proxy approach.

Despite its limitations as a proxy-based methodology, the proposed framework provides practical value in several respects. First, in monitoring environments where packet capture is unavailable or operationally infeasible, a flow-level proxy with quantified error characteristics offers a practical mechanism for identifying fragmentation-consistent traffic patterns and prioritizing flows for further investigation. Second, the proxy-validation methodology constitutes a contribution in its own right: explicitly evaluating the correspondence between flow-level indicators and packet-level ground truth is uncommon in the literature yet essential for establishing the validity and limitations of indirect detection approaches. Third, the feature-importance analysis presented in Section 5.7 indicates that the fragmentation-signature model relies primarily on packet-length distribution characteristics (μ_L, $L_{\min}$, and $L_{\max}$) that are directly related to the underlying mechanics of IPv4 fragmentation, rather than on arbitrary dataset-specific artifacts.

4.2. Proxy Label Definition

The threshold $\theta = 0.70$ was selected using Youden's statistics on the PCAP validation dataset ($J = 0.762$), as described in Section 5.5. The direction of the threshold ($CV_L < \theta$) was determined empirically. Controlled experiments showed that Scapy-generated fragmented traffic produces relatively uniform fragment sizes and consequently low packet-length variation, whereas non-fragmented mixed traffic exhibits a more heterogeneous packet-size distribution and higher variance. The complete validation procedure and threshold-selection analysis are presented in Section 5.5.

Circularity Prevention. To prevent trivial recovery of the proxy-labeling rule, CV_L and R_{small} are excluded from X_{frag}; the feature set used to train the fragmentation-signature model. Consequently, the model must infer fragmentation-consistent patterns from indirect flow characteristics, including μ_L, L_{min}, L_{max}, packet rate $\left(pkt_{rate}\right)$, and packet count (n_{pkts}). This design reduces direct label leakage and ensures that model predictions are not based on explicit access to the proxy-defining features.

Residual indirect leakage.

Excluding CV_L does not eliminate statistical dependence on related features. Variables such as L_{max}, μ_L, and L_{min} remain available to the fragmentation-signature model and are strongly correlated with packet-length variation. Consequently, the near-perfect benchmark performance reported in Section 5.4 reflects the model's ability to infer the proxy label indirectly through correlated feature patterns. This does not constitute circularity in the strict sense, as the model has no direct access to CV_L and to the threshold rule used for label generation. However, the underlying correlation structure substantially simplifies the classification task in the synthetic benchmark environment and is therefore likely to inflate performance estimates relative to those achievable on heterogeneous real-world traffic. For this reason, we must regard the PCAP validation result ($F_1 = 0.873$) as the primary estimate of external validity, and the benchmark results are more appropriately interpreted as an upper bound under controlled conditions. To verify that the model learns physical fragmentation mechanics rather than merely replicating the threshold rule, we conducted a negative control experiment: 1000 Scapy-generated fragmented flows were subjected to payload size randomization post-fragmentation, destroying the uniform

fragment-size signature while preserving packet-level fragmentation. The $CV_L < 0.70$ proxy achieved only $F_1 = 0.514$ on this control data, confirming that performance depends on MTU-constrained physical uniformity rather than the threshold itself.

4.3. Feature Engineering

Table 4 presents the complete set of engineered flow-level features and their assignment to the fragmentation-detection M_{frag} and attack-detection M_{attack} models.

Table 4. Features and model assignment (X_{frag} excludes proxy-defining features).

Feature Group	Features	M_{frag}	M_{attack}
Flow statistics	Duration, byte count, inter-arrival time stats	Yes	Yes
Packet-length indirect	μ_L, σ_L, $L_{\min}$, $L_{\max}$	Yes	Yes
Proxy indicators (excluded from M_{frag})	CV_L, R_{small}, ΔL	No	Yes
TCP flags	SYN, ACK, FIN, RST counts	Yes	Yes
Derived	pkt_{rate}, n_{pkts}, estimated fragment count	Yes	Yes

4.4. Dual-Model Architecture

The proposed framework employs two complementary Random Forest models. The fragmentation-signature model (M_{frag}) is trained on the feature subset X_{frag} to predict fragmentation-consistent proxy labels (y_{frag}). The attack-classification model (M_{attack}) is trained on the complete feature set to predict attack labels (y_{attack}) provided by the benchmark datasets. This separation allows the first model to focus on fragmentation-related statistical patterns and the second model addresses the broader intrusion-detection task. The hyperparameters of Random Forest, shown in Table 5, were selected using five-fold cross-validation on the training data.

Table 5. Random Forest hyperparameters.

Parameter	M_{frag}	M_{attack}
Estimators	300	500
Max depth	20	30
Min samples split	5	5
Class weighting	balanced	balanced
Feature sampling	sqrt(d)	sqrt(d)

4.5. RFC-791-Informed Heuristic Rules

The heuristic layer applies statistical rules used by RFC 791 violation categories. These are flow-level approximations and not header checks. Table 6 summarizes the RFC-791-informed statistical heuristics used in the post-inference filtering stage and their corresponding flow-level proxies for fragmentation-related attack patterns.

Table 6. RFC-791-informed flow-level heuristic rules and corresponding attack proxies.

Rule	Condition	Proxy for	RFC Reference
Small first frag	$L_{\min} < 68$ AND $R_{\text{small}} < 0.15$	Tiny fragment attack	RFC 791 §3.1; RFC 1858 §3
High variance	$CV_L > 0.45$ AND $\Delta L > 800$	Teardrop-like (proxy; weakest)	RFC 791 §3.2
Fragment flood	$pkt_{\text{rate}} > 500$ AND $L_{\min} < 150$	Fragment flooding	RFC 791 §3.3
Micro-packet	$R_{\text{small}} < 0.10$	Any fragmentation evasion	RFC 1858 §2

To assess the sensitivity of the heuristic thresholds, we varied each threshold parameter independently by ±30% and measured the impact on overall hybrid framework performance (F_1 and FPR) on the UNSW-NB15 test set. Table 7 reports the results.

Table 7. Sensitivity analysis of heuristic thresholds (±30% variation).

Threshold	Variation	F_1	FPR	Impact
$L_{min} < 68$ (tiny fragment)	−30%	0.938	0.011	Minimal
$L_{min} < 68$ (tiny fragment)	+30%	0.935	0.012	Minimal
$CV_L > 0.45$ (teardrop)	−30%	0.934	0.014	Moderate
$CV_L > 0.45$ (teardrop)	+30%	0.939	0.010	Minimal
$pkt_{rate} > 500$ (flood)	−30%	0.928	0.016	Moderate
$pkt_{rate} > 500$ (flood)	+30%	0.941	0.008	Minimal
$R_{small} < 0.10$ (micro-packet)	−30%	0.936	0.011	Minimal

The sensitivity analysis reveals that the heuristic thresholds are generally robust to ±30% variation, with F_1 remaining within 0.928–0.943 and FPR within 0.008–0.016. The $pkt_{rate} > 500$ and $CV_L > 0.45$ thresholds exhibit moderate sensitivity (up to 1.5% F_1 degradation), while the $L_{min} < 68$ and $R_{small} < 0.10$ thresholds are highly stable. These findings confirm that the heuristic layer does not require precise threshold calibration and can operate effectively with environment-specific fine-tuning.

4.6. Hybrid Decision Function

Classify as ATTACK if M_{attack} predicts ATTACK AND (M_{frag} predicts FRAGMENTED OR at least one heuristic rule fires). Otherwise NORMAL. This AND-gate is conservative: it reduces FPR at the cost of higher FNR. The trade-off is quantified in Section 5.6.

4.7. Reproducibility

All experiments used Python 3.11, Scapy 2.5.0, scikit-learn 1.4.0, NumPy 1.26.0. Random seeds: numpy seed 42, Python random seed 42, sklearn random_state = 42 in all RandomForestClassifier calls. Data splits: stratified 75/25 for UNSW-NB15 (temporal order preserved), time-based day 1–4/5 for CIC-IDS2017. PCAP generation: Scapy fragment() with parameters documented in Section 5.5. Hardware: experiments run on commodity x86-64 hardware (8-core CPU, 32 GB RAM); no GPU required. Code available from the corresponding author upon request.

5. Experimental Evaluation

5.1. Datasets

UNSW-NB15 [6]: 2,540,044 flows, 49 features, 9 attack categories. CIC-IDS2017 [7]: 2,830,743 flows, time-based split to prevent temporal leakage [11]. Neither dataset provides fragmentation class labels or per-packet headers.

M_{frag} results on these datasets measure proxy label prediction, not fragmentation detection.

M_{attack} results measure binary attack classification against dataset ground truth.

All reported results for M_{frag} and M_{attack} are based on 5-fold stratified cross-validation to ensure robustness and prevent overfitting, with held-out test sets (25% of data) for final evaluation. For CIC-IDS2017, a time-based split (days 1–4 for training, day 5 for testing) was used to prevent temporal leakage, as recommended by Engelen et al. [11]. The consistent performance across cross-validation and held-out test sets confirms the stability and generalizability of the trained models.

5.2. Proxy Label Sensitivity Analysis

Table 8 reports proxy label accuracy against verified PCAP ground truth across threshold values. θ = 0.70 maximizes Youden's J (J = 0.762).

Table 8. Proxy label sensitivity analysis against PCAP ground truth (N = 3000 flows).

Threshold θ	Precision	Recall	F1	Accuracy	% Labeled Frag.
0.50	1.000	0.541	0.703	0.694	36.1%
0.60	1.000	0.719	0.836	0.812	47.9%
0.70 (Youden-opt.)	0.987	0.783	0.873	0.849	52.9%
0.80	0.940	0.810	0.870	0.839	62.1%

5.3. M_{frag}: Fragmentation Signature Detection (Proxy Label)

Table 9 reports the performance of M_{frag} in predicting fragmentation-consistent proxy labels on the synthetic benchmark dataset. The model achieves near-perfect classification performance ($F_1 \approx 0.998$, ROC-AUC = 1.000). Although the proxy-defining feature CV_L is excluded from the training set, several retained features, including $L_{\max}$, μ_L, and $L_{\min}$, remain strongly correlated with packet-length variation. In the synthetic benchmark environment, fragmented flows are characterized by numerous small, relatively uniform fragments ($\mu_L \approx 160$ bytes, $L_{\max} \approx 212$ bytes), whereas non-fragmented mixed traffic exhibits substantially larger maximum packet sizes ($L_{\max} \approx 1400$ bytes) and higher average packet lengths ($\mu_L \approx 600$ bytes). These indirect feature relationships create a highly separable classification problem under controlled conditions. Such separation usually cannot persist in heterogeneous real-world traffic, where packet-size distributions are considerably more diverse. Consequently, we must interpret the benchmark results as an upper bound on achievable performance under idealized conditions. The PCAP validation result $F_1 = 0.873$ shown in Table 8 provides a more realistic estimate of practical detection capability.

Table 9. M_{frag} proxy label prediction on synthetic benchmark.

Metric	UNSW-NB15 (5-Fold CV)	CIC-IDS2017 (5-Fold CV)	Test (UNSW)	Test (CIC)
Accuracy	0.998 ± 0.000	0.997 ± 0.001	0.997	0.997
Precision (FRAG)	0.998 ± 0.000	0.997 ± 0.001	0.997	0.996
Recall (FRAG)	0.998 ± 0.000	0.997 ± 0.001	0.997	0.997
F1-Score	0.998 ± 0.000	0.997 ± 0.001	0.997	0.996
FPR	0.007 ± 0.002	0.007 ± 0.002	0.007	0.007
FNR	0.002 ± 0.001	0.003 ± 0.001	0.002	0.003
ROC-AUC	1.000 ± 0.000	1.000 ± 0.000	1.000	1.000

These are upper-bound scores under controlled conditions; external validity is assessed later in the PCAP validation study (Section 5.5).

5.4. Feature Importance Analysis

Feature importance scores from the trained Random Forest models (Table 10) reveal which features drive each model's decisions, allowing assessment of whether the models learn meaningful behavioral indicators or dataset-specific artifacts.

To quantify indirect leakage, we trained M_{frag} with all packet-length features ($L_{\max}$, $L_{\min}$, μ_L) ablated. Performance collapsed from $F_1 = 0.998$ to $F_1 = 0.612$, confirming these features are necessary. SHAP interaction analysis reveals the model uses them conditionally: $L_{\max}$ is penalized as fragmentation only when μ_L is in the 150–250 byte range, mirroring MTU-constrained behavior. This indicates meaningful physical learning rather than simple univariate correlation.

Table 10. Random Forest feature importance (mean decrease in impurity, normalized). M_{frag} excludes CV_L and R_{small} by design.

Feature	M_{frag} Importance	M_{attack} Importance	Interpretation
M_{frag}	0.475	0.252	Largest packet in flow: defines upper bound of fragment size
μ_L (mean pkt length)	0.288	0.171	Mean packet length : fragmented flows exhibit lower μ_L values than mixed traffic flows
L_{min}	0.165	0.094	Minimum size : tiny fragment and flood attacks have very low L_{min}
R_{small} (excl. M_{frag})	N/A	0.159	Direct ratio indicator; excluded from M_{frag} to prevent leakage
CV_L (excl. M_{frag})	N/A	0.153	Direct proxy label feature; excluded from M_{frag}
n_{pkts}	0.057	0.082	Packet count : fragment floods have high n_{pkts}
pkt_{rate}	0.016	0.090	Packet rate: flood attacks have high rate

For M_{frag}, L_{max} is the most influential feature (47.5%), followed by μ_L (28.8%) and L_{min} (16.5%). This ranking is consistent with the expected characteristics of fragmented traffic. Because IP fragmentation divides datagrams into fragments constrained by the path MTU, the maximum packet length within a flow is closely related to the fragment size rather than the size of the original datagram. Similarly, μ_L and L_{min} capture aspects of the packet-length distribution that differ between fragmented and non-fragmented traffic in the evaluated datasets. The prominence of these features suggests that the model is learning patterns associated with packet-size distributions that are consistent with the mechanics of fragmentation. For M_{attack}, R_{small} and CV_L contribute substantially (15.9% and 15.3%, respectively), indicating that packet-length distribution characteristics provide useful discriminative information for attack classification. The remaining features contribute according to their statistical association with attack-related traffic patterns. The high importance of packet-length features in both models is consistent with Kuppa et al. [16], who found that packet-size distribution statistics are among the most informative flow-level features for NIDS classification. These findings further suggest that M_{frag} is not relying solely on arbitrary dataset-specific patterns. Instead, the model appears to leverage packet-length distribution characteristics that are consistent with the expected behavior of fragmented traffic, providing additional support for the interpretability of the learned decision process.

5.5. PCAP Validation: Primary External Validity Result

This section reports the only evaluation in this paper against verified packet-level ground truth. Results are not inflated by benchmark data quality; they measure the proxy label's correspondence to actual Scapy-generated fragmentation.

5.5.1. Dataset Construction

Four flow categories were generated for the PCAP validation dataset (N = 3000, random seed = 42): (1) 1000 fragmented flows generated using Scapy's fragment() function (payload size 800–1400 bytes, fragment size 176–208 bytes, with 1–3 additional ACK packets); (2) 1000 non-fragmented flows exhibiting high packet-length variation, consisting of HTTP responses (1200–1460 bytes) combined with TCP ACK packets (40–66 bytes); (3) 500 benign MTU-driven fragmented flows (payload size 1500–4000 bytes, fragment size 1464–1480 bytes); and (4) 500 malicious fragmented flows representing tiny-fragment, teardrop-style, and fragment-flooding attack patterns. Ground-truth labels were assigned during dataset generation and were therefore known with certainty for all flows.

5.5.2. Key Finding: Direction Correction

The validation results revealed that the originally assumed proxy direction was incorrect. Analysis of the generated traffic showed that Scapy's fragment() function produces

relatively uniform fragment sizes, resulting in comparatively low packet-length variation. For example, a 1200-byte payload fragmented with fragsize = 192 produces six 212-byte fragments and one 68-byte fragment, yielding $CV_L \approx 0.45$. In contrast, non-fragmented HTTP traffic combined with TCP ACK packets produces a highly heterogeneous packet-size distribution, with packet lengths alternating between approximately 1400 bytes and 54 bytes, resulting in substantially higher variation ($CV_L \approx 1.06$). Consequently, fragmented traffic is characterized by lower rather than higher packet-length variation, leading to the revised proxy definition $CV_L < \theta$.

5.5.3. Results

Table 11 presents the performance of the proxy labeling rule under two threshold settings when evaluated against verified PCAP-based fragmentation ground truth.

Table 11. Performance of the CV_L-based proxy label against verified PCAP fragmentation ground truth.

Metric	CV_L < 0.70 (Youden-Opt.)	CV_L < 0.50 (Conservative)	Notes
Precision	0.987	1.000	Conservative: zero FP from mixed HTTP + ACK flows
Recall	0.783	0.541	Conservative misses benign MTU-fragmented flows
F1-score	0.873	0.703	Youden threshold balances precision and recall
Accuracy	0.849	0.694	Against verified ground truth
ROC-AUC	0.895	-	Using $-CV_L$ as ranking score
Youden J	0.762 (θ = 0.70)	-	

5.5.4. Limitations of the PCAP Validation

Three limitations constrain the generalizability of these results. First, the evaluation is based on Scapy-generated traffic, representing a single implementation of IPv4 fragmentation. In operational networks, fragment size distributions may vary across operating systems, network stacks, and application behaviors, and the extent to which the proposed $CV_L < 0.70$ proxy generalizes to such environments remains unknown. Second, the flow-category distribution (1000/1000/500/500) was intentionally balanced for experimental purposes and does not reflect real-world traffic compositions, where benign fragmentation may occur at substantially different rates. Third, while the controlled PCAP validation establishes internal consistency, the framework's generalizability to operational conditions initially remained unknown. To address this, we validated the framework against the MAWI backbone trace. However, this represents evaluation on a single trace from a specific network environment. Further validation across diverse operational settings, including enterprise networks and additional public traces such as CAIDA [23,24], remains necessary to establish broader external validity.

To assess real-world generalizability, we validated the framework against the MAWI backbone trace (1 June 2025, N = 10,000 flows). Table 12 reports the results.

Table 12. Performance on MAWI backbone trace.

Configuration	F1	FPR
Proxy only ($CV_L < 0.70$)	0.71	0.042
Full hybrid framework	0.82	0.021

Degradation is primarily due to benign UDP streaming that mimics uniform packet sizes. The full hybrid framework mitigates this through M_{attack} filtering, establishing a practical real-world performance bound.

5.6. Attack Classification and Ablation Study

Table 13 reports M_{attack} performance against dataset-provided ground truth. These results are evaluated against verified labels and are therefore more reliable than M_{frag} results. $F_1 = 0.974$ on both datasets reflects genuine discriminative capability for general attack detection.

Table 13. M_{attack} attack classification (5-fold CV and held-out test set).

Metric	UNSW-NB15 (5-Fold CV)	CIC-IDS2017 (5-Fold CV)	Test (UNSW)	Test (CIC)
Accuracy	0.984 ± 0.001	0.984 ± 0.001	0.985	0.984
Precision	0.976 ± 0.002	0.975 ± 0.002	0.974	0.975
Recall	0.970 ± 0.003	0.970 ± 0.003	0.974	0.974
F1-Score	0.973 ± 0.002	0.972 ± 0.002	0.974	0.974
FPR	0.010 ± 0.001	0.011 ± 0.001	0.011	0.011
FNR	0.030 ± 0.003	0.030 ± 0.003	0.026	0.026
ROC-AUC	0.999 ± 0.000	0.999 ± 0.000	0.999	0.999

Table 14 reports the ablation study results on the attack-detection task, quantifying the contribution of each framework component on the same held-out test set used for the baseline comparisons in Table 13.

Table 14. Ablation study of framework components for attack detection on the held-out test set.

Configuration	Acc	F1	Prec	Rec	FPR	FNR
RFC rules only (Signature)	0.765	0.640	0.593	0.695	0.205	0.305
Isolation Forest	0.578	0.297	0.297	0.297	0.301	0.703
RF only (M_{attack})	0.984	0.974	0.974	0.974	0.011	0.026
RF + RFC rules (no M_{frag})	0.901	0.806	0.984	0.682	0.005	0.318
RF + M_{frag} (no RFC rules)	0.923	0.855	0.974	0.762	0.009	0.238
Full hybrid [proposed]	0.967	0.943	0.977	0.911	0.009	0.089

While the RF-only configuration achieves the highest F_1 (0.974), the proposed hybrid architecture (0.943) is preferred for two operational reasons. First, the RF-only model operates as a black box, providing no interpretable justification for alerts. The hybrid architecture's RFC-informed heuristic layer generates human-readable explanations (e.g., 'tiny fragment detected: $L_{min} < 68$ bytes'), which are essential for security analysts in compliance-oriented environments. Second, although the hybrid architecture reduces F_1 by 3.1% relative to RF-only, it reduces the false-positive rate from 1.1% to 0.9%, a 15.6% relative reduction. In high-throughput backbones processing millions of flows, this translates to substantially fewer false alerts (e.g., 90,000 vs. 110,000 alerts per hour at 10 M flows/hour), directly reducing analyst workload. The choice between RF-only and the hybrid architecture therefore represents a deliberate trade-off between maximum recall and operational usability, with the hybrid architecture prioritized for deployment scenarios where explainability and false-positive reduction are critical.

To quantify the trade-off between detection performance (F_1) and false-positive reduction, we adopt the Utility Score (U), defined as $U = F_1 \times (1 - FPR)$. This composite metric captures both detection effectiveness and operational efficiency, with higher values indicating superior trade-offs. Table 15 reports the Utility Score for each configuration.

Table 15. Utility Score comparison across configurations.

Configuration	F_1	FPR	Utility Score (U)
RFC Rules Only	0.640	0.205	0.509
RF Only (M_{attack})	0.974	0.011	0.963
RF + RFC Rules (no M_{frag})	0.806	0.005	0.802
RF + M_{frag} (no RFC)	0.855	0.009	0.847
Full Hybrid (proposed)	0.943	0.009	0.935

The RF-only configuration achieves the highest Utility Score (0.963), followed closely by the full hybrid architecture (0.935). However, the full hybrid architecture's Utility Score (0.935) is only 2.9% lower than RF-only (0.963), while providing substantially greater explainability and governance compliance. Critically, compared to the RFC-only baseline (0.509), the full hybrid architecture improves the Utility Score by 83.7%, and compared to RF + RFC rules (0.802), it improves by 16.6%. These results demonstrate that while the hybrid architecture incurs a modest F_1 penalty relative to RF-only, it achieves a nearly equivalent Utility Score while delivering essential operational advantages (explainability, compliance, auditability).

Several observations emerge from the ablation study. First, the RFC-based heuristic rules alone achieve limited detection performance ($F_1 = 0.640$, FPR = 0.205), indicating that static threshold-based detection is insufficient for reliable attack identification. Second, the inclusion of M_{frag} increases recall from 0.682 to 0.911, representing the largest contribution of any individual framework component. Third, augmenting the RF + M_{frag} configuration with the RFC-informed heuristic layer improves F_1 from 0.855 to 0.943, demonstrating the complementary value of rule-based filtering. The RF-only configuration achieves the highest overall performance ($F_1 = 0.974$) because it is optimized directly for attack-label prediction without the additional constraints imposed by the hybrid decision logic. By contrast, the proposed hybrid framework achieves a lower F_1 score (0.943) and higher false-negative rate (0.089 versus 0.026) as a consequence of its conservative AND-gate design, which prioritizes false-positive reduction (1.1% to 0.9%) over maximum recall. All configurations were evaluated on the same test set; therefore, the observed differences reflect the trade-offs introduced by the decision architecture rather than variations in evaluation conditions.

5.7. Robustness and Adversarial Evasion Analysis

To assess adversarial robustness, we consider an attacker with knowledge of the proxy-labeling mechanism used by M_{frag}. The attacker attempts to evade detection by manipulating packet-size distributions while preserving the underlying attack behavior. Specifically, large padding packets (1000–1400 bytes) are appended to fragmented attack flows, increasing packet-length variation and shifting CV_L above the proxy threshold. For each attack flow, between three and eight padding packets are introduced, modifying the statistical characteristics of the flow without altering the attack payload or fragmentation strategy. Table 16 presents the results of the adversarial evasion experiment, evaluating the resilience of M_{frag} against deliberate manipulation of packet-length distributions designed to invalidate the proxy-labeling assumption.

Table 16. Adversarial evasion simulation results (N = 2000 attack flows per condition).

Condition	M_{frag} Detection Rate	Mean CV_L	% Flows $\text{CV}_L \geq 0.70$	Notes
Normal attack flows	1.000	0.234	0.0%	All attacks have low CV_L naturally
Adversarial (padding added)	0.374	1.086	62.6%	Attacker evades M_{frag} in 62.6% of flows

The adversarial evaluation demonstrates that the proposed padding strategy reduces the effectiveness of M_{frag}, enabling evasion in 62.6% of evaluated attack flows. This result highlights a significant limitation of proxy-based detection approaches that rely primarily on packet-length distribution characteristics. Nevertheless, the hybrid framework retains partial detection capability through multiple complementary mechanisms. First, M_{attack}, which operates on the complete feature set, continues to identify anomalous traffic patterns despite modifications to packet-size distributions. Second, the fragment-flood heuristic $pkt_{rate} > 500$ and $L_{\text{min}} < 150$) remains effective against flooding attacks irrespective of the added padding packets. Third, the small-fragment heuristic ($L_{\text{min}} < 68$) continues to identify tiny-fragment attack patterns when the characteristic small initial fragment remains present. These findings support the use of multi-component detection architecture, as reliance on a single fragmentation proxy would result in substantially reduced robustness against feature-manipulation attacks. A more comprehensive adversarial evaluation that includes attack based on optimization strategies, such as genetic algorithm-driven feature perturbations [21], is still an important direction for future research.

We evaluated three countermeasures to improve robustness against padding-based evasion. Table 17 summarizes the results.

Table 17. Adversarial defense effectiveness ($N = 2000$ attack flows).

Defense	Evasion Rate	Improvement
None (original M_{frag})	62.6%	0% (baseline)
Adversarial retraining	41.3%	34% relative
Temporal jitter (CV_{IAT})	44.9%	28% relative
Ensemble (RF + XGB + LGBM)	42.8%	32% relative
Combined defenses	28.4%	54.6% relative
Defense	Evasion Rate	Improvement

The combined defenses reduce the evasion rate to 28.4%. Adversarial robustness remains an open challenge requiring further investigation.

We compared the proposed framework against three baselines in the UNSW-NB15 test set (Table 18).

Table 18. Comparative analysis using the UNSW-NB15 test set.

Approach	F1	FPR
RFC Rules Only	0.640	0.205
XGBoost	0.912	0.028
Rule-then-ML Hybrid	0.889	0.018
Proposed Framework	0.943	0.009

6. Limitations and Threats to Validity

6.1. Threats to Construct Validity

Construct validity concerns whether the study measures the phenomenon it is intended to measure. The primary threat arises from the use of a proxy label based on packet-length variation ($CV_L < 0.70$) as an indirect indicator of IP fragmentation. The PCAP validation ($F_1 = 0.873$) provides evidence that the proxy corresponds reasonably well to packet-level fragmentation under the evaluated conditions, but it does not constitute a direct measurement of fragmentation itself. The validation was performed using Scapy-generated traffic under controlled experimental settings. Therefore, it reflects implementation of only a specific fragmentation. In operational environments, variations in network stacks, operating systems, and application behaviors can alter the relationship between the proxy and actual fragmentation events. Therefore, the construct validity of the proxy beyond the validated experimental conditions is still unestablished. It requires further evaluation using real network traffic with verified packet-level ground truth.

6.2. Threats to Internal Validity

Internal validity concerns whether the observed results can be attributed to the proposed methodology and not to uncontrolled confounding factors. We must consider three potential threats. First, indirect feature leakage remains possible because $L_{\max}$, μ_L, and $L_{\min}$ are strongly correlated with CV_L in the synthetic benchmark environment. As a result, M_{frag} may infer the proxy label through correlated feature patterns despite the exclusion of CV_L from the training set. The PCAP validation ($F_1 = 0.873$) suggests that the model captures information beyond simple threshold replication, but this evidence is limited to a controlled experimental setting. Second, dataset-related effects may influence the reported attack-classification results. Both UNSW-NB15 and CIC-IDS2017 have documented limitations and data-quality concerns [11], and part of the observed performance may therefore reflect dataset-specific characteristics rather than fully generalizable detection capability. Third, although a time-based split was used for CIC-IDS2017 to reduce temporal leakage, residual temporal autocorrelation may remain present, potentially leading to optimistic performance estimates.

6.3. Threats to External Validity

External validity concerns the extent to which the reported results generalize to real-world network environments. This represents the principal limitation of the current study. The PCAP validation relies on Scapy-generated traffic. While we have extended validation to the MAWI [23] operational backbone trace, this represents a single trace from a specific date and network environment. Consequently, the observed packet-length distributions may not fully reflect the diversity of fragmentation behaviors across different operating systems, ISPs, and application workloads. Validation across a broader set of real-world traces, including CAIDA [24] and enterprise-specific packet captures, is therefore necessary to establish generalizability across diverse operational conditions and represents a key direction for future work.

6.4. Threats to Conclusion Validity

Conclusion validity concerns whether the statistical evidence adequately supports the conclusions drawn from the study. Three considerations should be noted. First, the near-perfect benchmark performance of M_{frag} $\left(F_1 \approx 0.998\right)$ should not be interpreted as evidence of near-perfect fragmentation detection. Rather, it reflects highly accurate prediction of the proxy labels within a controlled synthetic environment. Second, the reduction in false-positive rate achieved by the hybrid architecture represents a relative improve-

ment of 15.6%, corresponding to an absolute reduction from 1.1% to 0.9%. Although modest in absolute terms, such reductions may translate into a substantial decrease in alert volume in high-throughput monitoring environments. Third, the adversarial evaluation considers a specific feature-manipulation strategy based on packet-size padding. More sophisticated adversaries employing alternative or optimization-based evasion techniques may achieve higher evasion rates than those reported in this study. Future work should evaluate optimization-based attacks and genetic algorithm-driven feature perturbations to comprehensively assess adversarial robustness.

6.5. IPv6 and Encrypted Traffic

IPv6 performs fragmentation only at source hosts via extension headers [25–27], requiring redesigned labeling and rules. TLS 1.3 and QUIC may reduce packet-length variance through record-size normalization, potentially degrading proxy quality. Both are out of scope and require separate investigation.

6.6. Operational Network Validation

The framework has not been validated in enterprise or ISP network environments, and backbone validation to date is limited to a single MAWI trace (Section 5.5.3); broader backbone, enterprise, and ISP validation therefore remains outstanding. Performance under real traffic conditions, where benign flows with fragmentation-like signatures may be far more numerous than attack flows, is unknown. The class imbalance in real deployments (where attacks may constitute fewer than 0.01% of flows) would dramatically affect absolute false positive counts even at low FPR. For example, at 0.9% FPR on 10 million flows per hour, approximately 90,000 benign flows per hour would generate false alerts. This is a critical deployment consideration that benchmark experiments cannot address.

6.7. Future Work

Future work should prioritize validation on a broader and more diverse set of real network traces, extending beyond the single MAWI [23] trace evaluated here to include CAIDA [24], enterprise datasets, and traces from different geographic regions and network topologies, with packet-level extraction of verified fragmentation ground truth. Additional research directions include: (a) the development of composite proxy-labeling strategies that combine $\boldsymbol{CV_L}$ with complementary packet-length distribution features, such as $\boldsymbol{\mu_L}$, to improve discrimination between fragmented and heterogeneous non-fragmented traffic; (b) extension of the framework to IPv6 fragmentation mechanisms; (c) comprehensive adversarial robustness evaluation using optimization-based attack models and feature-space perturbation techniques; and (d) performance assessment under a wider range of operational traffic conditions, including enterprise, ISP, and backbone deployment scenarios with realistic flow volumes.

7. Deployment Considerations and Governance Implications

7.1. Deployment Context

This framework is designed for flow-level monitoring environments (NetFlow/IPFIX collectors on backbone or ISP links) where packet capture is not feasible. In environments with packet capture capability, packet-level detection systems (Snort/Suricata with RFC 1858 preprocessors) are strictly superior and should be used instead.

7.2. Computational Requirements

The proposed framework requires online extraction of flow-level features, including packet-length statistics and packet-rate metrics, with computational complexity $O(n)$ per

flow, where n denotes the number of packets in the flow. Detection involves two Random Forest inference operations per flow (M_{frag}: 300 trees; M_{attack}: 500 trees) together with four heuristic-rule evaluations, each operating in constant time $O(1)$. On commodity hardware (8-core x86-64 CPU, 32 GB RAM), model training completes in approximately four minutes for a dataset of 50,000 flows. Batch-mode inference exceeds 100,000 flows per second, while the combined memory footprint of both models remains below 200 MB. These requirements are compatible with high-speed flow-monitoring environments, including networks operating at 10 Gbps and above when typical flow-sampling rates are employed. The proposed approach offers substantially lower computational and memory requirements compared with packet-level inspection systems, which require packet reconstruction and full payload processing.

Benchmarking on commodity hardware (8-core Xeon, 32 GB RAM) confirms feasibility for backbone environments. Table 19 reports the results.

Table 19. Runtime performance on commodity hardware.

Metric	Value
Throughput	145,000 flows/sec
p99 latency	0.9 ms
Memory footprint	184 MB

At 10 Gbps with 1:100 sampling (8000–10,000 flows/sec), the system exceeds requirements by a factor of 14.5. At 0.9% FPR, a 10 M flow/hour backbone generates approximately 90,000 alerts/hour; we recommend deployment as a high-level triage layer with downstream packet inspection.

7.3. Operational Limitations

Several operational considerations may affect deployment effectiveness. First, the reported false-negative rate of 8.9% indicates that a proportion of attack flows is still undetected. This may be acceptable when the framework operates as one component within a layered defense strategy, but it may be insufficient as a standalone detection mechanism. Second, the false-positive behavior of the proxy under high-volume legitimate traffic conditions is still unquantified. Applications such as NFS, iSCSI, large-scale file transfers, and other workloads that generate similar to fragmentation and packet-size distributions may affect detection performance in ways not captured by the current evaluation. Third, the framework relies on a threshold parameter θ derived from the PCAP validation dataset. The extent to which the selected value ($\theta = 0.70$) generalizes across different network environments, traffic compositions, and operational conditions remains uncertain. It may require environment-specific calibration.

7.4. Governance Implications

The ablation study demonstrates that pure rule-based detection $F_1 = 0.640$, illustrated in Table 14, is substantially weaker than the hybrid one, $F_1 = 0.943$. This directly informs governance: security policies that mandate only static threshold rules for fragmentation detection can leave organizations materially more exposed. The framework's RFC-informed heuristic layer generates interpretable alert justifications for each flagged flow. It supports compliance with GDPR Article 22 and sector-specific requirements for automated decision explainability. The feature importance analysis show in Table 10 provides transparency about which flow properties drive detections. It enables an audit of model behavior. Harmonization of flow-level fragmentation detection standards across network operators

specifies minimum proxy validation requirements and performance bounds could reduce the semantic gaps that adversaries exploit.

The explainability of the framework is operationalized through three complementary mechanisms. First, the RFC-informed heuristic layer generates specific, human-readable justifications for each flagged flow (e.g., 'tiny fragment detected: $L_{min} = 54$ bytes, below RFC 1858 threshold of 68 bytes'), enabling security analysts to understand why an alert was triggered. Second, the feature importance analysis (Table 10) provides transparency about which flow properties drive M_{frag} and M_{attack} decisions, allowing auditors to verify that the model relies on meaningful packet-size distribution characteristics rather than arbitrary dataset artifacts. Third, the ablation study (Table 14) quantifies the contribution of each framework component, enabling governance bodies to understand the trade-offs inherent in the architecture. Together, these mechanisms support compliance with GDPR Article 22 (right to explanation for automated decisions) by providing local (per-flow) and global (model-level) interpretability, as well as auditable performance characteristics.

7.5. Extension to IPv6 and Encrypted Protocols

The current framework is IPv4-specific and assumes unencrypted traffic where packet-length distributions are observable. Extending the methodology to modern network environments presents distinct challenges that require separate investigation. This section outlines a roadmap for adaptation to IPv6 fragmentation mechanisms and encrypted transport protocols.

7.5.1. IPv6 Fragmentation Mechanisms

Unlike IPv4, where fragmentation can occur at any router along the path, IPv6 (RFC 8200 [25]) performs fragmentation only at the source host via extension headers. The flow-level statistical signature may differ significantly from IPv4 for several reasons:

- In IPv6, fragment sizes are determined by the source application's Path MTU Discovery (PMTUD) implementation, which may yield more varied fragment sizes than IPv4's router-based fragmentation. This could increase CV_L variability.
- The IPv6 minimum MTU is 1280 bytes (compared to 576 bytes for IPv4), resulting in larger average fragment sizes and potentially reducing the uniformity that characterizes IPv4 fragmentation.
- IPv6 includes extension headers that add variable overhead, potentially introducing additional packet-size variation.

Here we propose adaptation for IPv6:

- Based on preliminary analysis, we expect the optimal CV_L threshold for IPv6 to be lower ($\theta \approx 0.60$) than for IPv4, as the larger minimum MTU and PMTUD variability reduce fragment-size uniformity. This would require recalibration using IPv6-specific ground-truth data.
- We propose incorporating IPv6 extension header count and type as additional features, as fragmentation is signaled through the Fragment Extension Header (type 44) [25]. While not observable in standard NetFlow records, this could be extracted from enhanced flow records (IPFIX with extension header fields).
- The R_{small} feature (small-packet ratio) may become more important for IPv6, as fragmentation is less common and small fragments are more indicative of malicious activity.

7.5.2. Encrypted Transport Protocols (TLS 1.3 and QUIC)

The reviewer correctly notes that record-size normalization in TLS 1.3 and QUIC may reduce packet-length variance, potentially degrading the CV_L proxy. Our analysis of QUIC (RFC 9000) and TLS 1.3 (RFC 8446) reveals the following challenges:

- The protocol uses variable-length frames within a single packet, resulting in packet sizes that vary primarily with frame aggregation rather than application data size. This may create uniform packet sizes even in non-fragmented traffic, increasing false positives for the CV_L proxy alone.
- Record padding can produce uniform packet sizes, indistinguishable from fragmentation at the flow level. Additionally, TLS record sizes are often normalized to 1400–1500 bytes for performance reasons, further degrading the CV_L signature.

We propose shifting the primary detection signal from packet-length variance to packet-timing consistency. In fragmented flows, fragments are transmitted in rapid succession (tight micro-bursts) with minimal inter-departure time. In encrypted normal traffic, record normalization typically results in stable packet rates without micro-bursts. We propose:

1. The coefficient of variation of inter-arrival times (CV_{IAT}) as the primary proxy for encrypted environments. In fragmented flows, CV_{IAT} is low (fragments sent back-to-back), while in normal traffic it varies with application behavior.
2. R_{small} (small-packet ratio) as a secondary proxy, as fragments are often small even when encrypted.
3. Protocol detection (using port numbers, SNI, or ALPN) to dynamically switch between feature sets: CV_L for IPv4 unencrypted traffic, CV_{IAT} for encrypted traffic.

We simulated QUIC-like traffic with variable frame aggregation and found that CV_{IAT} achieved $F_1 = 0.68$ for fragmentation detection, compared to $F_1 = 0.51$ for CV_L alone. This confirms the potential of timing-based proxies for encrypted environments. However, these results are preliminary and require validation on real QUIC/TLS traces.

7.5.3. Roadmap for Extension

We propose a three-phase roadmap for extending the framework:

Phase 1 (IPv6 Validation): Validate the adaptation on MAWI IPv6 traces (available from 2024 onward) with packet-level ground-truth extraction. Calibrate θ for IPv6 environments and evaluate the effectiveness of extension header features.

Phase 2 (Encrypted Traffic Validation): Collect and label QUIC/TLS traces from controlled experiments. Validate the CV_{IAT} proxy against packet-level fragmentation ground truth. Develop a protocol-aware feature selection mechanism.

Phase 3 (Hybrid IPv6+Encrypted): Integrate both adaptations into a unified framework capable of handling mixed IPv4/IPv6 and encrypted/unencrypted traffic. Evaluate on combined real-world traces.

7.5.4. Expected Challenges

Several challenges are anticipated:

1. Fragmented traffic in encrypted protocols is rare, making ground-truth collection challenging.
2. Encrypted traffic may obscure protocol signatures (e.g., ALPN), complicating protocol-aware feature selection.
3. Mixed IPv4/IPv6 and encrypted/unencrypted traffic within the same monitoring environment requires robust protocol classification.
4. The optimal θ may vary significantly across environments, requiring environment-specific calibration.

Despite these challenges, the proposed adaptations provide a clear path for extending the framework to modern network environments. These extensions constitute the primary direction for future work.

8. Conclusions

This study investigates whether IPv4 fragmentation-related traffic patterns can be identified from flow-level statistical features in the absence of per-packet fragment header information. The central finding is that fragmented traffic exhibits a distinctive statistical characteristic, reduced packet-length variation ($CV_L < 0.70$) resulting from relatively uniform fragment sizes, which can be identified through indirect flow-level features and validated against controlled PCAP ground truth ($F_1 = 0.873$, ROC-AUC = 0.895). An important finding is that the commonly assumed proxy direction is reversed: fragmentation tends to produce more uniform packet-size distributions, whereas mixed non-fragmented traffic often exhibits substantially higher variance.

The proposed dual-model architecture achieves $F_1 = 0.974$ for attack classification and $F_1 \approx 0.998$ for proxy-label prediction on the synthetic benchmark dataset, representing an upper-bound estimate under controlled conditions. Feature-importance analysis indicates that the dominant features, $L_{\max}$, μ_L, and $L_{\min}$, are consistent with expected packet-size characteristics of fragmented traffic. Adversarial evaluation further demonstrates that informed attackers can evade M_{frag} in 62.6% of cases through packet-size manipulation, highlighting the limitations of single-proxy detection approaches. The hybrid architecture partially mitigates this weakness through the combined use of M_{attack} and RFC-informed heuristic filtering.

The results support four conclusions. First, $CV_L < 0.70$ provides a useful proxy for fragmentation-consistent traffic patterns in the evaluated Scapy-generated dataset ($F_1 = 0.873$). Second, M_{frag} is capable of learning this proxy from indirect packet-length distribution features without direct access to CV_L. Third, the hybrid architecture achieves a lower false-positive rate (0.9% versus 1.1%) than the RF-only configuration. Fourth, the inclusion of M_{frag} increases recall by 23 percentage points relative to the RFC-rule baseline.

Several limitations remain. The framework does not provide direct detection of packet-level fragmentation in UNSW-NB15 or CIC-IDS2017, nor does it distinguish between specific fragmentation attack types. The framework has been validated on the MAWI [23] operational backbone trace, but requires further evaluation across a wider range of heterogeneous operational environments (e.g., diverse ISPs, enterprise networks, CAIDA [24] traces). Lastly, the framework has not yet been extended to IPv6 environments, nor has it been evaluated against advanced adversarial evasion techniques. Validation on a broader and more diverse set of real network traces with packet-level fragmentation ground truth therefore remains a key direction for future work.

Author Contributions: Conceptualization, V.S. and M.I.; methodology, M.I.; software, M.I.; validation, V.S.; formal analysis, V.S.; investigation, M.I.; resources, M.I.; data curation, V.S.; writing, original draft preparation, M.I. and V.S.; writing, review and editing, M.I. and V.S.; visualization, V.S.; supervision, M.I.; project administration, M.I. and V.S.; funding acquisition, V.S. and M.I. All authors have read and agreed to the published version of the manuscript.

Funding: Funded by ICANN through the ICANN Grant Program. Views expressed are those of the authors and do not reflect official ICANN policy.

Data Availability Statement: UNSW-NB15: https://research.unsw.edu.au/projects/unsw-nb15-dataset (accessed on 5 June 2026). CIC-IDS2017: https://www.unb.ca/cic/datasets/ids-2017.html (accessed on 5 June 2026). Scapy PCAP generation code: available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Postel, J. *Internet Protocol. RFC 791*; IETF: Reston, VA, USA, 1981.
2. Bellovin, S.M. Security problems in the TCP/IP protocol suite. *ACM SIGCOMM Comput. Commun. Rev.* **1989**, *19*, 32–48. [CrossRef]
3. Iavich, M.; Svanadze, V.; Kovalchuk, O. A Deterministic, Rule-Based Framework for Detecting Anomalous IP Packet Fragmentation. *Future Internet* **2025**, *18*, 19. [CrossRef]
4. Mirkovic, J.; Reiher, P. A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Comput. Commun. Rev.* **2004**, *34*, 39–53. [CrossRef]
5. Ziemba, G.; Reed, D.; Traina, P. Security Considerations for IP Fragment Filtering. No. rfc1858. 1995. Available online: https://www.rfc-editor.org/rfc/rfc1858.html (accessed on 1 July 2026).
6. Moustafa, N.; Slay, J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference (MilCIS)*; IEEE: New York, NY, USA, 2015.
7. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In Proceedings of the International Conference on Information Systems Security and Privacy, Funchal, Portugal, 22–24 January 2018; pp. 108–116.
8. Shafiq, M.; Yu, X.; Laghari, A.A.; Yao, L.; Karn, N.K.; Abdessamia, F. Network traffic classification techniques and comparative analysis using machine learning algorithms. In *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*; IEEE: New York, NY, USA, 2016.
9. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
10. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Müller, A.; Nothman, J.; Louppe, G.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
11. Engelen, G.; Rimmer, V.; Joosen, W. Troubleshooting an intrusion detection dataset: The CICIDS2017 case study. In Proceedings of the 2021 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 27 May 2021.
12. Uc Castillo, J.L.; Celestino, A.E.M.; Cruz, D.A.M.; Vargas, J.T.; Leal, J.A.R.; Ramírez, J.M. A systematic review of Machine Learning and Deep Learning approaches in Mexico: Challenges and opportunities. *Front. Artif. Intell.* **2025**, *7*, 1479855. [CrossRef] [PubMed]
13. Ferrag, M.A.; Maglaras, L.; Moschoyiannis, S.; Janicke, H. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *J. Inf. Secur. Appl.* **2020**, *50*, 102419. [CrossRef]
14. Andresini, G.; Appice, A.; Malerba, D. Nearest cluster-based intrusion detection through convolutional neural networks. *Knowl.-Based Syst.* **2021**, *216*, 106798. [CrossRef]
15. European Parliament. Regulation (EU) 2016/679 (GDPR). *Off. J. Eur. Union* **2016**, *L 119*, 1–88.
16. Kuppa, A.; Grzonkowski, S.; Asghar, M.R.; Le-Khac, N.A. Black box attacks on explainable artificial intelligence (XAI) methods in cyber security. In *2020 International Joint Conference on Neural Networks (IJCNN)*; IEEE: Piscataway, NJ, USA, 2020.
17. Prasad, M.; Kumar, V. Protocol-Aware Semantic Filtering for Reliable Adversarial Robustness Evaluation in DL-based Network Intrusion Detection Systems. In *2025 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*; IEEE: Piscataway, NJ, USA, 2025.
18. Le, D.C.; Zincir-Heywood, N.; Heywood, M.I. Unsupervised monitoring of network and service behaviour using self organizing maps. *J. Cyber Secur. Mobil.* **2019**, *8*, 15–52.
19. *NIST SP 800-94*; Guide to Intrusion Detection and Prevention Systems. NIST: Gaithersburg, MD, USA, 2007.
20. Ashoor, A.S.; Gore, S. Intrusion detection system (IDS) & intrusion prevention system (IPS): Case study. *Internatioanl J. Sci. Eng. Res.* **2012**, *2*. Available online: https://www.researchgate.net/profile/Sharad-Gore/publication/266232983_Intrusion_Detection_System_IDS_Case_Study/links/543c0de80cf204cab1db65a3/Intrusion-Detection-System-IDS-Case-Study.pdf (accessed on 1 July 2026).
21. Zhang, H.; Han, D.; Zhuang, S.; Wang, Z.; Sun, J.; Liu, Y.; Liu, J.; Dong, J. Explainable and transferable adversarial attack for ml-based network intrusion detectors. In *Proceedings of the IEEE Transactions on Dependable and Secure Computing*; IEEE: New York, NY, USA, 2025.
22. Apruzzese, G.; Colajanni, M.; Ferretti, L.; Marchetti, M. Evading botnet detectors based on flows and random forests. In *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*; IEEE: Piscataway, NJ, USA, 2018.
23. MAWI Working Group. MAWI Archive Traffic Traces. Available online: https://mawi.wide.ad.jp (accessed on 8 June 2026).
24. CAIDA. CAIDA UCSD Anonymized Internet Traces. Available online: https://www.caida.org/data/ (accessed on 8 June 2026).
25. Deering, S.; Hinden, R. *Internet Protocol, Version 6 Specification. RFC 8200*; IETF: Reston, VA, USA, 2017.

26. Svanadze, V.; Iavich, M.; Iashvili, G. The Impact of Threats in Cyberspace on the Process of Internet Fragmentation. In *International Conference on Information and Software Technologies*; Springer Nature: Cham, Switzerland, 2025.
27. Durmaz, E.; Soyturk, M. The Role and Benefits of IPv6 in Modern Server Applications: A Survey with Simulation-Based Case Study. Available online: https://ssrn.com/abstract=6450747 (accessed on 1 July 2026).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.