DDoS Attack Detection in 5G Control Plane Using Machine Learning

Anutosh Tiwari and Samaresh Bera
Dept. of Computer Science and Engineering
Indian Institute of Technology Jammu, India
Email: anutosh05@gmail.com, s.bera.1989@ieee.org

Zahid Akhtar
Dept. of Electrical and Computer Engineering
State University of New York Polytechnic Institute, USA
Email: akhtarz@sunypoly.edu

***Abstract*—In this paper, we analyze a detection framework for DDoS attacks targeting the 5G control plane using SCTP handshake flooding. Our work evaluates both static algorithms and dynamic machine learning approaches to detect traditional as well as low-rate DDoS attack patterns. We conduct multiple SCTP handshake flooding experiments to develop the first-ever dataset for this attack type in the 5G control plane. By extracting both direct and derived features, we assess several supervised machine learning models under high-rate and low-rate DDoS attack scenarios. Experimental results reveal that while the ML models achieve near-perfect accuracy for fast and volumetric attacks, unseen patterns and low-rate attacks remain challenging due to their subtle nature. These findings underscore a trade-off between detection accuracy and real-time processing efficiency, particularly when employing ensemble-based methods. Overall, our research demonstrates that a hybrid approach – integrating load balancing, rate limiting, and ML-based detection – holds significant promise for enhancing the security of the 5G control plane.**

***Index Terms*—5G control plane, DDoS detection, SCTP handshake flooding, Load balancing, Rate limiting, Machine learning**

I. Introduction

5G leverages innovative technologies such as network function virtualization (NFV), software-defined networking (SDN), and network slicing to overcome the limitations of 4G [1]. In the era of rapidly expanding connectivity, 5G is revolutionizing communication by supporting millions of devices and is expected to connect billions, including IoT devices, mobile phones, computers, and large-scale servers. However, the increasing prevalence of IoT devices, many of which have limited security, has heightened the risk of distributed denial-of-service (DDoS) attacks. While the advancements in 5G significantly enhance network capabilities, they also introduce new security challenges that may lead to widespread service outages.

In service-based architecture of 5G networks (5G-SA) [1], softwarized implementations using NFV have largely replaced traditional hardware-based architectures. Furthermore, it enables the consolidation of multiple network functions on a single platform, allowing for rapid deployment and the flexible addition of new capabilities. However, this consolidation can create single points of failure in case of DDoS attacks, as heavy load on one function may compromise the computational resources available to others. This risk underscores the necessity of securing the control plane, which is responsible for critical operations such as authentication, session management, and mobility management. In the 5G-SA, access and mobility management function (AMF) handles the control signals for association of a base-station (gNB) and end-user authentication to the core network using Stream Control Transmission Protocol (SCTP). When a user initiates a connection with the network, an `SCTP INIT` message is sent to the AMF, which then completes a full handshake to establish a new connection and allocate resources. As the AMF is directly exposed to user interactions, it becomes a prominent target for attackers [2], [3]. There exist static approaches, such as static rate limiting, to deal with the DDoS attacks at the AMF. However, sophisticated attackers can exploit the predictable behavior of static mitigation techniques, by dynamically adjusting traffic generation rates. As a result, there is a growing need for dynamic and intelligent detection mechanisms, such as machine learning-based DDoS attack detection, and traffic behavior analysis. These advanced approaches can learn from real-time traffic patterns, identify deviations from normal behavior, and respond proactively to emerging threats, providing a more robust defense against increasingly complex and stealthy DDoS attack strategies.

In this work, we build a fully softwarized 5G control plane that connects user equipment and gNBs to 5G core network. We integrate a load balancer between the AMF and the user equipment to balance the traffic load on AMFs. Our main goal is to understand the limitations of static algorithms and explore the performance of dynamic machine learning (ML) algorithms to detect attacks on the AMF using SCTP. Since publicly available DDoS datasets for the N2 interface are limited, we generate a unique dataset by simulating various DDoS attack scenarios against the AMF, creating a realistic environment to evaluate the proposed methods. We evaluate the machine learning algorithms in two key ways. First, we measure their detection accuracy using the F1 score, which balances the precision and recall of each model. Second, we look at how fast they can process incoming requests, since any delay could affect network performance in a live traffic scenario. In other words, these algorithms need to be both accurate and quick to be effective in real-world deployments. Additionally, we

address a common research limitation where most studies test on static datasets. Instead, we challenge our models with attack patterns that are not part of the training data to see how well they perform on unseen threats. This step is crucial to ensure that our solutions are robust and ready for the unpredictable nature of real-world attacks. The key technical contributions are as follows:

- First publicly available SCTP handshake flooding dataset for the 5G N2 interface, merging SCTP control-plane traces captured from 5G testbed with controlled attack simulations targeted at the AMF.
- Realistic softwarized 5G-SA testbed with AMF auto-scaling and load balancing, enabling reproducibility of handshake flooding under realistic core-network conditions.
- Comprehensive and protocol-aware feature engineering combined with packet-level attributes with temporal handshake statistics computed over 5 seconds and 30 seconds windows.
- Comparative analysis under seen and unseen DDoS and low-rate DDoS (LDDoS) attacks, highlighting model-specific robustness limitations.
- Joint accuracy–latency evaluation, reporting macro-averaged detection metrics together with per-record processing time to assess real-time feasibility.

The rest of the paper is organized as follows. Section II presents an overview of the existing works related to DDoS detection, SCTP vulnerabilities, and machine learning-based mitigation strategies. Section III discusses the architecture of the software implementation of 5G control plane and the inherent vulnerabilities in the SCTP protocol that make it a target for handshake-based flooding attacks. Section IV details the dataset generation process, including attack simulation and data creation methodologies. Section V describes the evaluation metrics adopted to benchmark the models. Section VI presents and analyzes the results, comparing detection performance across multiple algorithms. Finally, Section VII summarizes the findings and outlines future directions.

II. Related Work

Research on DDoS detection and mitigation within the context of the 5G control plane, particularly against SCTP-based flooding attacks, remains relatively limited. However, several studies have contributed valuable insights in related areas, such as SCTP vulnerabilities, SDN-based defenses, and machine learning-based attack detection [2], [4]–[10].

Rathgeb et al. [4] conducted one of the early foundational studies on the vulnerabilities of the SCTP. Their work demonstrated that even after the completion of SCTP's four-way handshake, the protocol remains susceptible to certain types of denial-of-service (DoS) attacks. This raised concerns about SCTP's robustness in signaling-heavy environments like 5G networks. Dey et al. [9] explored detection strategies for DoS attacks in 5G core networks, specifically targeting synchronization vulnerabilities in network slicing. Their approach highlights how multi-slice synchronization can become a critical weak point in 5G security.

Within the realm of SDN, Syafril et al. [10] analyzed the performance of intrusion prevention systems (IPS) against TCP and UDP flood attacks. Their study emphasized the flexibility of SDN in deploying lightweight yet responsive security mechanisms, a principle that aligns well with the dynamic nature of 5G. Park et al. [2] focused on machine learning-based detection methods for complex DDoS attacks in 5G standalone (SA) core environments. Their system achieved an impressive detection accuracy of 98%, demonstrating the potential of ML algorithms for identifying fast-evolving threat patterns in real-time. Alashhab et al. [7] investigated low-rate DDoS (LDDoS) attacks and their impact on SDN-based architectures. The authors proposed machine learning-based detection frameworks but also pointed out the limitations of deep learning approaches in generalizing across low-frequency attack patterns, an issue particularly relevant to LDDoS attacks on 5G signaling layers.

While prior work has explored ML for DDoS in SDN/IoT or other 5G layers, there is a lack of publicly available datasets and a comprehensive evaluation framework specifically for SCTP handshake flooding attacks against the 5G control plane's AMF, particularly assessing model robustness against novel, low-rate attack patterns. This work addresses this gap by creating a 5G control-plane attack dataset and proposing ML-based DDoS attack detection approach evaluated on a 5G softwarized testbed.

III. Vulnerabilities in Control Plane of 5G Network

A. Softwarized Implementation of 5G Network

We consider the softwarized implementation of 5G network with load balancing [11], where we deployed a full control-plane setup using Open5GS over a Kubernetes-based virtualized environment. The setup involved containerized deployment of core network functions such as the AMF, SMF, and UPF, along with simulated User Equipment (UE) and gNodeB (gNB) pairs, as shown in Figure 1. We also incorporated a custom load balancing module positioned on the N2 interface to distribute control-plane traffic across multiple AMF instances. The deployment enabled dynamic scaling of network functions and supported load balancing over the active AMF.

B. SCTP Vulnerabilities in 5G Network

SCTP is widely used in the control plane for signaling in modern telecommunication networks, offering advantages such as multi-streaming and multi-homing. The UE and gNB communicate with the control plane functions using the SCTP protocol, where a connection is established by exchanging four messages: a) `INIT`, b) `INIT ACK`, c) `COOKIE ECHO`, and d) `COOKIE ACK`. Consequently, SCTP employs a four-way handshake combined with a cookie-based mechanism as

a primary defense against Denial-of-Service (DoS) attacks. Despite these safeguards, SCTP remains vulnerable to specific attacks, including INIT flooding, long INIT flooding, real cookie flooding, fake cookie flooding, and handshake flooding [12]. In particular, handshake flooding attacks, where a large botnet initiates numerous handshake processes, can be especially damaging. This form of attack forces the system to complete multiple handshakes, thereby consuming significant computational and memory resources, and can severely degrade the availability of 5G services for legitimate users.

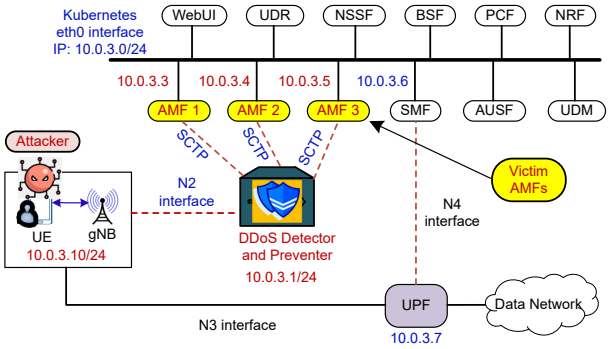

Fig. 1: Network details with auto-scaler and load balancer

IV. Dataset Creation

A. Challenges and Solution

Due to the limited research on these types of attacks and to the best of our knowledge, there is no publicly available dataset specifically for DDoS attacks targeting the SCTP protocol. While a few small datasets for DoS attacks exist [13], [14], none are suitable for analyzing the machine learning algorithms for SCTP DDoS detection. This limitation motivated us to create a custom dataset to support the evaluation of detection methods against `SCTP SYN` flood attacks. For the dataset creation task, it is essential to obtain data from real normal traffic flow for the AMF over the N2 interface in order to accurately mimic the data flow observed in a real network environment. For this purpose, we utilize the dataset from [15]. We extract multiple attributes and relevant information from the provided pcap files to construct a comprehensive dataset representing normal traffic flows. With this baseline in place, we then generate attack data by carrying out DDoS attacks on our softwarized implementation. Finally, we strategically merge the normal and attack datasets to enable robust analysis of the algorithms, as discussed in subsequent sections.

B. Attack Implementation

To emulate flooding attacks, we implemented two distinct strategies: a random interval DDoS and LDDoS attacks. In the random interval attack, multiple User Equipment (UE) devices initiate handshakes with the network, with each UE sending handshake requests at randomly determined intervals. We conducted several tests with varying interval ranges to evaluate the network's resilience under different timing conditions. In contrast, the LDDoS attack involves UEs sending handshake requests continuously over an extended period, with longer and consistent gaps between requests. This method is particularly challenging for virtualized network functions (VNFs) because it continuously loads the system and degrades the performance of other services. For both attack scenarios, we created a botnet of real users and further executed a script with enabled parallel execution to make of UE instances to send handshake request and complete these full requests again and again, allowing us to effectively target the Access and Mobility Management Function (AMF) via the load balancer. All packets passing through the load balancer were traced, and the resulting dataset was constructed with the same attributes as those in the normal traffic dataset.

C. Merging Benign and Malicious Dataset

After generating the attack dataset, the next challenge is to strategically merge it with the normal traffic dataset. This process posed two primary challenges. First, a direct merge could result in conflicting records—such as the same User Equipment (UE) sending both normal and malicious messages at overlapping timestamps—which is unrealistic in practical scenarios. Therefore, redundant and inconsistent entries had to be identified and removed. Second, there was a discrepancy in packet transmission and processing times between the AMF implementation used in our setup and the AMF from which the normal dataset was collected. Such timing inconsistencies could reduce the reliability of the final dataset for time-sensitive analysis and pattern recognition.

To address the first issue, we implement a custom masking strategy to exclude normal traffic records from UEs that were flagged as compromised during specific time intervals in the attack dataset. This ensures that no single UE was simultaneously represented as both benign and malicious during any time window. For the second issue, we conduct a comparative analysis of the handshake duration in both datasets. By calculating the median handshake time in the normal dataset and aligning it with the initial handshake duration in the attack dataset, we apply a scaling factor to adjust the timestamps in the attack data. This ensures temporal consistency across both datasets.

By resolving these challenges, the normal and attack datasets are merged along the time axis, thereby creating a unified and temporally coherent dataset that realistically simulates DDoS attacks occurring at different time intervals within a 5G control plane scenario. The dataset contains $1,226,745$ SCTP packet samples collected on the N2 interface of a softwarized 5G standalone (5G-SA) testbed. Of these, $37,324$ (3.04%) are labeled as flooding (`is_flood` = 1) and $1,189,421$ (96.96%) as benign. Packet timestamps span 0.0 to $8,919.317088$ seconds (≈ 2.48 hours) with an average throughput of 137.5 packets/seconds. The processed ML representation contains 56 features (9 numeric/temporal + 47 one-hot `Message_Type` indicators). The dataset contains 142 unique source IPs, 142 unique destination IPs. The created

dataset is classified into three categories: `normal`, `DDoS`, and `LDDoS`. This comprehensive approach enables our detection models to accurately distinguish between benign and malicious traffic in the context of DDoS attacks on the 5G control plane. The dataset is made publicly available through the IEEE Dataport [16].

D. Feature Extraction

With the creation of the dataset, the next critical step involves extracting meaningful attributes that capture the behavior and state of each network flow. These attributes are designed to provide insights such as the number of `SCTP-INIT` requests within a specified time window, the ratio of `INIT` messages to regular traffic, and patterns in packet flow. Due to the limited prior research on applying machine learning techniques to SCTP flooding attacks, we also incorporated insights from studies on IP and UDP flooding attacks to guide our feature extraction process [8]. The specific features, including packet headers, that are used for dataset creation are as follows: `source_ip`, `destination_ip`, `sequence_timing`, `association request intervals`: $1-5$ seconds for DDoS and $10-15$ seconds for LDDoS, `packet rate` using sliding window (5 seconds), `handshake count` in last 5 seconds using `SCTP INIT`, `repeated handshake requests`, `source entropy`. We note that the dataset presents direct packet traces from the testbed. These features help us in capturing instantaneous and temporal traffic changes due to the DDoS attacks.

E. Rationale of Extracted Features

Based on SCTP protocol behavior, the extracted features are designed to capture handshake dominance, temporal persistence, and source distribution characteristics that are used as the indicators of flooding attacks. We discuss the rationale behind some of the extracted features below.

• **Handshake activity features:** To measure the intensity and dominance of handshake initiation within a sliding temporal window $W_{5\text{sec}}(t)$, the following are calculated:

$$\texttt{\#Handshake}(t) = \sum_{i \in W_{5\text{sec}}(t)} 1_{\text{handshake}}(i) \tag{1}$$

$$\texttt{INIT\_Ratio}(t) = \frac{\texttt{\#INIT in } W_{5\text{sec}}(t)}{\texttt{\#Packets in } W_{5\text{sec}}(t)} \tag{2}$$

$$\texttt{Repeated\_Attempts}(t) = \max(0, \texttt{\#INIT} - 1) \tag{3}$$

Under flooding, the expected handshake rate increases substantially as follows:

$$\mathbb{E}[\texttt{\#Handshake}|\texttt{attack}] >> \mathbb{E}[\texttt{\#Handshake}|\texttt{benign}] \tag{4}$$

Empirically, this increase is ≈ 2.48 to 68.73 (≈ 27.7 times), confirming strong separability.

• **Temporal aggregation windows (5 seconds and 30 seconds):** Short-term windows (5 seconds) capture burst or rapid handshake surges characteristic of volumetric flooding. Whereas the longer aggregation ($\texttt{Handshake\_Mean}_{30\text{sec}}$) captures sustained, low-rate attack scenario, which is typical of LDDoS attacks. The multi-scale temporal representation therefore enables detection of both high-intensity and stealthy low-rate flooding behaviors.

• **Source diversity and entropy:** Distributed attacks typically originate from multiple compromised endpoints. An increase in the number of distinct sources and the entropy of their distribution reflects the randomness and dispersion expected in botnet-driven SCTP flooding, it is distinguishable from the legitimate signaling by stable peers.

• **Message-type indicators:** Specific SCTP control messages – `INIT`, `INIT_ACK`, and `COOKIE_ECHO` – are strongly associated with handshake activity. Encoded message types preserve protocol-state information, enabling models to differentiate malicious handshake-dominated traffic from normal 5G control-plane signaling exchanges.

• **Short-term packet rate:** Packet rate provides contextual traffic intensity, but it is not independently sufficient for SCTP flooding detection. Benign control-plane traffic (e.g., heartbeat and signaling exchanges) can produce high packet rates. Therefore, packet rate must be interpreted jointly with handshake-centric and protocol-aware features.

V. ML Models and Evaluation Metrics

A. Train/Test Split Strategy

To evaluate both in-distribution and out-of-distribution behavior, the dataset was split by scenario (not by random packet sampling), preserving temporal scenario and structure as follows:

- Training: benign traces with a collection of simulated attack scenarios (high-rate and some low-rate patterns).
- Test with `Seen` attacks: traffic with attack timing/distribution similar to training (measures in-distribution performance).
- Test with `Unseen LDDoS`: low-rate and random-interval attack patterns parameterized differently from training (measures generalization).

We avoid random cross-fold validation for final reporting because the temporal and scenario dependencies make random folds unrealistic. However, model hyper-parameter tuning, used for time-consistent validation folds, is drawn from training scenarios to avoid leakage.

B. Machine Learning Models Used

To detect DDoS attacks targeting the AMF over the N2 interface, we explored a range of machine learning models, each offering different advantages based on their underlying learning mechanisms. We use the following ML algorithms: Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Linear Support Vector Classifier (Linear SVC), Gradient Boosting Classifier, Extreme Gradient Boosting (XGBoost), and Gaussian Naive Bayes (Gaussian NB). We limit

our discussion on each of these algorithms, as they are well-established in the literature [5].

C. Evaluation Metrics

To rigorously assess the performance of our proposed models, we evaluate them on four key metrics: macro-average precision, macro-average recall, macro-average F1 score, and the average time taken to process each record. The metrics provide a comprehensive view of both detection accuracy and practical feasibility in real-time systems. Due to the imbalanced nature of the dataset—where normal traffic significantly outweighs attack traffic—traditional accuracy is not a meaningful metric. It may give an inflated sense of performance by favoring the majority class. Instead, we use macro-averaged metrics that treat each class equally, regardless of its frequency, thereby offering a more balanced evaluation.

1) Macro-Average Precision, Recall, and F1 Score: The macro-average approach computes the precision, recall, and F1 score independently for each class and then averages them as follows:

- Precision (Macro-Avg) = $\frac{1}{C}\sum_{i=1}^{C}\frac{TP_i}{TP_i+FP_i}$.
- Recall (Macro-Avg) = $\frac{1}{C}\sum_{i=1}^{C}\frac{TP_i}{TP_i+FN_i}$.
- F1 Score (Macro-Avg) = $\frac{1}{C}\sum_{i=1}^{C} 2\times\frac{\text{Precision}_i\times\text{Recall}_i}{\text{Precision}_i+\text{Recall}_i}$,

where C is the number of classes. TP_i, FP_i, and FN_i are the true positives, false positives, and false negatives for class i, respectively. These metrics are essential for understanding how well the model performs across all classes—especially in distinguishing between attack and normal traffic.

By combining macro-average precision, recall, and F1 score with runtime efficiency, our evaluation framework offers a holistic view of each model's strengths and limitations. This dual focus allows us to identify models that are both effective in detection and suitable for real-time deployment in 5G network infrastructures.

VI. Results and Discussion

The ML algorithms are deployed inside the load-balancer virtual machine, running with Ubuntu 20.04 LTS operating system, Intel 12-th Gen, 12500H, 8 cores CPU, and 8 GB RAM.

TABLE I: F1 macro average score on test-set with similar data patterns as Train Set

Model	F1 Score
Logistic Regression	0.991
Decision Tree	**0.999**
Random Forest	**0.999**
Linear SVC	0.991
Gradient Boosting	0.997
XGBoost	0.993
Gaussian NB	0.947

Table I presents the F1 scores achieved on the standard test-set. The models consistently reached F1 scores above 0.99, demonstrating exceptional detection capability. This high level of performance is largely attributable to the comprehensive set of features extracted from the network flows, which allowed the models to effectively distinguish between normal and malicious traffic patterns.

The model was subsequently evaluated on a second dataset comprising LDDoS attack data along with additional, previously unseen attack patterns. This experiment was specifically designed to assess the generalization capabilities of the detection algorithms—essentially testing how well the models perform when confronted with novel attack strategies that were not encountered during training.

Table II presents the comparative performance of the various models on this unseen dataset. Notably, traditional models such as Logistic Regression and Decision Trees exhibited a significant decline in performance, highlighting their limited ability to adapt to new attack patterns. In contrast, advanced ensemble methods like Gradient Boosting, as well as the Gaussian NB classifier, maintained high performance even on the unseen data. The Gradient Boosting approach, in particular, demonstrated robust generalization capabilities compared to a simpler ensemble method like Random Forest, which experienced a notable drop in performance under these conditions. The strong performance of Gaussian NB is attributed to its high generalization ability; although it achieved a slightly lower F1 score on the regular (seen) testing data, its performance remained stable and effective when applied to the unseen attack scenarios. These findings emphasize the importance of selecting models that not only excel on known datasets but also maintain resilience against evolving and unpredictable threat vectors.

TABLE II: F1 Scores on two test-sets

Model	F1 Macro Score	
	Seen Data	Unseen Data
Logistic Regression	0.991	0.803
Decision Tree	0.999	0.688
Random Forest	0.999	0.637
Linear SVC	0.991	0.809
Gradient Boosting	0.997	**0.927**
XGBoost	0.993	0.767
Gaussian NB	0.947	**0.931**

TABLE III: Comparison of Recall and Precision values

Model	Macro Score	
	Precision	Recall
Logistic Regression	0.731	0.949
Decision Tree	0.625	0.936
Random Forest	0.590	0.942
Linear SVC	0.737	0.952
Gradient Boosting	0.907	0.951
XGBoost	0.698	0.922
Gaussian NB	**0.990**	0.886

Further analysis of the results using precision and recall metrics, as presented in Table III, reveals an important trend across the evaluated models. Most models demonstrated a

higher recall compared to precision, indicating that while they were effective at detecting attack instances, they struggled to accurately identify benign (normal) traffic. This imbalance implies a tendency towards false positives, where normal data packets may be misclassified as malicious – an undesirable outcome in real-time traffic monitoring systems.

An exception to this trend was observed in the case of the Gaussian NB classifier. It achieved a balanced performance with both high recall and precision, suggesting that it can effectively detect attack data while minimizing the misclassification of legitimate traffic. This balance is crucial for practical deployment, as overly aggressive detection mechanisms that hinder normal traffic can degrade user experience and overall system performance. The consistent behavior of GNB across both seen and unseen datasets highlights its robustness and makes it a strong candidate for real-world DDoS detection in dynamic 5G environments.

In addition to classification performance, we also measure the average time taken to process each input record. This metric is critical for practical deployment in live networks. Detection models that introduce significant processing delays can become bottlenecks, rendering them unsuitable for real-time DDoS detection and consequently, DDoS mitigation. Table IV shows the average processing time for the different models. Some algorithms notably provided rapid and accurate detection, whereas the ensemble-based methods, while still achieving high F1 scores, incurred significantly higher processing times. This clearly illustrates the trade-off between detection accuracy and computational efficiency.

TABLE IV: Processing time for different models

Model	Processing Time (seconds)
Logistic Regression	0.010
Decision Tree	0.021
Random Forest	1.279
Linear SVC	0.011
Gradient Boosting	0.307
XGBoost	0.480
Gaussian NB	0.248

Together, these results highlight not only the robustness of our detection models under both standard and unseen conditions but also the importance of balancing speed and accuracy when considering real-time DDoS detection strategies in a 5G environment. Furthermore, once a flow/traffic is classified as malicious, it can be blocked/dropped using rule-based DDoS mitigation approaches [17].

VII. Conclusion

In this paper, we addressed an under-explored yet increasingly significant threat in modern 5G networks – DDoS attacks targeting the control plane via SCTP vulnerabilities. With the rapid growth of IoT and the proliferation of VNF-based 5G implementations, securing the control plane has become critical. We exploited inherent weaknesses in the SCTP protocol to design attacks on the Access and Mobility Function (AMF) through the N2 interface, and in doing so, we created the first-of-its-kind dataset specifically tailored for DDoS attacks on the 5G core. Our comprehensive evaluation framework goes beyond conventional accuracy metrics, incorporating macro-average precision, recall, F1 scores, and, importantly, the detection time per record – an aspect often neglected in previous studies but crucial for real-time deployment. Through our experiments, we highlighted the trade-off between the generalization power of various machine learning models and the associated computational overhead required for rapid detection. Moreover, our investigation into low-rate DDoS (LDDoS) attacks revealed the additional challenges they pose to traditional detection mechanisms.

References

[1] "5G programmable infrastructure converging disaggregated network and compute resources," 5GPPP, Tech. Rep., Jan. 2018.

[2] S. Park, B. Cho, D. Kim, and I. You, "Machine Learning Based Signaling DDoS Detection System for 5G Stand Alone Core Network," *Applied Sciences*, vol. 12, no. 23, p. 12456, 2022.

[3] J. Park, J. Kim, S. Woo, K. Park, J. Kim, and J.-H. Lee, "Experimental Approach to Internal Security Threats for 5G-Advanced Core Networks," in *Proc. of IEEE PIMRC*, 2024, pp. 1–6.

[4] E. P. Rathgeb, C. Hohendorf, and M. Nordhoff, "On the Robustness of SCTP against DoS Attacks," in *Proc. of Intl. Conf. on Convergence and Hybrid Information Technology*, 2008, pp. 1144–1149.

[5] T. E. Ali, Y.-W. Chong, and S. Manickam, "Machine Learning Techniques to Detect a DDoS Attack in SDN: A Systematic Review," *Applied Sciences*, vol. 13, no. 5, p. 3183, 2023.

[6] Ismail, M. I. Mohmand, H. Hussain, A. A. Khan, U. Ullah, M. Zakarya, A. Ahmed, M. Raza, I. U. Rahman, and M. Haleem, "A Machine Learning-Based Classification and Prediction Technique for DDoS Attacks," *IEEE Access*, vol. 10, pp. 21 443–21 454, 2022.

[7] A. A. Alashhab, M. S. M. Zahid, M. A. Azim, M. Y. Daha, B. Isyaku, and S. Ali, "A Survey of Low Rate DDoS Detection Techniques Based on Machine Learning in Software-Defined Networks," *Symmetry*, vol. 14, p. 1563, 2022.

[8] M. S. Raza, M. N. A. Sheikh, I.-S. Hwang, and M. S. Ab-Rahman, "Feature-Selection-Based DDoS Attack Detection Using AI Algorithms," *Telecom*, vol. 5, no. 2, pp. 333–346, 2024.

[9] M. R. Dey, P. Nithiyasri, and M. Patra, "Early Detection of DoS Attacks in 5G Core Networks," in *Proc. of IEEE ANTS*, 2024, pp. 1–6.

[10] W. I. Syafril, B. Arifwidodo, and D. Pranindito, "Analysis Of Intrusion Prevention System (IPS) On Software Defined Network (SDN) In Preventing Distributed Denial of Service (DDoS) Attacks," in *Proc. of IEEE Intl. Conf. on Communication, Networks and Satellite*, Nov. 2024, pp. 759–765.

[11] W. Dev, S. Bera, and A. Tiwari, "Control-Plane Load Balancing and Auto-Scaling in 5G and Beyond Networks," in *Proc. of IEEE ICOIN*, Chiang Mai, Thailand, 2025.

[12] J. Ginesin, M. von Hippel, E. Defloor, C. Nita-Rotaru, and M. Tüxen, "A formal analysis of SCTP: Attack synthesis and patch verification," in *Proc. of the USENIX Conference on Security Symposium*, ser. SEC '24, USA, 2024, pp. 3099–3116.

[13] S. S. Samarakoon, Y. S. Siriwardhana, P. P. Porambage, M. L. Liyanage, S.-Y. C. Chang, J. K. Kim, J. K. Kim, and M. Y. Ylianttila, "5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network."

[14] "IDS 2017 | Datasets | Research | Canadian Institute for Cybersecurity | UNB."

[15] S. M. S. Subramanian, M. R. K. Kanagarathinam, and K. M. S. Sivalingam, "Packet Traces from Large Scale 5G Performance Testing," IEEE Dataport, 2024.

[16] A. Tiwari and S. Bera, "Dataset of handshake flooding on 5g control plane," IEEE Dataport, 2025.

[17] Devzery, "Guide to Suricata: Network Security, IDS, IPS, and NSM," 2023.