A Study on Programmable Switch Enabled Network Defense Methods Against Volumetric Attacks

Dong Hyeok Kim*
kimdh98@kaist.ac.kr
KAIST

Yaojun Huang
hyjun@kaist.ac.kr
KAIST

Abstract

Volumetric attacks pose serious risks to modern computer networks, and efficiently mitigating them remain an active challenge to this day. Advancements in programmable switches, high-level network control languages and customizable data plane logic, has enabled new methods in rapid detection and mitigation of attack traffic. This paper surveys recent works leveraging programmable switches while addressing key challenges such as limited memory (Elastic sketch), high-throughput requirements (Euclid), and attack diversity (Patronum).

CCS Concepts

• **Networks → Programmable networks;Security protocols**.

1 Introduction

Volumetric attacks, such as distributed denial of service (DDoS) attacks, continue to plague modern computer networks, due to its highly versatile and disruptive nature to online services. These attacks flood targeted networks with overwhelming amounts of data often masquerading as benign requests, rendering critical services inaccessible to legitimate users. As network infrastructures grow in complexity, due to the growing reliance on massive data centers, the scale of volumetric attacks also increases, requiring increasingly more efficient defense methods to keep up.

One key advancement in networking technology is the programmable switch, which unlike fixed-function traditional switches allow data-plane programmability at line-rate processing speeds, proving to be a valuable asset in designing new in-network defenses. This capability is particularly valuable for mitigating volumetric attacks, as it enables real-time detection and response, reducing the latency and overhead associated with conventional, centralized defense mechanisms [1].

In this paper, we present a study of recent approaches in designing efficient network defense methods against volumetric attacks using programmable switches. The goal of this study is to analyze the key challenges research questions in handling volumetric attacks at high-throughput network links, and the methods suggested in recent works to solve these questions.

2 Background

2.1 Volumetric attacks

Volumetric attacks can be classified into two types, heavy hitters and DDoS attacks [2]. Heavy hitters refer to particular flows in a network link that send massive amount of data compared to other typical flows, consuming a large amount of bandwidth. If such heavy hitter flows are left unchecked, they can easily clog up network links, affecting service delays or routing decisions. On the other hand, DDoS attacks typically involve an attacker controlling a large number of network nodes to simultaneously funnel a large amount of data to a single destination. These two types of attacks are not necessarily mutually exclusive, as some DDoS attacks such as UDP flooding involve multiple sources acting has heavy hitters to perform the attack.

2.2 Programmable Switches and P4

Through the combination of the novel Protocol Independent Switch Architecture (PISA) and Programming Protocol-independent Packet Processors (P4), programmable switches are able to maintain line-rate processing speeds of fixed-function switches, while providing the flexibility of software-defined processors such as Openflow switches and DPDK. This is largely enabled by the introduction of P4, a high-level language designed specifically for programming the data plane of network devices [1]. P4's design allows operators to update packet processing logic in real time, adapting to evolving network conditions and security requirements. While these features make programmable switches well-suited for addressing network security problems, programmable switches have limited hardware memory, which makes handling large flows a non-trivial task.

3 Survey on Existing Techniques

This section presents works utilizing probabilistic data structures and several notable frameworks that demonstrate the capabilities of programmable switches in volumetric attack mitigation while circumventing the memory limitation.

3.1 Count-min Sketch and Elastic Sketch

Heavy hitter detection requires keeping track of packet counts for every flow, which is infeasible under the memory constraints of programmable switches. To overcome this limitation, probabilistic data structures (PDS) [6], which use fixed-size memory to provide approximations of collected data, are often employed. One of the most fundamental PDS used for heavy-hitter detection is the count-min sketch (CMS) [3], which serves as the basis of many advanced mitigation methods.

CMS utilizes a fixed number of hash tables, each corresponding to a different hash function. The table update procedure uses the hashes the flow identifier (e.g. 5 tuple) of incoming packet by each hash function to increment the counter in the corresponding entry of each hash table. The approximate packet count of each flow is given by the minimum of the corresponding entries in each hash table. While overestimation is inevitable due to hash collisions, CMS

*Both authors contributed equally to this research.

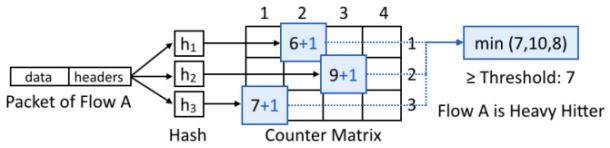

Figure 1: The count-min sketch [2].

provides a good trade-off between accuracy and memory usage for heavy hitter detection.

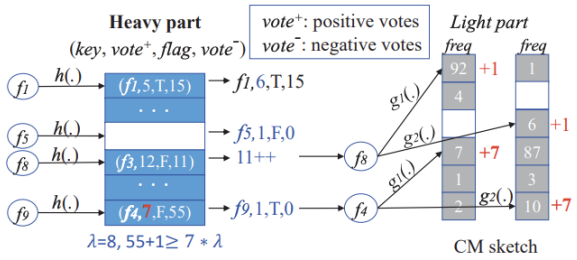

Figure 2: Elastic Sketch overview [8]

However, as heavy-hitter flows are scarce compared to regular flows, one downside of CMS is that most of the entries are occupied by uninteresting non-heavy-hitter flows. Elastic sketch [8] improves on this through adding a hierarchical structure to flow counting. It combines two components: a heavy part, which accurately tracks large flows using a small hash table, and a light part, which approximates smaller flows using a count-min sketch. By dynamically adjusting the allocation of resources between these two parts, Elastic Sketch achieves both high accuracy for large flows and efficient processing of smaller flows.

3.2 Jaqen [5]: A Switch-Native Approach for Detecting and Mitigating Volumetric Attacks

Jaqen is a high-performance defense solution designed to perform inline detection and mitigation on volumetric DDoS attacks directly within programmable network switches, leveraging the capabilities of P4-based hardware.

The design of Jaqen focuses on three main steps, as shown in Figure 3. In the broad-spectrum in-band detection phase, the switch layer uses an optimized universal sketch that aggregates multiple metrics of the traffic to continuously monitor and detect different attack types. These counters are reported to the control layer, where the controller measures these metrics with various criteria for different types of attacks and conducts detections. Based on the detection result, the network-wide resource manager of Jaqen dynamically allocates resources across the network for mitigation. A mixed-integer program (MIP) is used to optimize resource allocation across ISP switches to minimize possible hardware resources, and a heuristic algorithm is applied for better responsiveness. For mitigation, Jaqen provides a flexible API that allows for various

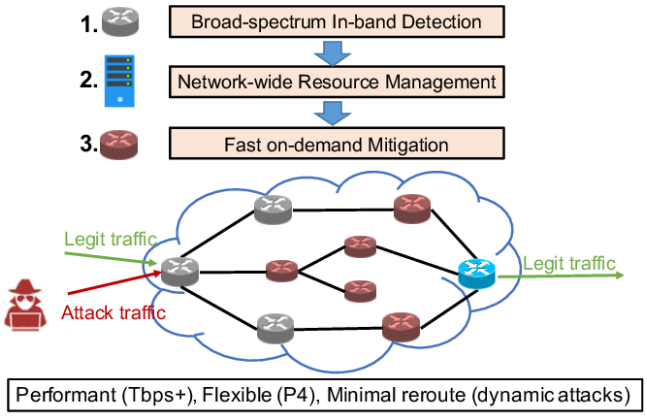

Figure 3: The three steps of Jaqen.

mitigation strategies, including filtering and rate limiting, to be implemented directly on the switch hardware with switch-optimized components like bloom filters.

Jaqen's architecture achieves high line-rate performance while maintaining low false positives/negatives in attack mitigation. It handles multiple attack types like TCP/UDP/ICMP-based attacks and Application-layer attacks by combining sketch-based detection and switch-level mitigation. However, Jaqen requires frequent interaction between the control plane and the data plane, which makes it less responsive on defense efficiency.

3.3 Euclid [4]: Detection and Mitigation Fully in Data Plane

To address the issue of Jaqen, Euclid offloads both the detection and mitigation algorithms to the data plane to further improve defense efficiency.

Euclid is an entropy-based DDoS detection mechanism designed for deployment within autonomous systems near the victim, ideally at border routers. During a DDoS attack, the entropy of the source IP is expected to increase due to the introduction of new values in the attack packet, while the entropy of the destination IP is expected to decrease due to the increasing frequency of the victim being the destination.

In light of this observation, Euclid has put forward a framework with detection and mitigation mechanisms fully integrated into the data plane. An overview of Euclid is depicted in Figure 4.

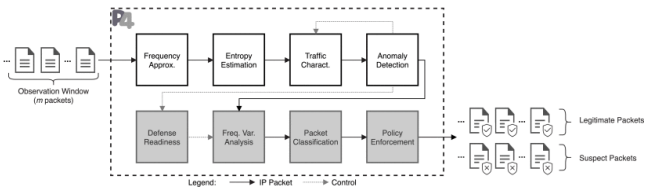

Figure 4: An overview of Euclid.

Euclid initially divides incoming packets into fixed-size observation windows and computes the frequencies of the source and destination IPs using Count-sketches. Following the collection of

frequency approximation, Euclid estimates the entropy of the set of IP addresses using the equations:

$$S(X) = \sum_{x=1}^{N} f_x \log_2 (f_x) .$$

$$H(X) = \log_2(m) - \frac{1}{m} S(X)$$

where m represents the number of packets in the observation window, N denotes the number of distinct addresses in address set X, and f_x is the frequency approximation of address x. To overcome the challenge of lacking operations (i.e. multiplication, logarithmic, and loop) in programmable switches when computing the equations, Euclid adopts the incremental updates mechanism by adding the difference of the new term and the old term to S. Additionally, it defines a pre-computed function and utilizes the longest prefix match lookup table to retrieve the final function values.

Following the estimation of the entropy of the IP address, the system compares the entropy measurement with the exponentially weighted moving averages (EWMAs) and the mean deviations (EWMMDs) measured with previous traffic to check whether the traffic is in an anomalous state. If the condition is not met, an alarm is triggered and the system enters a defensive state. In this state, Euclid calculates the frequency variation for each packet and uses this metric to classify the packet as legitimate or suspect. Once the packet has been classified, Euclid applies security policies, including discarding, throttling, and diverting, to the suspect packets.

Euclid has demonstrated its high effectiveness through software simulation. By appropriately selecting the sensitivity coefficient, defense threshold, and other parameters, Euclid achieves an excellent true positive rate in detecting malicious traffic while maintaining minimal impact on legitimate packets with a short response time.

3.4 Patronum [7]: A More Comprehensive Volumetric DDoS attacks Detection Framework

Euclid is effective in detecting and mitigating many-to-few (M2F) DDoS attacks within the data plane, but it struggles to identify few-to-few (F2F) attacks. In an F2F attack scenario, the entropy of source and destination IP addresses exhibits a similar distribution, making it difficult for the entropy-based method to recognize such malicious traffic.

To address this vulnerability, Patronum utilizes two mechanisms to detect these two types of attacks with two separate modules. Figure 5 illustrates the system architecture of Patronum, which comprises three components: a high-frequency periodic in-network measurement (HFPIM) module, an entropy difference-based detection module, and an in-network bandwidth monitor.

The HFPIM module performs network measurement using a time window and a Count-Min sketch, similar to previous approaches. The entropy difference-based detection module is dedicated to detecting M2F attacks and measures the biased distribution of source and destination IP by computing the entropy difference metric (EDM) of these two sets of addresses. Like Euclid, it employs a look-up table approach to calculate the entropy difference but enhances it by rounding to reduce the stored entries, thereby minimizing the

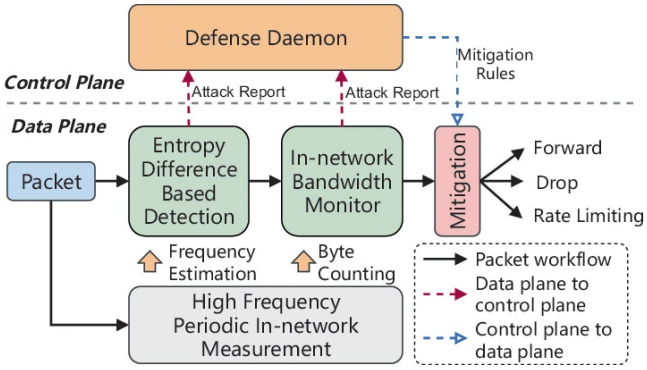

Figure 5: The system architecture of Patronum.

memory footprint. The in-network bandwidth monitor is responsible for identifying F2F attacks. It measures the packet-sending rate of a source IP address using the formula:

$$Bandwidth_{src} = \frac{Bytes_{src}}{Time}$$

where $Bytes_{src}$ is collected by Count-Min sketch and Time represents the time window size. If the sending rate exceeds a threshold, a potential attack is detected and rate limiting is enforced. The system also sends alert messages to the control plane.

The control plane periodically gathers statistical data from the data plane and calculates the gradient of speed variation to identify an attack. Upon detecting an attack, the control plane uses the information received from the data plane to filter out the attack traffic and implement a security policy on the data plane to mitigate the attack.

Patronum has been implemented on Intel Tofino, and experiments have demonstrated that Patronum has minimal resource requirements while surpassing AccTurbo and Jaqen in detecting both M2F and F2F attacks.

4 Conclusion

Through the analysis multiple representative works of utilizing programmable switches against volumetric attacks, the following observations are made:

- Programmable switches have great potential in detecting and mitigating DDoS attacks with a short response time.
- Probabilistic data structures achieve high accuracy while having a low memory footprint.
- The limitation of syntactic expressiveness in current programmable switches requires to carefully design the algorithm to achieve feasibility and performance.
- Offloading some decisions to control plane is required for mitigation robust against various attack types.

References

[1] Gibb G Izzard M McKeown N Rexford J Schlesinger C Talayco D Vahdat A Varghese G Walker D. Bosshart P, Daly D. 2014. P4: Programming protocol-independent packet processors. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 87–95.

[2] Chunming Wu Xuan Liu Qun Huang Dong Zhang Haifeng Zhou Qiang Yang Chen, Xiang and Muhammad Khurram Khan. 2023. Empowering network security

with programmable switches: A comprehensive survey. *IEEE Communications Surveys Tutorials* 25, 3 (2023), 1653–1704.

[3] Graham Cormode and Shan Muthukrishnan. 2005. An improved data stream summary: the count-min sketch and its applications. *Journal of Algorithms* 55, 1 (2005), 58–75.

[4] Alexandre da Silveira Ilha, Ângelo Cardoso Lapolli, Jonatas Adilson Marques, and Luciano Paschoal Gaspary. 2021. Euclid: A Fully In-Network, P4-Based Approach for Real-Time DDoS Attack Detection and Mitigation. *IEEE Transactions on Network and Service Management* 18, 3 (2021), 3121–3139. https://doi.org/10.1109/TNSM.2020.3048265

[5] Zaoxing Liu, Hun Namkung, Georgios Nikolaidis, Jeongkeun Lee, Changhoon Kim, Xin Jin, Vladimir Braverman, Minlan Yu, and Vyas Sekar. 2021. Jaqen: A High-Performance Switch-Native Approach for Detecting and Mitigating Volumetric DDoS Attacks with Programmable Switches. In *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, 3829–3846. https://www.usenix.org/conference/usenixsecurity21/presentation/liu-zaoxing

[6] Sahil Garg Ravneet Kaur Shalini Batra Neeraj Kumar Singh, Amritpal and Albert Y. Zomaya. 2020. Probabilistic data structures for big data analytics: A comprehensive review. *Knowledge-Based Systems* 188 (2020).

[7] Jiahao Wu, Heng Pan, Penglai Cui, Yiwen Huang, Jianer Zhou, Peng He, Yanbiao Li, Zhenyu Li, and Gaogang Xie. 2024. Patronum: In-network Volumetric DDoS Detection and Mitigation with Programmable Switches. In *Computer Security – ESORICS 2024*, Joaquin Garcia-Alfaro, Rafał Kozik, Michał Choraś, and Sokratis Katsikas (Eds.). Springer Nature Switzerland, Cham, 187–207.

[8] Jie Jiang Peng Liu Qun Huang Junzhi Gong Yang Zhou Rui Miao Xiaoming Li Yang, Tong and Steve Uhlig. 2018. Elastic sketch: Adaptive and fast network-wide measurements. *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication* (2018), 561–575.