2021 17th International Conference on Mobility, Sensing and Networking (MSN) | 978-1-6654-0668-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/MSN53354.2021.00076

TAP: A Traffic-Aware Probabilistic Packet Marking for Collaborative DDoS Mitigation

Mingxing Liu*‡, Ying Liu*‡, Ke Xu†‡§, Lin He*‡, Xiaoliang Wang†, Yangfei Guo*, Weiyu Jiang¶
*Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing, China
†Department of Computer Science and Technology, Tsinghua University, Beijing, China
‡Beijing National Research Center for Information Science and Technology (BNRist)
§Peng Cheng Laboratory (PCL), Shenzhen, China
¶Huawei Technologies, Beijing, China

***Abstract*—In recent years, Distributed Denial-of-Service (DDoS) attacks have become more rampant and continue to be one of the most serious security threats facing network infrastructure. In a classic DDoS attack, the attacker controls numerous bots from many sources to send a significant volume of traffic to flood the victim end or the bottleneck link. In practical networks, it is inefficient and costly to request all partner routers to collaboratively mitigate DDoS attacks. The common feature of DDoS attacks is the abnormal distribution of traffic to the victim. In this paper, we propose TAP, a collaborative DDoS mitigation framework, based on traffic-aware probabilistic packet marking (PPM). TAP enables the victim to select a few hit routers as collaborators to mitigate attack traffic efficiently depending on the traffic distribution. Our evaluation results show that TAP greatly reduces attack traffic within seconds and mitigate the damage caused by DDoS with less overhead, which demonstrates that TAP is an effective, efficient, and rapid-response scheme for collaborative DDoS mitigation.**

***Index Terms*—collaborative DDoS mitigation, traffic awareness, IP traceback, probabilistic packet marking**

I. Introduction

In the past few decades, Distributed Denial-of-Service (DDoS) attacks continue to plague network infrastructure availability. Recent years have witnessed that "DDoS as a service", as D.Makrush described in [1], has made DDoS easier for attackers to access. DDoS is still a growing and severe threat to high-throughput networks, *i.e.*, Internet Service Providers (ISPs), data centers and cloud services. In February 2020, AWS shield service observes a CLDAP reflection attack with a peak volume of 2.3 Tbps, which is the most enormous DDoS attack traffic recorded so far [2]. *Tbps* has become the new common unit of attack traffic peaks.

In order to cope with such a large volume of traffic attacks, the research community has proposed several mitigation schemes [3]–[20]. These schemes are roughly divided into two categories: single-entity-based defense and collaborative defense. Single-entity-based defense [3]–[7] approaches place high resource requirements on the entity, and network latency and identification accuracy are often unsatisfactory. Subject to limited resources, victims require collaborative DDoS mitigation against a large volume of attack traffic. Depending on the collaborator selected, there are three collaborative strategies: intermediate network-based [8]–[11], source-based [13]–[15], and hybrid collaboration [17]–[20]. Both source-based and network-based collaborations are effective in mitigating DDoS damage, with typically more deployment overhead and collaboration cost due to the large number of collaborators. Hybrid collaboration advocates choosing the optimal non-specific collaborators, but the selection based on DDoS feature detection introduces long response time, typically minutes or even hours.

Through in-depth analysis of DDoS traffic distribution, we have two key observations. *One is that the essential feature of DDoS is the abnormal distribution of traffic routed to the victim in the network.* DDoS is diverse and has different attack features, but its common goal is the victim's inability to respond to legitimate requests. According to the attack methodology, the attacker either blocks the victim's legitimate traffic in the network or sends junk traffic to consume the victim's resources. *Another is that the traffic distribution is uneven, with a small number of hit routers forwarding the majority of the attack traffic.* The closer to the victim, the more pronounced this distribution becomes, which allows the victim to select some hit routers with sufficient resources as collaborators to filter DDoS traffic. By selecting these hit routers as collaborators, the defense can greatly mitigate DDoS damage with less overhead and lower latency.

Therefore, to realize efficient and rapid-response collaborative DDoS mitigation, we argue that the victim should select some hit routers which forward most attack traffic and own available resources, which requires the victim's ability to sense the distribution of traffic routed to itself across the network. In practice, the separation of traffic awareness and collaborative mitigation results in manual troubleshooting and long response time, often minutes or even hours.

In this paper, we propose TAP, which is a collaborative DDoS mitigation framework based on IP traceback and traffic awareness. When facing DDoS, the upstream routers may drop most of the traffic before reaching the victim [21]. Trading off the scheme's overhead and robustness, we design a novel

This work was in part supported by the National Key R&D Program of China with No. 2018YFB1800405 and No. 2018YFB0803405, the China National Funds for Distinguished Young Scientists with No. 61825204, the NSFC Project with No. 61932016 and No. 61772307, the Beijing Outstanding Young Scientist Program with No. BJJWZYJH01201910003011, the Beijing National Research Center for Information Science and Technology (BNRist) with No. BNR2019RC01011, and the PCL Future Greater-Bay Area Network Facilities for Largescale Experiments and Applications with No. LZC0019. Ying Liu is the corresponding author.

978-1-6654-0668-0/21/$31.00 ©2021 IEEE
DOI 10.1109/MSN53354.2021.00076

traffic-aware probabilistic packet marking (TAPPM) for IP traceback and traffic awareness. Using TAPPM, the victim can discover and locate hit routers. The victim selects the hit routers on the attack path as collaborators to limit the forwarding rates to achieve effective and rapid-response mitigation against DDoS. Compared with state-of-the-art TDFA [15], TAP reduces the collaboration overhead by almost 80% , and fully initiates collaborative mitigation within seconds.

We make the following contributions:

- We propose a novel TAPPM, which marks packets with tags carrying the traffic semantics and path information, so that the destination can perceive the distribution of the traffic routed to itself in the network.
- We propose DDoS traffic discovery based on traffic volume changes. This method allows the victim to trace the source of most attack traffic and locate the attack point in seconds.
- We propose a collaborative DDoS mitigation framework based on traffic awareness and IP traceback, enabling the victim to locate attack points and select a few hit routers as collaborators within seconds.
- Extensive simulation experiments are conducted. We compare the performance of TAP with similar works , which validates that TAP is effective, efficient, and rapid-response in reducing the attack traffic and allowing the legit traffic.

The remainder of this paper is organized as follows: Section II reviews related work on Probabilistic Packet Marking (PPM) and collaborative mitigation along with our motivation. Section III provides a high-level overview of TAP. In Section IV , we introduce the design details of traffic-aware PPM and collaborative mitigation. Section V presents the experimental evaluation results of TAP. Finally, we conclude the work in Section VI.

II. Motivation and Related Work

In this section, we first introduce our motivation and the adversary model, and then briefly review related work on PPM and collaborative defense.

A. Motivation

DDoS attackers always attempt to send much malicious traffic to block the victim or the network, resulting in a dramatic increase or decrease in traffic to the victim. An attacker may employ a variety of tactics to hide the malicious traffic, but cannot hide the volume of the traffic routed to the destination. If the victim finds too much or too little traffic routed to it, the traffic distribution indicates that there is a high probability of DDoS.

There are still two challenge in collaborative mitigation against DDoS. Firstly, the traffic distribution is uneven in the network, and a few hit routers forward most of the attack traffic so that it is inefficient and costly request all collaborators to activate defense. Thus, the most crucial challenge is how to quickly perceive the distribution of the attack traffic. Another challenge is that DDoS traffic is mostly based on IP spoofing to hide true attack traffic distribution.

To overcome the above challenges, we propose TAP based on traffic-aware PPM, which enables the victim to reconstruct the attack path and perceive the distribution of the traffic routed to itself in the network. This work is motivated by the victim's ignorance of the traffic distribution and the lack of efficient collaboration between victims and network entities, leading to unaffordable cost of DDoS detection and defense.

B. Adversary Model

The adversary initiates DDoS by controlling lots of botnet. In general, there are two categories of attack that should be considered when the victim fails to respond to the legitimate request.

DDoS against victim ends. The adversary controls lots of botnets to directly send a large volume of junk traffic to exhaust the victim's bandwidth resources. Exploiting host vulnerability to bypass the security mechanism, DDoS greatly amplifies the volume of traffic routed to the victim, aiming at flooding the victim. The victim does not have the ability to filter all DDoS traffic individually.

DDoS against bottleneck links. The adversary sends the attack traffic to congest the bottleneck link. The attack traffic may not directly target the victim but consumes the bottleneck link's bandwidth to force it to discard legitimate packets. Since the router does not notify the destination of packet loss and the attack traffic may not use the victim IP as the destination IP, it is difficult for the victim to locate the attack point and be aware of the attack traffic.

Compromised routers may generate attack traffic, *e.g.*, traffic hijacking, traffic injection, load imbalance. A malicious router may compromise and modify the packets passing through it, confusing the traffic awareness at the victim. TAP needs to prevent tags from tampering and forgery by encrypting.

C. Related Work

PPM: The classical PPM was first proposed by Savage *et al.* in 2000 [22], [23]. Subsequently, Authenticated Packet Marking and Adjusted Packet Marking are also proposed to prevent the attacker from forging information and address the "weakest chain" problem [24], [25], [26], [27]. PPM effectively reduces additional communication overhead to perform large-scale traceback and packet identification, but requires more packets with tags to reconstruct the complete forwarding path [14], [28], [29]. Samant Saurabh *et al.* [29] improve the effectiveness of packet marking. Bo Wu *et al.* [27] propose Probabilistic Path Verification to save communication overhead and verify the source and the forwarding path of packets. In [30], Mohammed M. Kadhum *et al.* redesign the packet probability drop function for "Fast Congestion Notification", which inspires how TAP limits the attack traffic. However, these PPM schemes can not accurately deliver the information about destination-specific traffic to the destination.

Collaborative Mitigation: The collaborative defense is efficient and attractive to defend against DDoS attacks for

capacity-limited victims. [31] and [32] have proved the feasibility of providing collaborative DDoS mitigation services for downstream customers for in current and future networks. For some potential attacks, some collaboration methods [33], [34] based on machine learning have been proposed and the deployment requires high capabilities of the operators. [19], [35], [20] present some collaboration schemes based on the novel underlying network architecture. They design particular protocols to transmit attacker information or assist in cleaning the attack traffic. However, they do not explain how to obtain attacker information to realize efficient collaborative defense effectively. Collaboration based on IP traceback can effectively prevent attackers from forging addresses and improve defense efficiency against DDoS. Vahid Aghaei Foroushani *et al.* [15] propose TDFA, a framework for defending against DDoS flooding attacks based on deterministic packet marking of edge routers. The victim can not perceive the distribution of the traffic to itself in the network.

III. Design Overview

In this section, we present a high-level overview of the proposed TAP based on traffic-aware PPM.

A. Assumption

This work aims to build an efficient, effective and rapid-response collaborative DDoS mitigation system. The effectiveness depends on a few assumptions about the underlying network and entity.

- *We assume that each TAP-enabled entity has an identifier.* The identifier can be IPv4 address or Autonomous System Number (ASN), depending on whether TAP design is at the router- or AS-level granularity. A custom shorter identifier is more bandwidth-efficient.
- *The entity has a secret symmetric key with the victim using existing key exchange protocols* [36]. They can also use public keys to identify each other and generate a symmetric key regularly.
- *We assume that the entity knows the IP prefixes of the victim.* Border entities (*e.g.*, border routers and gateways) build a secure entry that is not aggregated with other addresses for the protected IP.
- *For each specific tag, the victim can receive at least one of its replicas.* TAP using PPM enhances the robustness by sending multiple tag replicas and ensures that the victim gets the traffic semantics and path information in each tag.

B. Architecture

As shown in Figure 1, TAP consists of two processes: traffic-aware PPM and collaborative mitigation. In the first process (§ IV-A), the TAP-enabled router generates the tag, according to the traffic volume routed to the victim. Every time a particular volume of traffic is successfully forwarded, a traffic-aware tag is generated to probabilistically mark the packet. The victim collects the tag to calculate the volume of traffic routed to itself by the router within the tag's validity

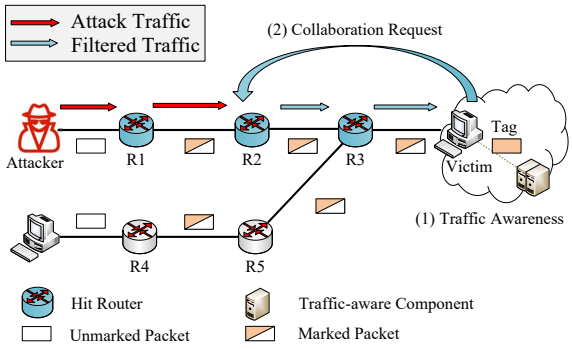


Fig. 1. The high-level overview of TAP.

period, which is the key to performing traffic awareness in the network. Furthermore, by comparing the distribution of traffic in different periods, the paths of most attack traffic are found as well as the hit routers.

In the second process (§ IV-E), TAP allows the victim to request efficient DDoS protection from hit routers directly. The routers, $R1$, $R2$ and $R3$, can be selected as hit routers in Figure 1. Considering the trade-off between overhead and revenue of collaboration, the victim selects $R2$ as the collaborator to mitigate attack traffic. Upon receipt of the request, $R2$ immediately cooperates to constrain the attack traffic and protect the legitimate traffic of the Vicitm.

IV. Traffic-Aware PPM

In this section, we describe the design of TAP in detail.

A. Tag Generation

In traffic-aware PPM, the tag contains two types of information. On the one hand, it carries partial path information for the destination to perform IP traceback to determine the true source or path of attack traffic. On the other hand, the information of the traffic volume forwarded to the destination is also embedded in the tag by the router. The destination can perceive the traffic rate on the routers and infer attack traffic distribution.

*1) **Routing Entry**:* To generate the tag implying traffic volume, as shown in Figure 2, the router maintains a secure entry for the IP prefix of the protected destination. The IP prefix exactly belongs to one ASN or protected destination while not preventing the router automatic routing. Besides, the router adds a loop counter and a timestamp to the routing entry. The loop counter is denoted as *Counter*. A counting cycle is denoted as $\mathcal{N}$ that is notified to the destination. The timestamp with a budget of 4 bytes records the time of generating the current tag. When *Counter* is greater than $\mathcal{N}$, the router updates the timestamp to the current time and regenerates new tag.

*2) **Generating Tags**:* Figure 3 shows the two steps of marking a packet with the tag. In the first step, the forwarding entity assembles the link information composed of *localR*, *nextR* and the traffic information implied by the timestamp

dst/mask	nextR	Counter	timestamp	key
20.0.0.0/24	R2	n	HH:MM:SS	K*i*

Fig. 2. The routing entry of the protected destination on the router. $nextR$ is the identifier of next hop router. key is the shared key with the destination.

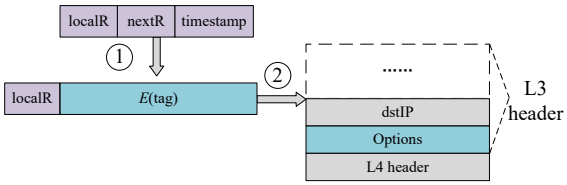


Fig. 3. The two steps generate traffic-aware tags to mark packets.

into the original tag denoted as *originalTag*. *localR* and *nextR* are the identifiers of the local router that marking the packet and the next hop router respectively, with a budget of 4 bytes. If customized, they will be shorter. *localR* and *nextR* together determine a link on the path for IP traceback.

In the second step, *localR* and *E(tag)* together form the final tag, denoted as *finalTag*, which the router embeds in the extension of the IP header. *E(tag)* stands for the operation of encrypting the original tag with the symmetric key to prevent the tag from being tampered or forged. The outer layer *localR* in *finalTag* indicates the identity of the router that added the tag to the packet so that the destination can index the corresponding decryption key.

As described in Algorithm 1, when a packet with the corresponding destination IP is successfully forwarded, *Counter* is incremented by one until it is equal to $\mathcal{N}$. If $\mathcal{N}$ is the number of bytes, the router increases *Counter* by the packet's size.

Each router has its own $\mathcal{N}$ notified to the destination in advance, and the parameter remains stable for a long period, for example, an hour. $\mathcal{N}$ is the number of packets the router expects to forward to the destination within the lifetime of a tag. When there are multiple links to the destination, the router has a $\mathcal{N}$ for each neighboring link.

In terms of the collaboration overhead, $\mathcal{N}$ determines the frequency of the label. The smaller $\mathcal{N}$, the more frequently the timestamp and label are updated, increasing computational overhead of routers. λ and the probability of marking p determine the average spacing of tagged packets. The smaller λ, the more packets are marked, resulting in more bandwidth overhead for the attached tags.

B. Packet Marking

Tags are added to the extension headers of the IP header with a probability. On high-speed forwarding routers, a port's packet forwarding rate may reach millions packets per second (*pps*). The router generates a random number for each packet and millions random number per second. It brings high latency and computational overhead to the router.

Algorithm 1 Tag Generation Algorithm

Input:
$\mathcal{N}$: the number of packets successfully forwarded by the router between different tags
key: the symmetric key of the router and the destination
localR: the identifier of the local router
nextR: the identifier of the next hop router
$originalTag$: the original tag without encapsulation
Output:
$finalTag$: the tag that will be probabilistically added to the next $\mathcal{N}$ packets

Counter$\leftarrow 0$
timestamp$\leftarrow$ the current time
$originalTag \leftarrow [localR, nextR, timestamp]$
for each *packet successfully forwarded* **do**
  Counter $=$ *Counter* $+ 1$
  if Counter $>= \mathcal{N}$ **then**
    Update $timestamp$ to the current time
    Update $originalTag$
    Counter = 0
  end if
end for
$E_{key}(originalTag)$: Encrypt $originalTag$ with *key*
$finalTag = [localR, E_{key}(originalTag)]$
return $finalTag$;

Algorithm 2 Packet Marking Algorithm

The router generates a random number λ
if Counter$\%\lambda == 0$ **then**
  Mark the packet
  Forward the packet to the next hop
  Counter=*Counter*+1
else
  Directly forward the packet to the next hop
  Counter=*Counter*+1
end if

1) Random Interval: Empirical analysis shows that reducing the number of random numbers can greatly improve the performance of marking. In order to improve the efficiency of marking, the router does not generate a random number for each packet, but a random interval λ in Algorithm 2. λ obeys a random distribution with the mean of the reciprocal of the probability p, which reduces the generation of random numbers by more than 90%.

C. IP Traceback

We argue that the tag is beneficial to not only reconstruct the forwarding path, but also imply the distribution of IP addresses in cyberspace. When DDoS occurs, the attack IP should be quickly identified and blocked. However, large-scale IP traceback is still a task with high storage load for the destination. In a DDoS scenario, the IP range of DDoS is so extensive that IP traceback may be a DDoS vulnerability for the destination. To perform efficient and periodic IP traceback,

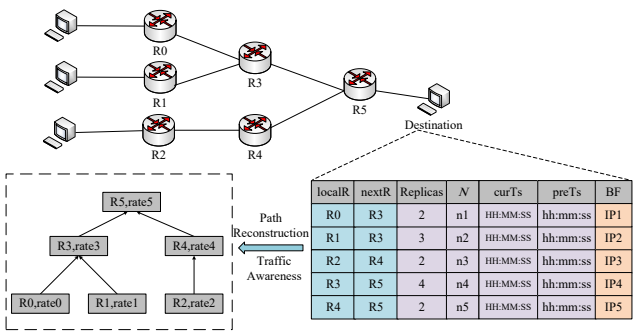


Fig. 4. Path reconstruction and traffic awareness. The current timestamp and previous timestamp are denoted as *curTs* and *preTs*, respectively. *Replicas* represents the number of replicas of the latest tag. **BF** represents the Bloom Filter that stores the source IP on the link. The destination leverages this information in the purple area to calculate the rate of traffic to itself on each router and in the orange area to reconstruct the forwarding path.

the destination builds an IP set for each link rather than a link set for each flow.

We use space-efficient and link-specific Bloom filters [37] to record IPs that appear in link, because the link-specific tag tells the destination on which link the packet may appear. IP traceback is transformed into the process of traversing whether IP is in Bloom filters. As shown in Figure 4, the destination maintains a table of link. The set of all links is denoted as L. $path_{ip}$ represents the complete forwarding path of ip. It is a subset of L, denoted as $path_{ip} = link_1, link_2, \cdots, link_d$. The Bloom filter corresponding to the link in $path_{ip}$ is positive for ip. The destination reconstructs the complete path by traversing whether or not ip is in Bloom filters.

D. Traffic Awareness

Traffic awareness is the key to achieving efficient collaborative DDoS mitigation. To perceive the traffic distribution, as shown in Figure 4, the destination records the latest two timestamps of each link. The difference between the current timestamp *curTs* and the previous timestamp *preTs* is the time for the router to forward $\mathcal{N}$ packets. The rate of the traffic forwarded to the destination is calculated as follows:

$$rate = \frac{\mathcal{N}}{curTs - preTs} \tag{1}$$

According to the traffic volume change on links, the destination can locate link failures and detect abnormal traffic, *e.g.*, packet loss, traffic hijacking [38]. However, because of link delay and the different tag update interval, the inconsistent time of the rates may cause a big error. Therefore, it is meaningful to count the number of replicas of the tag received. As shown in Figure 4, $Replicas$ can be used to accurately evaluate the real-time traffic rate on the router before the next tag arrives. Before the next tag arrives, the number of packets is calculated by the formula: $sc = Replicas/Q$.

DDoS Traffic Discovery. The destination maintains several observation periods. In the periods without attacks, normal traffic distribution pattern can be established. It has some features, *e.g.*, tag update frequency, the packet loss rate, IP forwarding paths. In the periods with DDoS attacks, abnormal characteristics indicates that there may be attack traffic.

For example, we can set the parameter $\mathcal{N}$ which makes the router without attack traffic update the tag approximately every ten seconds. When the DDoS traffic against the destination arrives, the tag is updated once in hundreds of milliseconds because the traffic forwarded to the destination has increased dozens of times on the hit router. The high frequency of updating tags indicates that there may be attack traffic. When the anomaly shows up on multiple links, we think that a DDoS attack against the victim may occurr. However, multiple reasons may also cause the same results, so further source detection and feature detection are needed to launch collaborative mitigation.

In this paper, the method of IP source detection is traceback. DDoS feature detection is not the focus of our work. Therefore, we reproduce some known DDoS attacks to simulate the distribution of attack traffic and demonstrate the effectiveness of collaborative DDoS mitigation.

E. Collaborative Mitigation

Following the above discussion, we propose the collaborative mitigation scheme which consists of the following steps.

*1) **Identifying DDoS Traffic**:* We focus on two general DDoS attacks: Destination Flooding Attacks and Link Flooding Attacks [39]. The former directly sends the traffic flooding the victim in order to exhaust the victim's resources. This type of attack traffic is visible to the victim. The victim identifies this type of attack by perceiving the distribution of traffic and detecting the aggregated traffic at the destination.

The latter aims to congest the bottleneck link. This type of attack traffic may be visible to the victim. Specifically, it is challenging for the victim to locate the attacked link because the router does not notify the destination of the packet loss. If the bottleneck link is attacked, the packet loss rate increases significantly because of the volume difference between the outgoing traffic and the incoming traffic, so that the victim identifies and locates the type of attack.

*2) **Selecting Collaborators**:* Selecting collaborators based on traffic awareness (§ IV-D) is the key to realizing efficient, rapid-response and low-cost collaborative DDoS mitigation. The selection of the collaborators depends on the forwarding path of most attack traffic and the available resources of routers on the path. The victim needs to trade off resources of collaborators and accuracy of mitigation. On the routers close to the victim, the more attack traffic is aggregated, and the more resources required for a single router to identify and filter the attack traffic with less additional overhead and higher detection accuracy [40]. On the routers far away from the victim, the attack traffic is more scattered, and the router requires fewer resources to identify the attack traffic. However, the victim pays more overhead to request more collaborators to filter attack packets collaboratively. An efficient attack packet identification algorithm can offer TAP a better trade-off.

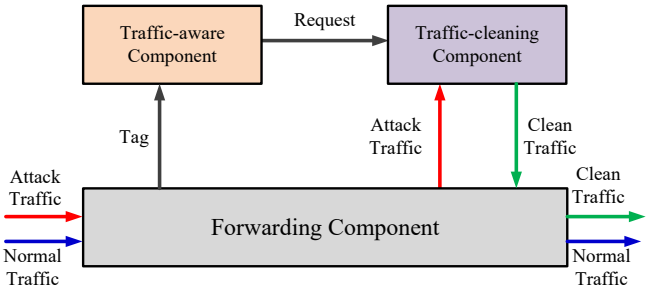


Fig. 5. Three components in the network. The forwarding component adds tags to the forwarded packets. The traffic-aware component collects tags to detect DDoS and request traffic-cleaning components of collaborators to clean traffic.

*3) **Setting Rate Thresholds**:* After receiving the victim's request, collaborator routers have to clean attack traffic and limit the volume. When facing DDoS flooding the terminal system, the victim sets packet forwarding rate thresholds (Upward threshold) for the collaborator routers to rapidly respond to DDoS and reduce damage to legitimate users. When facing DDoS that aims to congest the bottleneck link, the effective defense is that the routers on the bottleneck link reserve network bandwidth (Downward threshold) for the legitimate traffic to the victim. After receiving the rate threshold, collaborators immediately forward the cleaned traffic or reserve network bandwidth for legitimate traffic to the victim at the threshold rate.

*4) **IP-based Filtering**:* A simple way for collaborators to filter packets is to block the source IP addresses in blacklists. IP-based filtering uses few resources to achieve outstanding effect against most known attacks. Traceback can help the victim identify spoofing IP by comparing the paths in different periods or the IP range owned by the source. The victim generates blacklists for each collaborator by IP traceback.

*5) **Improving Collaboration Policy**:* The victim performs traffic distribution awareness on collaborator routers and its downstream network to evaluate the effect of filtering traffic on the hit routers and adjust collaboration strategies (*i.e.*, reselecting collaborators, adjusting the thresholds). Besides, the victim can request collaborators to filter the spoofed IP and allow trusted IP.

V. Implementation and Evaluation

In this section, simulation experiments are conducted to demonstrate that TAP is efficient, effective and rapid-response.

A. Implementation

Following the above discussion, we propose the collaborative mitigation framwork based on traffic awareness and IP traceback. We use OMNet to conduct the prototype of TAP and the simulation of the collaborative network. As shown in Figure 5, the network entity consists of three components, forwarding component, traffic-aware component and traffic-cleaning component.

*1) **Forwarding Component**:* Each router installs a forwarding component (§IV). The forwarding component uses the IP address prefix of the protected destination as a secure routing entry. The egress port of the traffic forwarded to the destination IP can be set by dynamic autonomous routing. When no attack, the forwarding rate is not limited by the destination. After the router determines $\mathcal{N}$ and p, the destination is notified of the parameters. The router has a shared key with the destination, using the existing protocols [36].

*2) **Traffic-aware Component**:* A destination has a traffic-aware component (§IV-D). The defense component can be a trusted server in the AS or a third-party module with functions, *e.g.*, traffic awareness, IP traceback and attack detection in the router. It has shared keys with forwarding components of other entities, decrypts the tag and records source IP in Bloom filters. It can reconstruct the attack traffic distribution to locate hit routers and select collaborators for collaborative mitigation.

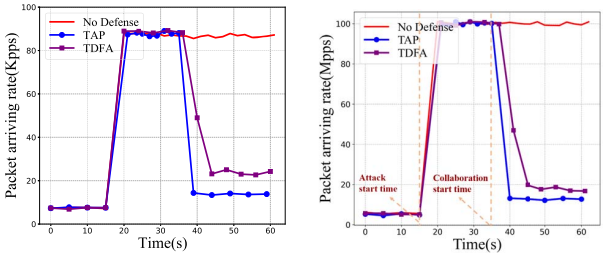
*3) **Traffic-cleaning Component**:* A forwarding entity has a cleaning component (§IV-E). It may be a server in the AS or a third-party module in the router for filtering packets and limiting the rate of the outgoing traffic. The entity receives the collaboration request from the victim and then forwards the related traffic to the component that limits the forwarding rate or reverse bandwidth. Various detection algorithms or tools can be implemented to identify known and unknown attack packets.

B. Simulation Setup

Since the common feature of DDoS attack is the abnormal traffic distribution, we simulate the distribution of DDoS traffic in a real-world topology to prove the effectiveness of TAP. More specifically we build our simulators using OMNet++ that is a modular and component-based C++ simulation framework for discrete-event systems. We simulate the sending of packets using Poisson process. For a real-world topology, we have used one dataset of Internet Topology Zoo [41]. The original dataset contains dozens of routing nodes. We connect each routing node to ten end nodes, thus expanding the data set to include hundreds of nodes.

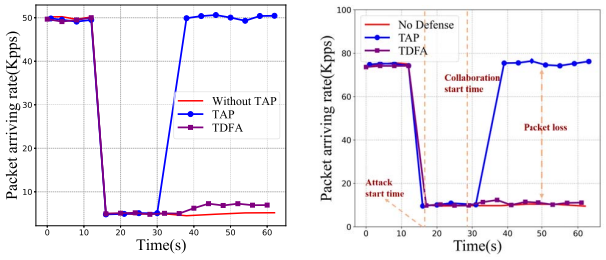
We simulate an experiment network with routing nodes that implement forwarding components and traffic-cleaning components and end nodes that has implemented traffic-aware components. The probability of the routing node marking the packet is denoted as p. The end node can request collaboration from the router by sending packets containing IP blocking lists and rate thresholds. These packets represent the communication overhead for collaboration. The maximum number of hops between two end nodes is 15.

End nodes can be divided into two categories in the experiment: normal hosts and attack hosts. We set the packet sending rate parameter to 1 *Kpps* for the normal hosts and 10 *Kpps* for the attack hosts. In addition, the maximum packet receiving rate of the victim C is set to 10 *Mpps* and the throughput of the bottleneck link to 100 *Kpps*.



(a) (b)

Fig. 6. Case 1: DDoS attack against the victim. (a) shows the packet arriving rate on a hit router. (b) shows the total number of packets per second received by the victim.



(a) (b)

Fig. 7. Case 2: DDoS attack against the bottleneck. (a) shows the forwarding rate of legit packets routed to the victim on the bottleneck link. (b) shows the number of packets per second received by the victim.

C. Case Studies

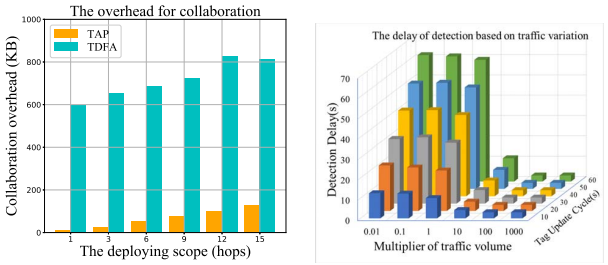
We have studied the performance of TAP in the face of two types of DDoS and compared with TDFA by using the following important metrics: (i) the response time that refers to the time from attack start time until the attack traffic is reduced to a tolerable level ; (ii) the communication overhead used for collaboration which consists mainly of IP lists and control information, which implies the scalability.

1) Case 1: In the case, many attack hosts simultaneously send a large volume of attack traffic to the victim. As shown in Figure 6, the packet arriving rates sharply increase on the hit router and the victim. The volume of traffic, received by the victim greatly exceeds its capacity. After about ten seconds, the peak traffic is quickly reduced to an acceptable level at the victim. In contrast, TDFA takes more time to slowly reduce attack traffic. During the first ten seconds of DDoS, the victim receives more tags generated by the hit router to perceive the abnormal traffic distribution and selects optimal collaborators for defense against DDoS. After the collaborators clean the traffic routed to the victim, the forwarding rate is subject to the rate threshold, and the packet arriving rate at the victim is approximately 100*Kpps*.

In order to reduce false positives as much as possible, the collaborators forward the clean traffic at the threshold rate in Figure 6(a), which enables TAP to keep the packet loss rate of legal traffic at a low level. Figure 8(a) shows the communication overhead of collaboration is positively related to the number and distance of selected collaborators. TAP efficiently selects a small number of hit routers as collaborators, while TDFA takes more packet overhead and selects more edge routers as collaborators. Compared to TDFA, TAP saves 80% of collaboration overhead in Figure 8(a) and only requires response time of several seconds in Figure 6(b).

2) Case 2: The experiment simulates DDoS targeting the bottleneck link. The attacker sends lots of packets whose forwarding path contains the bottleneck link. The attack traffic squeezes the bottleneck link's bandwidth and causes the router to discard legitimate packets of the victim in bulk. However, the attack traffic is not directly sent to the victim so that the victim cannot perceive it.

When facing DDoS against bottleneck link, the arriving rate



(a) (b)

Fig. 8. (a) shows the communication overhead for collaboration between the victim and the hit routers in Case 1. (b) shows the detection delay in relation to tag update cycle and abnormal traffic volume.

of the victim's packet on the bottleneck link remains stable, but it receives little traffic and less tags of the bottleneck link. The inbound traffic routed to the victim on the bottleneck link is larger than the outbound traffic, leading to the high packet loss rate perceived within a tag update cycle in Figure 7(b). After about ten seconds, the victim requests the routers on the bottleneck link to reserve enough bandwidth resources in Figure 7(a).

By contrast, TDFA does not sense traffic changes and flow disruption on the bottleneck link. Selecting edge routers as collaborators is helpless to provide network bandwidth on the bottleneck link for legit traffic, so that TDFA's curve is almost identical to the curve without any defense in Figure 7(b). After the collaborative mitigation is activated in TAP, the packet loss rate is still 1% and the case has little impact on the victim's traffic.

D. Performance Analysis

Figure 8(b) shows that the shorter the tag update cycle, the faster the victim finds abnormal traffic to detect possible DDoS attacks and locate the attacked link, implying more computational overhead on the router. In addition, the more attack traffic is sent to the victim, the faster the victim is able to detect the traffic attack, which obliges us to tradeoff the possible attack scale, the overhead of routers and the collaboration effect.

The response time to collaboration requests depends on the router's rules for identifying and filtering traffic. Compared to per-flow optimization of TDFA, TAP is more efficient, rapid-response, and places higher demand on the routing node. We advocate that for TAP, it is more practical for an AS to act as a routing node.

Figure 8(a) shows both TAP and TDFA are in effect in Case 1, but TAP significantly reduces the overhead used for collaboration in seconds. When TDFA cannot cope with Case 2, as shown in Figure 7(b), TAP is still able to mitigate DDoS damage in seconds with essentially no change in overhead.

Based on the above analysis, TAP is more preferable for the inter-AS scenarios with fewer hops and the overhead increases with the number of AS hops. With a certain probability, the bandwidth overhead increases approximately linearly, while the collaboration overhead increases non-linearly due to the number of hit routers selected and the number of flows with spoofed addresses to be filtered.

VI. Conclusion

In this paper, we propose TAP, which is a collaborative DDoS mitigation framework. Specifically, the victim selects the hit routers on the path of most attack traffic as collaborators to mitigate traffic attacks and quickly respond to mitigation requests by assigning forwarding rate thresholds based on traffic awareness and IP traceback. DDoS collaborative characteristic detection and flow-granularity traffic control are interesting improvements in future work.

References

[1] D. Makrushin, "The cost of launching a ddos attack," 2017. [Online]. Available: https://securelist.com/the-cost-of-launching-a-ddos-attack/77784/

[2] A. Shield, "Threat landscape report - Q1 2020." [Online]. Available: https://aws-shield-tlr.s3.amazonaws.com/2020-Q1_AWS_Shield_TLR.pdf

[3] Y. Jia, Y. Liu, G. Ren, and L. He, "Revisiting inter-as ip spoofing let the protection drive source address validation," in *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)*. IEEE, 2017, pp. 1–10.

[4] P. Ferguson and D. Senie, "Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing," *RFC*, vol. 2827, pp. 1–10, 2000. [Online]. Available: https://doi.org/10.17487/RFC2827

[5] L. He, G. Ren, Y. Liu, and J. Yang, "Pavi: Bootstrapping accountability and privacy to ipv6 internet," *IEEE/ACM Transactions on Networking*, vol. 29, no. 2, pp. 695–708, 2021.

[6] J. Ioannidis and S. M. Bellovin, "Implementing pushback: Router-based defense against ddos attacks," in *Proceedings of the Network and Distributed System Security Symposium, NDSS 2002, San Diego, California, USA*. The Internet Society, 2002. [Online]. Available: https://www.ndss-symposium.org/ndss2002/implementing-pushback-router-based-defense-against-ddos-attacks/

[7] Y. Jia, Y. Liu, G. Ren, and L. He, "Risp: An rpki-based inter-as source protection mechanism," *Tsinghua Science and Technology*, vol. 23, no. 1, pp. 1–12, 2018.

[8] B. Rashidi, C. Fung, and E. Bertino, "A collaborative DDoS defence framework using network function virtualization," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 10, pp. 2483–2497, 2017.

[9] N. B. Mohammadi, C. Barna, M. Shtern, H. Khazaei, and M. Litoiu, "CAAMP: completely automated ddos attack mitigation platform in hybrid clouds," in *12th International Conference on Network and Service Management, CNSM 2016, Montreal, QC, Canada, October 31 - Nov. 4, 2016*. IEEE, 2016, pp. 136–143. [Online]. Available: https://doi.org/10.1109/CNSM.2016.7818409

[10] S. M. Hezavehi and R. Rahmani, "An anomaly-based framework for mitigating effects of ddos attacks using a third party auditor in cloud computing environments," *Cluster Computing*, vol. 23, no. 4, pp. 2609–2627, 2020.

[11] J. Cao, Y. Liu, M. Liu, L. He, Y. Jia, and F. Yang, "psav: A practical and decentralized inter-as source address validation service framework," in *2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS)*, 2021, pp. 1–7.

[12] A. Mortensen, T. R. K., F. Andreasen, N. Teague, and R. Compton, "Ddos open threat signaling (DOTS) architecture," *RFC*, vol. 8811, pp. 1–29, 2020. [Online]. Available: https://doi.org/10.17487/RFC8811

[13] A. Yaar, A. Perrig, and D. Song, "Stackpi: New packet marking and filtering mechanisms for DDoS and IP spoofing defense," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 10, pp. 1853–1863, 2006.

[14] ——, "FIT: Fast internet traceback," in *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, vol. 2. IEEE, 2005, pp. 1395–1406.

[15] V. A. Foroushani and A. N. Zincir-Heywood, "TDFA: traceback-based defense against DDoS flooding attacks," in *28th IEEE International Conference on Advanced Information Networking and Applications, AINA 2014, Victoria, BC, Canada, May 13-16, 2014*, L. Barolli, K. F. Li, T. Enokido, F. Xhafa, and M. Takizawa, Eds. IEEE Computer Society, 2014, pp. 597–604.

[16] A. C. Snoeren, "Hash-based IP traceback," in *Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, August 27-31, 2001, San Diego, CA, USA*, R. L. Cruz and G. Varghese, Eds. ACM, 2001, pp. 3–14.

[17] X. Liu, X. Yang, and Y. Xia, "Netfence: preventing internet denial of service from inside out," in *Proceedings of the ACM SIGCOMM 2010 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, New Delhi, India, August 30 -September 3, 2010*, S. Kalyanaraman, V. N. Padmanabhan, K. K. Ramakrishnan, R. Shorey, and G. M. Voelker, Eds. ACM, 2010, pp. 255–266. [Online]. Available: https://doi.org/10.1145/1851182.1851214

[18] S. Hameed and H. A. Khan, "SDN based collaborative scheme for mitigation of DDoS attacks," *Future Internet*, vol. 10, no. 3, p. 23, 2018.

[19] B. Rodrigues, T. Bocek, A. Lareida, D. Hausheer, S. Rafati, and B. Stiller, "A blockchain-based architecture for collaborative DDoS mitigation with smart contracts," in *Security of Networks and Services in an All-Connected World - 11th IFIP WG 6.6 International Conference on Autonomous Infrastructure, Management, and Security, AIMS 2017, Zurich, Switzerland, July 10-13, 2017, Proceedings*, ser. Lecture Notes in Computer Science, vol. 10356. Springer, 2017, pp. 16–29.

[20] X. Wang, K. Xu, W. Chen, Q. Li, M. Shen, and B. Wu, "ID-based SDN for the internet of things," *IEEE Network*, vol. 34, no. 4, pp. 76–83, 2020.

[21] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, "Controlling high bandwidth aggregates in the network," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 3, pp. 62–73, 2002.

[22] S. Savage, D. Wetherall, A. R. Karlin, and T. E. Anderson, "Practical network support for IP traceback," in *Proceedings of the ACM SIGCOMM 2000 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, August 28 - September 1, 2000, Stockholm, Sweden*, C. Partridge, Ed. ACM, 2000, pp. 295–306.

[23] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Network support for IP traceback," *IEEE/ACM transactions on networking*, vol. 9, no. 3, pp. 226–237, 2001.

[24] D. X. Song and A. Perrig, "Advanced and authenticated marking schemes for IP traceback," in *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No. 01CH37213)*, vol. 2. IEEE, 2001, pp. 878–886.

[25] M. T. Goodrich, "Probabilistic packet marking for large-scale IP traceback," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 15–24, 2008.

[26] T. Peng, C. Leckie, and K. Ramamohanarao, "Adjusted probabilistic packet marking for IP traceback," in *International conference on research in networking*. Springer, 2002, pp. 697–708.
[27] B. Wu, K. Xu, Q. Li, Z. Liu, Y. Hu, M. J. Reed, M. Shen, and F. Yang, "Enabling efficient source and path verification via probabilistic packet marking," in *26th IEEE/ACM International Symposium on Quality of Service, IWQoS 2018, Banff, AB, Canada, June 4-6, 2018*. IEEE, 2018, pp. 1–10.
[28] A. Yaar, A. Perrig, and D. Song, "Pi: A path identification mechanism to defend against DDoS attacks," in *2003 Symposium on Security and Privacy, 2003*. IEEE, 2003, pp. 93–107.
[29] S. Saurabh and A. S. Sairam, "Increasing the effectiveness of packet marking schemes using wrap-around counting Bloom Filter," *Secur. Commun. Networks*, vol. 9, no. 16, pp. 3467–3482, 2016.
[30] S. Hassan, "A linear packet marking probability function for fast congestion notification (FN)," *IJCSNS*, vol. 9, no. 5, p. 45, 2009.
[31] S. Simpson, S. N. Shirazi, A. K. Marnerides, S. Jouet, D. Pezaros, and D. Hutchison, "An inter-domain collaboration scheme to remedy DDoS attacks in computer networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 15, no. 3, pp. 879–893, 2018.
[32] J. Steinberger, B. Kuhnert, A. Sperotto, H. Baier, and A. Pras, "Collaborative DDoS defense using flow-based security event information," in *2016 IEEE/IFIP Network Operations and Management Symposium, NOMS 2016, Istanbul, Turkey, April 25-29, 2016*, S. Oktug, M. Ulema, C. Cavdar, L. Z. Granville, and C. R. P. dos Santos, Eds. IEEE, 2016, pp. 516–522.
[33] Y. Zhao, K. Xu, H. Wang, B. Li, and R. Jia, "Stability-based analysis and defense against backdoor attacks on edge computing services," *IEEE Netw.*, vol. 35, no. 1, pp. 163–169, 2021. [Online]. Available: https://doi.org/10.1109/MNET.011.2000265
[34] Y. Zhao, K. Xu, H. Wang, B. Li, M. Qiao, and H. Shi, "Mec-enabled hierarchical emotion recognition and perturbation-aware defense in smart cities," *IEEE Internet of Things Journal*, pp. 1–1, 2021.
[35] B. Rashidi and C. J. Fung, "Cofence: A collaborative DDoS defence using network function virtualization," in *12th International Conference on Network and Service Management, CNSM 2016, Montreal, QC, Canada, October 31 - Nov. 4, 2016*. IEEE, 2016, pp. 160–166.
[36] T. H.-J. Kim, C. Basescu, L. Jia, S. B. Lee, Y.-C. Hu, and A. Perrig, "Lightweight source authentication and path validation," in *Proceedings of the 2014 ACM conference on SIGCOMM*, 2014, pp. 271–282.
[37] Wikipedia contributors, "Bloom filter," 2020. [Online]. Available: https://en.wikipedia.org/wiki/Bloom_filter#Algorithm_description
[38] B. Wu, K. Xu, Q. Li, B. Liu, S. Ren, F. Yang, M. Shen, and K. Ren, "RFL: Robust fault localization on unreliable communication channels," *Computer Networks*, vol. 158, pp. 158–174, 2019.
[39] X. Liu, X. Yang, and Y. Lu, "To filter or to authorize: network-layer DoS defense against multimillion-node botnets," in *Proceedings of the ACM SIGCOMM 2008 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Seattle, WA, USA, August 17-22, 2008*, V. Bahl, D. Wetherall, S. Savage, and I. Stoica, Eds. ACM, 2008, pp. 195–206.
[40] K. Singh, P. Singh, and K. Kumar, "A systematic review of IP traceback schemes for denial of service attacks," *Comput. Secur.*, vol. 56, pp. 111–139, 2016.
[41] T. I. Murphy, "Line spacing in latex documents," [EB/OL], http://www.topology-zoo.org/dataset.html Accessed April 4, 2010.