Combating DDoS Attacks with Fair Rate Throttling

Abdullah Yasin Nur
Department of Computer Science
University of New Orleans
New Orleans, LA, USA 70148
Email: ayn@cs.uno.edu

***Abstract*—Distributed Denial of Service (DDoS) attacks are among the most harmful cyberattack types in the Internet. The main goal of a DDoS defense mechanism is to reduce the attack's effect as close as possible to their sources to prevent malicious traffic in the Internet. In this work, we examine the DDoS attacks as a rate management and congestion control problem and propose a collaborative fair rate throttling mechanism to combat DDoS attacks. Additionally, we propose anomaly detection mechanisms to detect attacks at the victim site, early attack detection mechanisms by intermediate Autonomous Systems (ASes), and feedback mechanisms between ASes to achieve distributed defense against DDoS attacks. To reduce additional vulnerabilities for the feedback mechanism, we use a secure, private, and authenticated communication channel between AS monitors to control the process. Our mathematical model presents proactive resource management, where the victim site sends rate adjustment requests to upstream routers. We conducted several experiments using a real-world dataset to demonstrate the efficiency of our approach under DDoS attacks. Our results show that the proposed method can significantly reduce the impact of DDoS attacks with minimal overhead to routers. Moreover, the proposed anomaly detection techniques can help ASes to detect possible attacks and early attack detection by intermediate ASes.**

***Index Terms*—DoS, DDoS, Rate Adjustment, Router Throttle**

I. Introduction

The Internet is designed to forward packets with minimal intervention, including malicious packets. Cybercriminals take advantage of this freedom in the architecture to deploy cyberattacks towards various targets. Cyberattacks can be defined as malicious and deliberate attempts to breach the information system, computer networks, or infrastructures that use the Internet as a communication medium. Usually, attackers seek some benefit from disrupting the victim's network, including financial gain, reputation, political reasons, or cyberwarfare. Denial-of-Service (DoS) attack and its variants, such as Distributed Denial-of-Service attacks (DDoS), are one of the most hazardous attack types in the Internet. In DoS attacks, perpetrators aim to exhaust a system to make it unavailable to provide services to intended users. Typically, the attack involves flooding a target with an excessive amount of traffic to overload the system to disrupt its services. DDoS attack is a more severe version of the DoS attack, where a large number of hosts simultaneously attack a target. In DDoS, the perpetrators often leverage the attack effect by compromising multiple hosts through a common vulnerability and use all compromised hosts to flood the victim site.

Many jurisdictions have laws considering that DoS attacks are illegal. For example, the Computer Fraud and Abuse Act (CFAA) in the US indicates that DoS attacks perpetrators may be charged legal offenses at the federal level with penalties that include imprisonment and fine penalty for any damage to the victim site. In September 2012, a large-scale DDoS attack targeted the websites of several US banks, including JPMorgan Chase, Bank of America, and SunTrust Banks. The attack was successful and disrupted their services for extended periods of time. In December 2015, perpetrators targeted the BBC and took its on-demand TV service, iPlayer services, and radio services down for several hours. In October 2016, a large-scale DDoS attack targeted the DNS infrastructure of Dyn, which caused major Internet platforms unaccessible, including Amazon, Netflix, Twitter, and the New York Times. In April 2018, attackers targeted GitHub with 1.35 Tbps, which corresponds to 126.9 million packets per second. Most recently, Amazon reported a world record DDoS attack sustaining a 2.3 Tbps to their Amazon Web Services.

Unlike most cyberattacks, DDoS attacks are usually not caused by a vulnerability of the victim site's network. Under standard operation time, TCP-like congestion control usually provides a fair usage of the available resources in the victim site. However, the DDoS attacks do not obey the congestion rules and send packets at a drastic rate. Also, filtering the attack traffic within the Autonomous System (AS) network is not a valid solution because it creates congestion in the incoming links. Moreover, the rogue traffic also affects intermediate ASes, which carry the attack traffic. These ASes are unintentional victims. Hence, the main goal of a DDoS defense mechanism is to reduce the attack's effect as close as possible to their sources to prevent malicious traffic in the Internet.

We have communicated with several Internet Service Providers (ISP) giving service worldwide regarding the monitoring and filtering the rogue traffic within their network. ISPs have several motivations to reduce malicious activities inside their network. Some of the motivations are reputation, economic incentives, resource management, and government regulations. Major ISPs tend to monitor the internal traffic and drop the malicious packet before relaying to the next AS for reputation. However, they need a clear sign that the packet is malicious. Most of the DDoS packets act like regular traffic packets with having a proper header, request, and option field. Therefore, it is hard to analyze the type of the packet without the victim site's feedback. ISPs avoid blocking the source without a strong reason because blocking their subscribers who are intended users would be a reputation problem for the ISPs. Next, economic incentives have an essential motivation for ISPs to reduce malicious traffic. Customer-to-Provider (c2p) is one of the popular business relations between ASes. In a c2p relation, the provider AS provides global reachability to its customer AS. In return, the customer pays to the provider for the traffic exchanged between them. By that means, ISPs pay money to other ISPs to relay the traffic, which is not desirable in an attack case because it is a waste of money. Attack traffic travels inside of an ISP use the bandwidth of the internal links. Since the bandwidth is limited, ISPs do not want to waste their bandwidth on malicious traffic, which brings another essential motivation, resource management. Finally, government regulations encourage the ISPs to reduce the botnet activity within their network (e.g., Federal Communications Commission US Anti-Bot Code of Conduct).

In this paper, we propose a proactive resource management method via rate throttling. Additionally, we propose attack detection mechanisms at the victim site, early attack detection mechanisms, and feedback mechanisms between ASes to achieve distributed defense against DDoS attacks. We use Bollinger bands, Stochastic Oscillators, trusted IP address control, passive IP set control, and passive AS set control to detect traffic anomalies between two ASes to detect possible DDoS attacks. Each AS on the path between attackers and the victim site analyzes the traffic pattern. In case an AS detects a traffic anomaly, e.g., a rapid increase in traffic flow between two hosts, it sends a question message to the destination site to ask if it is under attack. The feedback mechanism helps ASes to prevent incorrect filtration. ASes which have trust relations can discard the attack packets directly without additional concerns. However, not all ASes have trust relations, which can create additional problems to filter out the entire traffic of a customer AS without a clear sign that it is an attacker. Therefore, we provide a new rate management technique to throttle possible attack traffic by using feedbacks from upstream ASes.

We conducted several experiments using a real-world dataset to demonstrate the efficiency of our approach under DDoS attacks. Our results show that our anomaly detection techniques help ASes to detect possible attacks and early attack detection by intermediate ASes. Additionally, our proposed rate throttling algorithm provides a fair rate adjustment to protect the destination. Our approach complies with the current IP protocol and does not require any changes in the protocol itself. Similar to other defense mechanisms, the proposed approach in this study requires support from router vendors and Internet Service Providers. The experimental results show that our approach can be very effective against DoS and DDoS attacks with a small overhead on the routers.

The rest of the paper is organized as follows. In Section II, we present the related work. We explain the details of our approach in Section III. Section IV demonstrates our experimental results. Finally, we conclude the paper in Section V.

II. Related Work

Researchers have proposed several DoS defense mechanisms over two decades [9]. We classify the defense mechanisms into three categories: attack detection, attack source identification, and attack reaction.

The main approach of attack detection techniques is to detect the DoS attacks by monitoring the incoming traffic. Detection techniques usually identify DoS attacks in case of an anomaly from changes in the observed traffic pattern. MULTOPS [5] is a data-structure that assumes that the packet rate between two hosts should be proportional during normal operations. In case of a significant change in the packet rate from one side of the flow indicates a volumetric DDoS attack. Since many hosts simultaneously attack a target in DDoS attacks, Peng et al. [6] assume an extreme increase in the set of new source IP addresses indicates an attack.

The second category is attack source identification. Attackers in the Internet can use IP spoofing to hide their real IP addresses. Therefore, if the victim site blocks the suspicious IP addresses, it may accidentally block one of its legitimate users. One of the solutions to prevent this problem is to infer the path between attackers and the destination site, which is called IP traceback [7, 8]. One of the earliest works can be credited to Savage et al. [7]. They propose the Fragment Marking Scheme, which uses the IP ID field in the IP packet header to probabilistically mark the partial path information. Once the victim site receives enough number of packets, it can construct the forward paths between attackers toward itself. Yaar et al. [3] use more space for encoding to decrease the number of required packets and reduce false positives. In our previous work, RRTrace [1], we propose a probabilistic packet marking scheme by exploiting the Record Route feature of the IP protocol. In RRTrace, a router inserts one of its IP addresses in the Record Route (RR) options field of a packet as long as there is room. In case there is no room in the RR field, the router rewrites the field with probability p or skips rewriting with probability $1 - p$. The victim site starts from an empty graph and gradually builds up the graph by incorporating the sub-paths from the received packets.

Attack reaction techniques involve resource management to mitigate the impact of DoS attacks in a timely fashion. High profile service providers, such as Microsoft and Yahoo, dynamically increase service and network resources during attacks [4]. The increasing popularity of cloud services brought new approaches to DoS defense [10, 11]. Cloud-based security companies such as Cloudflare and Imperva provide a cloud layer between their customers, which allows them to monitor and analyze traffic patterns in real-time. When a DDoS attack is detected by monitoring systems, they apply a filtering technique and drop the malicious traffic without forwarding to their customers. Mahajan et al. [16] propose an aggregate-based congestion control mechanism, which suggests monitoring and controlling high bandwidth aggregates at routers. An aggregate corresponds to a collection of packets sharing a common property such as source address, destination address, protocol type, or application type. The mechanism identifies the aggregates causing congestion and rate limit the aggregates at the local or upstream routers. Yau et al. [2] propose a feedback control scheme on the router to throttle the traffic flow with max-min fairness. The proposed scheme aims to proactively limit the traffic rate before it reaches the server. Malialis and Kudenko [17] introduce a decentralized approach, where upstream routers independently deploy multiple reinforcement learning agents. Upstream routers use multiagents to learn throttle towards the victim site. Xia et al. [18] propose a centralized router throttling method via reinforcement learning. They apply a deep deterministic policy gradient network for each router to reduce the communication cost.

In this work, we propose a new collaborative rate management method to combat volumetric DDoS attacks. Our method provides a trust-based and untrust-based collaboration scheme for ASes. In addition, we provide additional techniques for anomaly detection to discover DDoS attacks as close as to the attacker site. We provide a mathematical methodology for fair rate throttling and anomaly detection. Additionally, we introduce message types to exchange requests between ASes safely. Our results show that the proposed techniques require less overhead to routers and filtering systems of ASes. Moreover, it can be very effective for combating DDoS attacks and protect the destination site.

III. Methodology

The ultimate goal of the defense mechanism is to filter out the attack packets as close as the source. In a DDoS attack case, there can be thousands of attackers from thousands of ASes. ISPs avoid blocking the source without a strong reason because blocking their subscribers who are intended users would be a reputation problem. It is hard to detect the attack without the victim site's feedback. In this work, we propose a collaborative rate management approach to combat DDoS attacks. In case of an attack, the victim site sends rate throttling requests to its upstream ASes to reduce the attacking rate. Additionally, the upstream ASes monitor possible anomalies to early detect the possible attacks and communicate with possible victims to solve the problem as soon as possible.

ASes must fully control the request and reply messaging process to avoid additional vulnerabilities. For example, perpetrators can send lots of filtering requests to another AS, which is a different and new type of attack that we want to avoid. In order to achieve a more robust system, the control process is held by specific control servers deployed by each AS. Note that, most of the ASes already has these type of servers to monitor their network traffic and Quality-of-Service (QoS) parameter. Furthermore, network admins use traffic mirroring or tapping techniques to analyze the traffic within the internal network, e.g., NetFlow and IPFIX [19]. Therefore, the proposed method does not require additional cost to the ASes. The server nodes construct a secure, private, and authenticated communication channel between them. We use Public Key Infrastructure (PKI) framework for authentication. The framework contains two PKIs where the first one is employed for IP prefix attestation and the second one is for AS number attestation between end-hosts. The communication routing is over the path chosen by the Border Gateway Protocol (BGP), which is already a defacto intra-AS routing protocol. Next, each AS assigns a static IP address to its control servers and broadcasts the IP addresses to upstream ASes. Control servers hold the ”AS-ServerIP” pair in their database to be able to exchange messages. When a victim sends an ”I am under attack” (See Section III-D for details) request to its AS'es control server, the server evaluates the validity of the claim. If the claim is valid and the victim site is under attack, the AS sends a filtering request to each ASes' control server on the path. This approach helps for partial deployment cases because some of the ASes in the Internet may not want to collaborate. If the attackers' AS are not collaborative, the filter is deployed at the closest collaborative AS to the attacker. The request messages are specific for the control servers. ASes' egress routers drop the outgoing request packets and block the originator of the packet if the source is not a control server for removing the possible vulnerability.

The validity of a request is decided in two-folds, which are the deterministic and non-deterministic approaches. In the deterministic approach, ASes assume that the source is an attacker and drop all incoming packets from the source. The deterministic approach is for the ISPs, which have trust relations. Levy et al. [12] reported a collaboration between two major ISPs in the US, CenturyLink and AT&T, during DDoS attacks. This observation shows that ISPs are willing to collaborate to defeat the DDoS attacks. In the non-deterministic approach, ASes check the monitoring results to analyze the possible attackers' activity. In case the traffic sent by the source is above a threshold, the AS applies a rate-throttling to the source instead of filtering out all traffic.

A. Anomaly Detection

Anomaly detection is a significant part of DDoS defense. In order to make a filtering request, the victim site is required to distinguish malicious traffic and regular traffic. We identify DDoS attacks by analyzing anomalies in the observed traffic pattern by monitoring incoming traffic. We propose the following techniques for anomaly detection. Note that, the upstream ASes use the first two methods for possible early attack detection, whereas the victim site uses all five methods.
Bollinger Bands: It is a prevalent statistical approach to detect overbought and oversold stocks in the stock market. A similar approach is used for computing the TCP's retransmission timer timeout calculations [15]. We apply it to determine over packet and under packet sending cases. Gil et al. [5] proposed that the packet rate from Host A to Host B should be proportional to the packet rate from Host B to Host A during normal operations. Therefore, a dramatic change in the packet rate from one side of the flow indicates a volumetric DDoS attack. Bollinger bands generates a band at a standard deviation level above and below a moving average of the current rate. We assume that during a regular operation, the current flow rate should be within the band. In case the rate is out of the band, it

shows an anomaly of the current incoming traffic. We present the mathematical model for bollinger bands in Section III-C.

Stochastic Oscillator: Similar to bollinger bands, stochastic oscillator is a popular statistical momentum indicator for stocks. The stochastic oscillator presents the current rate in relation to the high and low range of the rate over a period of time. In case of the current rate coming from an AS close to the high range, the AS can be considered as a possible attacker. We present the mathematical model for the stochastic oscillator in Section III-C.

Trusted IP Address Control: User profiling is a common anomaly detection technique in cyberspace [20, 21]. The main goal for user profiling is to detect atypical behavior activities by analyzing the actions triggered by users. It is easy to create a profile for an intended user since the behavioral act would be similar during regular times, e.g., access the destination site with specific requests and activity times. Based on the user profiles, the victim site can crate the intended user IP address set and analyze the suspicious traffic coming from specific IP addresses via their profiles.

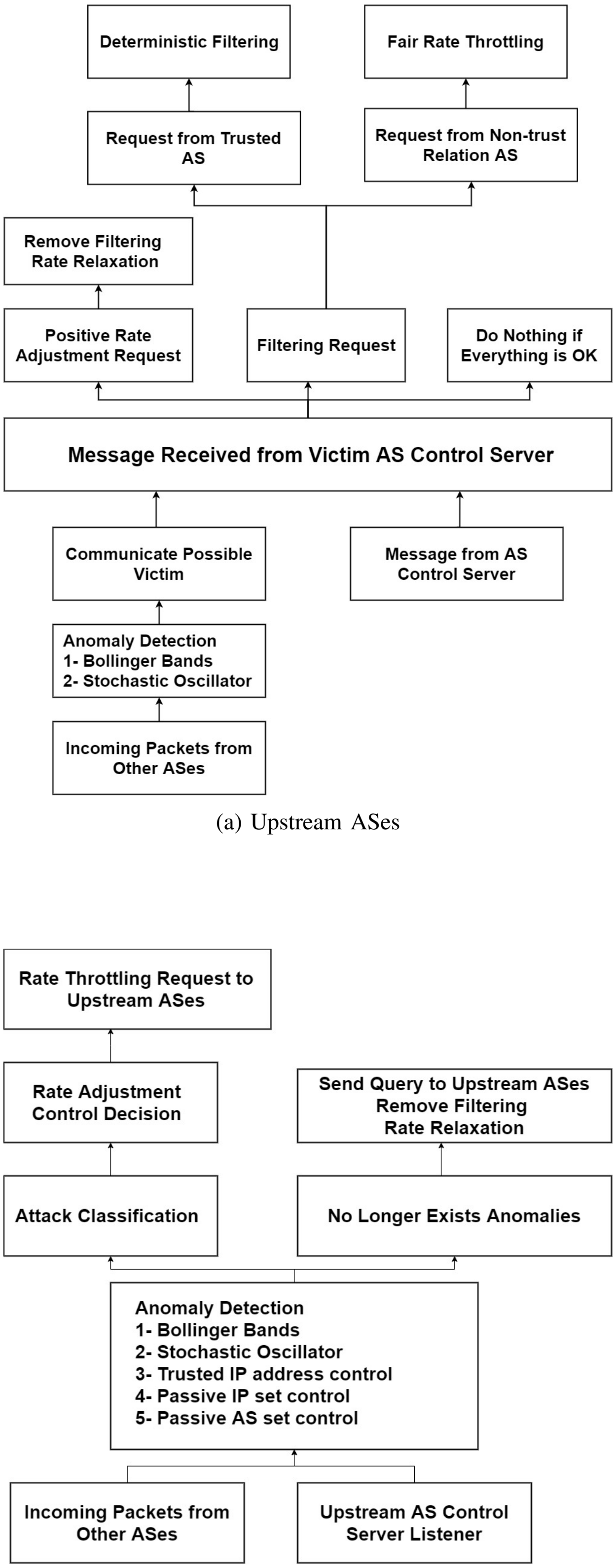
Passive IP Set Control: Since many hosts simultaneously attack a target in DDoS attacks, Peng et al. [6] assume that an extreme increase in the set of new source IP addresses indicates an attack. We also use this assumption to analyze possible volumetric DDoS attacks. We use the trusted IP address control technique explained above to eliminate the possible intended users, and extra analyze the unrecognized IP sources.

Passive AS Set Control: Similar to the previous assumption in passive IP set control, we assume that the number of different ASes that send traffic will be larger during DDoS attacks. By using this assumption, we also create AS profiling to analyze possible suspicious activities.

B. System Architecture

Figure 1 presents the conceptual architecture for upstream ASes and victim ASes. Upstream ASes between the attacker AS and the victim AS have two components, as represented in Figure 1a. The first component is early attack detection by analyzing the incoming traffic via anomaly detection techniques. We use bollinger bands and stochastic oscillator for that purpose. In case of anomaly detection, the upstream AS sends a query to the possible victim AS to ensure that the victim is under attack. Also, the upstream ASes' control servers listen to the other ASes for possible filtering requests. The second component of the defense mechanism is to respond to attacks. Whenever the upstream AS receives a request from another AS, it checks if the incoming request comes from a trusted AS or a regular AS. In case the filtering request comes from a trusted AS, the AS deploys a deterministic filtering mechanism to filter out traffic coming from the attacker AS. For other requests, the upstream AS deploys a fair rate throttling to reduce the amount of possible attack packets.

Figure 1b presents the conceptual architecture for victim AS. The victim site needs to analyze the packet traffics to detect possible attacks. Also, it needs to respond to queries


Deterministic Filtering
Fair Rate Throttling
Request from Trusted AS
Request from Non-trust Relation AS
Remove Filtering Rate Relaxation
Positive Rate Adjustment Request
Filtering Request
Do Nothing if Everything is OK
Message Received from Victim AS Control Server
Communicate Possible Victim
Message from AS Control Server
Anomaly Detection
1- Bollinger Bands
2- Stochastic Oscillator
Incoming Packets from Other ASes
Rate Throttling Request to Upstream ASes
Rate Adjustment Control Decision
Send Query to Upstream ASes
Remove Filtering
Rate Relaxation
Attack Classification
No Longer Exists Anomalies
Anomaly Detection
1- Bollinger Bands
2- Stochastic Oscillator
3- Trusted IP address control
4- Passive IP set control
5- Passive AS set control
Incoming Packets from Other ASes
Upstream AS Control Server Listener

(a) Upstream ASes

(b) Victim AS

Fig. 1: Conceptual Architecture for Upstream and Victim ASes

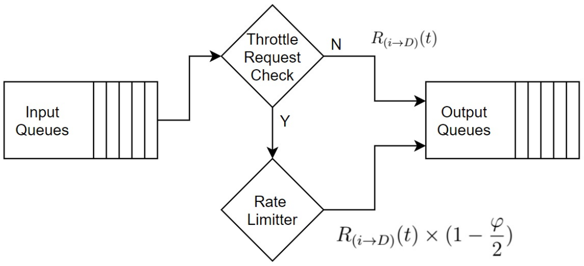


Fig. 2: Simplified Router Architecture with Rate Limiter

coming from upstream ASes for early attack detection. For anomaly detection, we use five different methods; bollinger bands, stochastic oscillator, trusted IP address control, passive IP set control, and passive AS set control. The victim AS is responsible for sending filtering requests to upstream ASes with a calculated throttling rate. Additionally, the victim AS keeps previously sent filtering requests logs. The reasons for maintaining logs are to resend a filtering request in case of timeout, send a new rate adjustment request in case of the increased level of attack, and send a remove filter request if the attack is no longer exists.

Figure 2 presents a partial view of a router that handles throttling requests from the victim site and applies rate-limiting for the attack sources. If an AS does not receive any throttling request, e.g., no attack case, the router transmits the data without applying specific filtering. However, when the AS receives a request from the victim, which specifies the attacker AS number, the router applies rate limiting based on the mathematical model presented in Section III-C.

C. Mathematical Model for Rate Throttling

In this part, we provide our mathematical approach for the fair rate throttling mechanism.

Assumptions:

- $G(V,E)$ is AS level graph where V presents ASes and E presents the connection between ASes. A subset of the graph $G'(V',E',D)$ presents a directed acyclic graph with a root D as a victim site where $G' \subset G$.
- The function $R_{(i\to j)}(t)$ presents the rate between AS_i and AS_j in time t.
- The function $R_{(i\to D)}(t)$ presents the total rate from AS_i to the victim site in time t.
- $R_{D_{cur}}$ is defined as the destination's current total rate, $R_{D_{reg}}$ is the destination's operation rate during normal time, $R_{D_{lim}}$ is the load limit rate of the destination, and $R_{D_{max}}$ is the maximum rate that the destination can operate.
- Our goal is to keep the destination under $R_{D_{max}}$ and close to $R_{D_{lim}}$. Therefore, the throttling rate (φ) and functions (f) are defined as equations 1 and 2.

The throttling rate φ is in the range of $[-1,1]$, where the value close to -1 presents under-rate and 1 presents over-rate. In case the value of $R_{D_{cur}} < R_{D_{lim}}$, the φ value will be negative, which presents no throttling required. Additionally, supposing the destination sent a filtering request previously, the negative value generates a new request to adjust the packet rate, e.g., the attack is ended and no need more further filtering. If the value of $R_{D_{cur}} > R_{D_{lim}}$, the φ value will be positive, and the destination requests a throttling from upstream ASes. The throttle model is fair because whenever the current rate is greater than but close to $R_{D_{lim}}$, φ value is close to 0. However, φ is close to 1 once the current rate gets closer to $R_{D_{max}}$.

$$\varphi = \begin{cases} \dfrac{R_{D_{cur}} - R_{D_{lim}}}{R_{D_{max}} - R_{D_{lim}}} & \text{if } R_{D_{cur}} > 0 \\ -1 & \text{if } R_{D_{cur}} = 0 \end{cases} \tag{1}$$

$$f = \begin{cases} HardThrottle & \text{if } R_{D_{cur}} \approx R_{D_{max}} \\ SoftThrottle & \text{if } R_{D_{cur}} \gtrapprox R_{D_{lim}} \\ NoThrottle & \text{if } R_{D_{cur}} \text{ is } [R_{D_{reg}}, R_{D_{lim}}] \\ Relaxation & \text{if } R_{D_{cur}} < R_{D_{reg}} \end{cases} \tag{2}$$

$$\begin{aligned} \Delta R_{(i\to D)} &= 1 - \frac{\varphi}{2} \\ R_{(i\to D)}(t) &= R_{(i\to D)}(t-1) \times (1 - \frac{\varphi}{2}) \end{aligned} \tag{3}$$

Equation 3 presents the rate adjustment. If the destination generates a throttling signal, it sends requests to upstream ASes to adjust the rate. In case the current rate is close to the maximum rate that the destination can operate, it requests a rate adjustment for half the current rate ($HardThrottle$). In case the current rate is smaller than the load limit, it requests a rate adjustment to soften the rate throttling ($Relaxation$).

Gil et al. [5] proposed that the packet rate between two hosts should be proportional during normal operations. Therefore, a dramatic change in the packet rate from one side of the flow indicates a volumetric DDoS attack. We use this assumption to create our anomaly detection approach. We define the rate function as 30 days exponential moving average (EMA) between 2 ASes. To define and discover the outlier cases where over packet sending from one AS towards the destination site, e.g., attack case, we use Exponential Bollinger Bands (EBB) and Stochastic Oscillator (SO). Equation 4 presents the exponential moving average traffic rate from AS_i to the destination in time t, where n is the number of days. We use EMA to

$$\begin{aligned} EMA_{(i\to D)}(t) = (&R_{(i\to D)_{cur}} \times \frac{2}{1+n}) \\ &+ (EMA(t-1) \times (1 - \frac{2}{1+n})) \end{aligned} \tag{4}$$

$$\sigma_{(i\to D)}(t) = \sqrt{\frac{\sum_{t=1}^{n}(R_{(i\to D)}(t) - \overline{R_{(i\to D)}})^2}{n-1}} \tag{5}$$

$$EBB_{upper_{(i\to D)}}(t) = EMA_{(i\to D)}(t) + m \times \sigma_{(i\to D)}(t) \tag{6}$$

$$EBB_{lower_{(i\to D)}}(t) = EMA_{(i\to D)}(t) - m \times \sigma_{(i\to D)}(t) \tag{7}$$

$$Stochastic\ Osc_{(i\to D)} = \frac{R_{(i\to D)_{cur}} - R_{(i\to D)_{low30}}}{R_{(i\to D)_{high30}} - R_{(i\to D)_{low30}}} \tag{8}$$

give more significance to the most recent traffic data behavior. Equation 5 presents the standard deviation of traffic rate from AS_i to the destination in time t. Finally, equations 6 and 7 present the upper and lower bound of Exponential Bollinger Band, where m is the number of standard deviation. In the stock market, the m value is usually set to 2. However, the TCP Round Trip Time estimation suggests 4 [15].

In case $R_{(i \to D)_{cur}} > EBB_{upper_{(i \to D)}}(t)$, the destination site detects the irregularity of the current packet transmission from AS_i. Next, the destination site checks if this irregularity, e.g., traffic peak in the current time, happened before in the last 30 days to ensure the claim. Therefore, it uses Stochastic Oscillator (SO) to double-check over packet transmission case. Equation 8 presents the Stochastic Oscillator, where $R_{(i \to D)_{low30}}$ presents the lowest traffic rate and $R_{(i \to D)_{high30}}$ presents the highest traffic rate over 30 days from AS_i to the destination. Assuming the highest traffic rate is not equal to the lowest traffic rate, the value of SO is between 0 to 1. A value close to 1 indicates an over packet sending from AS_i towards the destination. In case $R_{(i \to D)_{cur}} < EBB_{lower_{(i \to D)}}(t)$, the destination site analyzes the current attack condition and updates the upstream ASes if the attack is no longer exists.

D. Message Types

In this part, we explain the message types between the victim AS and upstream ASes. We construct a secure, private, and authenticated communication channel between them by using Public Key Infrastructure (PKI) framework for authentication. The framework contains two PKIs where the first one is employed for IP prefix attestation and the second one is for AS number attestation between end-hosts. The communication routing is over the path chosen by BGP, which is already a defacto intra-AS routing protocol.

I am under attack: After detecting an anomaly in the incoming traffic, the victim site calculates the rate throttling value and sends a query to all upstream ASes. The message includes throttling rate, suspicious AS number, and suspicious IP address. Depends on the throttling rate value, the upstream AS deploys hard throttle or soft throttle to adjust the traffic rate from suspicious AS to the victim AS.

Are you under attack?: This message type helps for early attack detection. In case an upstream AS between the attacker and the victim site detects an anomaly, it sends a message to the victim site to ensure that suspicious activity is attack activity. The message includes suspicious AS number, suspicious IP address, current rate value, and 30 days high and low traffic rate value. The victim site responds to this type of message by "I am under attack" or "filter is not required".

Filter is not required: The upstream ASes adjust the traffic rate by receiving throttling rate from the victim site. Since the upstream AS needs a confirmation from the victim site for removing the filter or adjusting the current rate of the AS in case the attack is no longer exists. The victim site uses this message type to inform upstream ASes to remove previously deployed filters. The message includes suspicious AS number, suspicious IP address, and relaxation parameter (e.g. $\varphi = -1$).

Extending timeout: In order to prevent additional vulnerabilities, the upstream ASes deploy their filters with a timeout value. If the attack still exists, the victim site needs to send a message indicates that the filter is still valid.

IV. Experimental results

One of the main drawbacks of the network security research field is to deploy the proposed methods in real networks for testing purposes. ISPs in the Internet are private companies having individual policies and strategies. Therefore, it is improbable to test the proposed strategies over a real large-scale Internet network. The best option is to generate a simulation environment and mimic the Internet. For that purpose, researchers usually use NS3 type simulators, which are discrete-event network simulators for Internet systems, or previously proposed network generators [23].

In our experiments, we used a network simulator that we developed in one of our previous work [1]. We implement it by using Matlab, which emulates the approach presented in Section III. To mimic the real-world Internet, we used real-world datasets presenting the current Internet topology. We used the CAIDA IPv4 Prefix-Probing Traceroute Dataset [13] consisting of more than 20 million (20,377,233) path traces. The dataset consists of $899,916$ different IP addresses. Note that we only included the loop-free path traces that reach their specified destinations. The minimum and maximum Interface level hop lengths in our dataset are 1 and 31, respectively. The average hop length is 15.43. Additionally, we used RouteViews prefix to AS mapping dataset obtained from CAIDA [14]. In order to generate an AS Level Internet topology, we mapped IP addresses reported in the traceroute dataset to their corresponding ASes. The dataset consists of 39,148 different ASes. The minimum and maximum AS level hop lengths in our dataset are 1 and 12, respectively. The average AS level hop length is 4.16. We used both datasets to generate Interface-AS dual-level topology maps.

Because of the simulation limitation, we needed to scale down our parameters. These parameters are traffic rates, server load rate limit, server maximum rate, and time. In our method, we use daily rates with a 30-day exponential moving average. However, we use 1 minute, representing 1 day for our experimental part. We use a memoryless Poisson distribution model for traffic generation. Poisson model is one of the most widely used techniques for traffic modeling in the Internet [22]. The probability density function of X is defined by:

$$P(X = k) = \frac{(\lambda^k) \times (e^{-k})}{k!}$$

where λ is the expected value of X. We assume that a rate from legitimate user to the victim site is between $[0, 50]$ Mbpm with a mean of 25 Mb per minute. We randomly choose a traffic rate for each legitimate user by using the Poisson distribution. Figure 3 presents a DoS attack case to present our bollinger band and stochastic oscillator techniques. Note that, we did not apply any rate throttling in this part of the experiment. Our main focus is to show the usefulness of anomaly detection

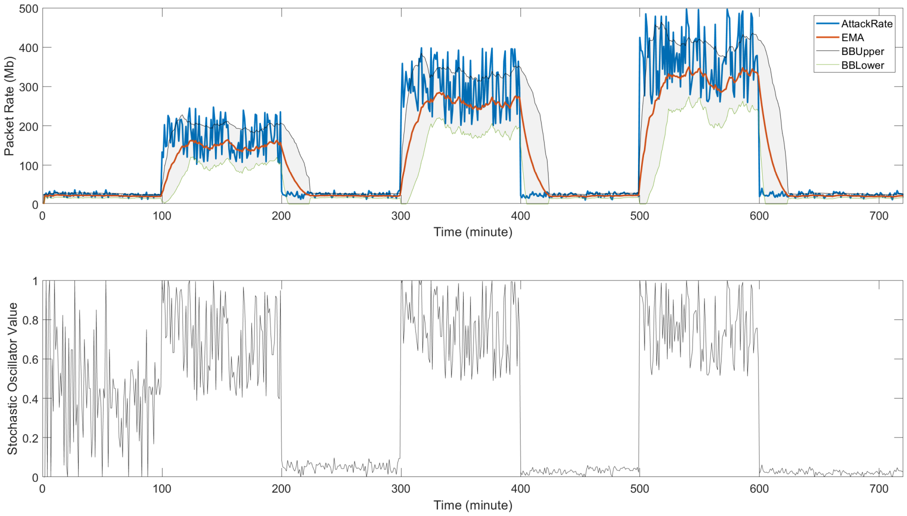


Fig. 3: Traffic coming from a source towards to the victim site without rate throttling

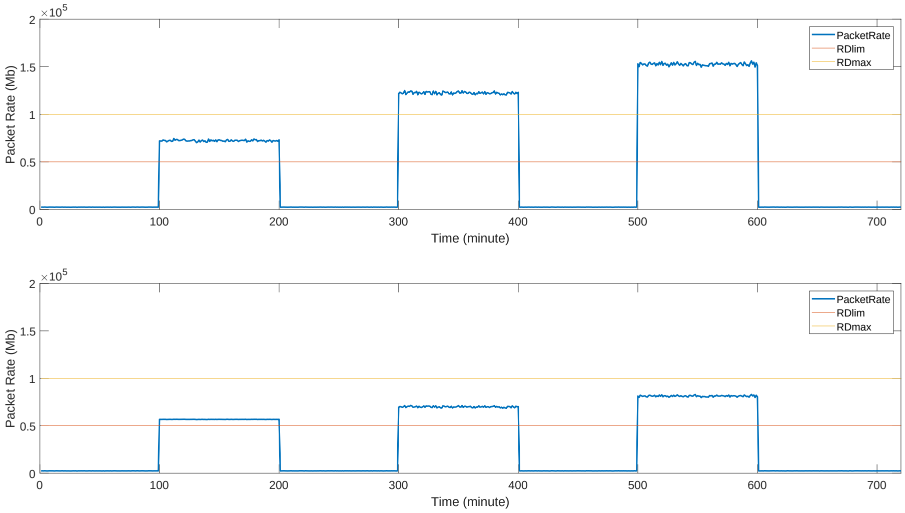


Fig. 4: DDoS attack towards to the victim site without rate throttling and with rate throttling

techniques that we presented. The blue line presents the traffic rate, the red line presents the exponential moving average, and the gray area presents the bollinger band where the black line is the upper bound and the green line is the lower bound of the band. In the figure, we show three different attack cases from a single attacker. We assume that between the time intervals $[0, 100]$, $[200, 300]$, $[400, 500]$, and $[600, 720]$, the user does not attack and sends only legitimate packets. The time interval between $[100, 200]$, the first attack starts with an attack rate between $[100, 250]$ Mbpm. The time interval between $[300, 400]$, the second attack starts with an attack rate between $[200, 400]$ Mbpm. Finally, the last attack occurs in the time interval $[500, 600]$ with an attack rate between $[250, 500]$ Mbpm. It is clear that during the attack time, the rate spikes outside of the bollinger band, which indicates suspicious activity. Also, whenever the attack ends, the rate drops under the lower band, indicating a possible end of the attack. Stochastic Oscillator is also helpful for us to determine possible attacks. The only drawback in the figure for Stochastic Oscillator is the time interval between 0 to 100, where the profiling of the previous activity phase. Since the first attack did not occur yet, the Stochastic Oscillator takes the maximum rate from the intended packet rate. However, the profiling phase occurs at the beginning of the test, whereas once it is done, the remaining part has no problems, e.g., attack case in 300, 400. Additionally, $R_{(i \to D)_{cur}}(t)$ - $R_{(i \to D)_{reg}}(t) \approx 0$ where $R_{(i \to D)_{reg}}(t) = EMA_{(i \to D)}(t)$ during not attack case. Therefore, the victim site is able to differentiate the usefulness of Stochastic Oscillator by checking the regular traffic rate from the suspected attacker.

Next, we implement a DDoS attack case and show rate throttling results. In this experiment, we randomly choose a victim and 500 path traces. In case we are unable to get enough path traces for a specific victim, we restart the victim selection process. We assume that the 100 of the end hosts are legitimate users without sending any attack packets. The remaining 400 of the end hosts are attackers. We follow the same attack rate and time intervals presented above for each end host. We set the load limit rate of the destination, $R_{D_{lim}}$, as $50,000$ Mb. We set the maximum rate that the destination can operate, $R_{D_{max}}$, as $100,000$ Mb.

Figure 4 presents the rate throttling algorithm in action. The top figure shows the attack without rate throttling and the bottom figure shows the attack with rate throttling. Note that, our goal is to keep the current rate of the destination under $R_{D_{max}}$ and close to $R_{D_{lim}}$. In the first attack at time $[100, 200]$, the attack rate is under the $R_{D_{max}}$. Therefore, the fair rate throttling algorithm applies soft throttling. However, the attacks in $[300, 400]$ and $[500, 600]$ exceeds the $R_{D_{max}}$. The victim site needs to take action immediately to prevent service failure. The algorithm applies hard throttling to reduce the attack traffic under the maximum load rate.

V. Conclusions

DDoS attacks are among the most harmful cyberattack types in the Internet. The main goal of a DDoS defense mechanism is to reduce the attack's effect as close as to their sources to prevent malicious traffic in the Internet. In this work, we propose a collaborative fair rate throttling mechanism to combat DDoS attacks. Additionally, we propose attack detection mechanisms at the victim site, early attack detection mechanisms, and feedback mechanisms between ASes to achieve distributed defense against DDoS attacks. To reduce additional vulnerabilities for the feedback mechanism, we use a secure, private, and authenticated communication channel between AS monitors to control the process. Our mathematical model presents proactive resource management, where the victim site calculates a throttling rate based on the current conditions of the traffic load and sends filtering requests to upstream routers to reduce the traffic rate coming from suspicious ASes. We conducted several experiments using a real-world dataset to demonstrate the efficiency of our approach under DDoS attacks. Our results show that the proposed method can significantly reduce the impact of DDoS attacks with minimal overhead to routers. Moreover, the proposed anomaly detection techniques can help ASes to detect possible attacks and early attack detection by intermediate ASes.

References

[1] A. Y. Nur and M. E. Tozal, "Record Route IP traceback: Combating DoS Attacks and the Variants", Computers & Security 72 (2018): 13-25

[2] J. C. S. Lui, F. Liang, and Y. Yam, "Defending Against Distributed Denial-of-Service Attacks with Max-Min Fair Server-Centric Router Throttles." IEEE/ACM Transactions on Networking 13.1 (2005): 29-42.

[3] A. Yaar, A. Perrig, D. Song, "FIT: Fast Internet Traceback," IEEE INFOCOM, 2005

[4] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems." ACM Computing Surveys, 2007

[5] M. T. Gil and M. Poletto. "MULTOPS: A Data-Structure for Bandwidth Attack Detection." USENIX, 2001

[6] T. Peng, C. Leckie, and K. Ramamohanarao, "Proactively Detecting Distributed Denial of Service Attacks Using Source IP Address Monitoring", International Conference on Research in Networking, Springer, 2004

[7] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Network support for IP traceback", IEEE/ACM Transactions on Networking 9.3 (2001): 226-237

[8] A. Y. Nur and M. E. Tozal, "Defending Cyber-Physical Systems Against DoS Attacks", IEEE SMARTCOMP, 2016

[9] S. T. Zargar, J. Joshi, and D. Tipper, "A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks." IEEE Communications Surveys & Tutorials, 2013

[10] O. Osanaiye, K. K. R. Choo, and M. Dlodlo, "Distributed Denial of Service (DDoS) Resilience in Cloud: Review and Conceptual Cloud DDoS Mitigation Framework", Journal of Network and Computer Applications 67 (2016): 147-165.

[11] S. Yu, Y. Tian, S. Guo, and D. O. Wu, "Can we beat DDoS attacks in clouds?", IEEE Transactions on Parallel and Distributed Systems 25.9 (2013): 2245-2254.

[12] N. Levy, D. Smith, and J. Schiel, "Operationalizing ISP cooperation during DDoS attacks", NANOG, 2017

[13] CAIDA IPv4 Prefix-Probing Traceroute Dataset - 2020/02/02, https://www.ImpactCyberTrust.org, DOI 10.23721/107/1354205

[14] Routeviews Prefix to AS mappings Dataset for IPv4 and IPv6 - 2020/02/02, https://www.caida.org/data/routing/routeviews-prefix2as.xml

[15] Computing TCP's Retransmission Timer - RFC6298 - https://tools.ietf.org/html/rfc6298

[16] 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 32.3 (2002): 62-73.

[17] K. Malialis and D. Daniel, "Distributed response to network intrusions using multiagent reinforcement learning", Engineering Applications of Artificial Intelligence, 2015

[18] S. Xia, S. Guo, W. Bai, J. Qiu, H. Wei, and Z. Pan, "A New Smart Router-Throttling Method to Mitigate DDoS Attacks", IEEE Access 7 (2019): 107952-107963.

[19] L. Shi, M. Zhang, J. Li, and P. Reiher, "PathFinder: Capturing DDoS Traffic Footprints on the Internet", IEEE IFIP Networking Conference, 2018

[20] Y. Yong and X. Lin, "Method and system for user network behavioural based anomaly detection", U.S. Patent Application No. 11/644,993

[21] J. Peng, K. R. Choo, and H. Ashman, "User profiling in intrusion detection: A review", Journal of Network and Computer Applications 72 (2016): 14-27.

[22] B. Chandrasekaran, "Survey of network traffic models", Washington University in St. Louis CSE 567 (2009)

[23] C. Jin, Q. Chen, and S. Jamin, "Inet: Internet Topology Generator", (2000)