Poster: SDN-based System to Filter Out DRDoS Amplification Traffic in ISP Networks

Priyanka Dodia
Qatar Computing Research Institute, HBKU
Doha, Qatar
pgdodia@hbku.edu.qa

Yury Zhauniarovich
Perfect Equanimity
Minsk, Belarus
yury@perfectequanimity.com

ABSTRACT

Distributed Reflected Denial of Service (DRDoS) attacks remain one of the most popular techniques to drain victim's network bandwidth. Despite the goal of disrupting network services of a particular victim, indirectly these attacks affect a large number of benign Internet citizens. In particular, the owners of services vulnerable to amplification have to waste their resources to process incoming requests. Moreover, the voluminous attack traffic generated as a result of the amplification lavishes Internet Service Provider (ISP) resources, bandwidth and money, causing Quality of Service (QoS) degradation and subscription fee increase for the customers.

In this work we demonstrate a Software Defined Networking (SDN) based system to filter out garbage traffic from an ISP network. We employ a special type of a honeypot developed to collect information about ongoing DRDoS attacks. The firewall rules derived from this data are used to block incoming amplification requests from reaching amplifiers located within the provider network rescuing vulnerable services from being abused. In its turn, this prevents garbage traffic generation saving ISP's money and improving QoS.

CCS CONCEPTS

• **Security and privacy → Denial-of-service attacks**; *Firewalls*;
• **Networks → Network architectures**; *Programmable networks*.

KEYWORDS

Amplification attacks; SDN; garbage traffic filtering; honeypot; ISP networks

ACM Reference Format:
Priyanka Dodia and Yury Zhauniarovich. 2019. Poster: SDN-based System to Filter Out DRDoS Amplification Traffic in ISP Networks. In *2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19), November 11–15, 2019, London, United Kingdom.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3319535.3363277

1 INTRODUCTION

Due to the low resource requirements from an attacker and a possibility to stay stealthy, DRDoS attacks remain a major threat for the Internet. In these attacks, crafted requests with spoofed IP addresses of a victim are sent to genuine machines called amplifiers, which return amplified responses that can be thousands times larger in size than the corresponding requests, completely exhausting victim's bandwidth. For instance, a well-known software development platform called Github has experienced such attack of 1.35 Tb/sec [7] in February, 2018.

The most effective way to combat with DRDoS attacks globally is to apply ingress filtering of the traffic allowing only the packets with valid source IP addresses to pass. Such recommendations have been provided in RFC 2827 [13], also known as Best Current Practice No. 38 (BCP 38). However, ISPs are not incentivized to apply these recommendations. This best practice requires an ISP to spend its resources on traffic filtering, although this does not protect the ISP and its customers from the DRDoS traffic. ISPs also ignore the problem of garbage traffic generated by amplifiers located in their network. In this case, an ISP simply charges the owners of hosts vulnerable to amplification.

Still, indirectly ISPs suffer a lot from this unwanted traffic. First, it exhausts ISPs' and their customers' bandwidth affecting QoS. Second, last-mile ISPs usually buy bandwidth from upper tier providers paying for the traffic, especially, if it is asymmetric [2]. Indeed, the traffic generated by amplifiers hosted within the perimeter of an ISP network may reach substantial amounts. An ISP network may host hundreds or even thousands of such amplifiers that if abused, could potentially waste significant amount of ISP's resources and money. Third, processing of attack requests may affect negatively the performance of vulnerable hosts.

Contributions. In this demo, we describe the SDN-based implementation of the approach proposed in our paper [16]. Unlike the majority of the existing DRDoS mitigating solutions [11, 14, 15] that are focused on protecting victim's network, we aim at shielding amplifiers from illegitimate requests. Our prototype can be used to detect spoofed traffic and filter it out at the edge of an ISP network. This prevents garbage traffic being generated by amplifiers located within the network saving ISP's and its customers resources. Whilst the previous our paper concentrates on the idea and evaluates potential gains of ISPs, this work shows viability of our approach showing a real prototype of our system.

2 APPROACH DESCRIPTION

The large scale impact with even limited resources and the ease of launching make DRDoS attacks very popular among criminals. Figure 1 explains how this kind of attack works. An attacker (IP `198.51.100.1`) issues requests to the hosts running vulnerable protocols called amplifiers (in our case, a DNS server with IP `192.0.2.3` and an NTP server with IP `192.0.2.2`) with a spoofed source IP address (written in red) that corresponds to a victim

CCS '19, November 11–15, 2019, London, United Kingdom

ACM ISBN 978-1-4503-6747-9/19/11.
https://doi.org/10.1145/3319535.3363277

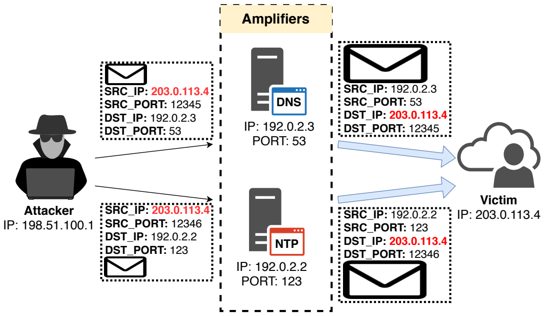

Figure 1: Amplification in DRDoS Attacks

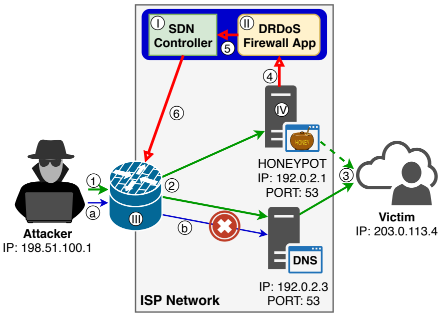

Figure 2: System Overview

(IP 203.0.113.4). Due to the existing functional vulnerabilities in these protocols, some types of requests (e.g., the *monlist* command in NTP) may generate considerably larger responses. Upon receiving this kind of request, the service replies with a much bigger response to the spoofed IP address of the victim. The reflective nature of these attacks brings an additional benefit to criminals, allowing them to stay anonymous (attacker source IP address is not exposed).

A network of an ISP provider may contain hundreds or even thousands of hosts vulnerable to amplification attacks. If being abused, collectively they can generate a huge amount of unnecessary traffic. As *garbage*, we consider the amplified traffic that flows from amplifiers to victims (bold blue arrows in Figure 1). According to our evaluation [16], *a single host vulnerable to NTP amplification could generate more than 2 TB of garbage traffic a day*).

Figure 2 describes our approach. There are four main components in our system (marked with Roman numerals): (I) SDN Controller; (II) DRDoS Firewall Application; (III) SDN Forwarding Device; and (IV) Amplification Honeypot. All incoming traffic to the ISP network passes through the *SDN Forwarding Device*. This edge device plays the role of a firewall, blocking the flows that match filtering rules. These rules are generated by the *DRDoS Firewall Application* based on the data provided by the *Amplification Honeypot*. The DRDoS Firewall Application uses the *SDN Controller* to deploy the rules on the SDN Forwarding Device.

Our system operates in the following way. During a preliminarily scan, an attacker discovers two hosts in ISP's network vulnerable to the DNS amplification attack: an open resolver with IP 192.0.2.3 and our amplification honeypot with IP 192.0.2.1 (see Figure 2). Later, the adversary uses the discovered hosts to launch an attack to the victim (IP 203.0.113.4). During the attack, the adversary sends request packets to the discovered vulnerable hosts targeting the vulnerable protocol running on a predefined UDP port (e.g., port 53 for DNS), changing source address to the victim IP (see green arrows of Steps 1 and 2). The vulnerable host and the honeypot replies to the victim IP with the amplified responses. However, the honeypot also starts monitoring the traffic heading to the victim IP. If its volume exceeds a predefined threshold, the honeypot sends an alert to the DRDoS Firewall Application (Step 4). The application through the SDN Controller (Step 5) issues an OpenFlow firewall rule to the SDN Forwarding Device (Step 6) that blocks all incoming packets with the source address and destination port matching the victim IP and the DNS port correspondingly. Hence, all consecutive requests from the attacker (Step a, blue arrow) will be blocked by the edge device, will not reach the vulnerable hosts (Step b) and will not be amplified.

The DRDoS Firewall Application installs a static rule on the SDN Forwarding Device that prevents blocking the traffic coming to the Amplification Honeypot (this "proactive" rule is set with a higher priority than the ones issued by the DRDoS Firewall Application). This allows the honeypot to continue receiving attack packets to detect the moment when the attack is over. Then, the honeypot notifies the DRDoS Firewall Application to drop the corresponding firewall rule. Despite being whitelisted, the honeypot mildly participates in the attack (and only in the beginning) due to its internal rate limiting mechanisms, therefore its impact to the victim is minimal.

3 PROTOTYPE DETAILS

We implemented the prototype of our system using the GNS3 network simulator [1]. GNS3 supports multiple emulators including Dynamips; Qemu, Virtualbox and VMWare virtual machines (VMs), however, in this work, we extensively rely on Docker containers. Docker containers run on the same host kernel, thus consuming considerably less system resources than traditional VMs. This allows researchers to increase considerably the number of emulated devices. In this work, we build separate Docker images for the SDN Controller (and our DRDoS Firewall Application); the Amplification Honeypot; a vulnerable host; attacker and victim machines; and the SDN Forwarding Device.

Figure 3 presents a screenshot of our GNS3 test bed. Router *RS* plays the role of network interconnection point with the installed routing rules that connect ISP (10.1.x.x), attacker (10.2.x.x) and victim (10.3.x.x) networks. So as our test bed is connected to the Internet as well, we use private IP address ranges for these networks. The connection to the Internet is required to perform the initial setup of the components, e.g., to download Docker images and Python packages necessary for our system to operate. Open vSwitch (10.1.0.1) plays the role of an ISP edge device that guards the network where the amplifiers (the honeypot with IP 10.1.0.4 and vulnerable hosts with IP 10.1.0.5) are located.

We use POX [4] as the SDN Controller. POX is widely adopted by researchers in the SDN community, and it implements OpenFlow 1.0 specification [12] that is forward compatible with newer OpenFlow standards. The DRDoS Firewall Application runs over POX and

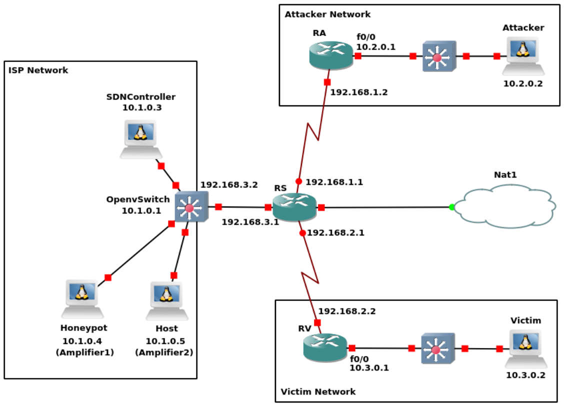

Figure 3: GNS3 System Simulation Setup

uses its API. Both of them are just programs that run on the same host (blue rectangle in Figure 2).

We use *AmpPot* [8] as the Amplification Honeypot that is developed by Lukas Krämer et al. It is widely adopted by the research community for the analysis of DRDoS attacks [5, 6, 9, 10]. Contrary to usual honeypots that act as an easy and attractive attack target, AmpPot mimics a service vulnerable to amplification attack. For instance, in Figure 1 AmpPot runs vulnerable DNS and NTP services. Thus, AmpPot is able to collect the information about amplification requests and to identify victims' IP addresses. It should be noted that AmpPot carefully participates in the attacks through a number of limiting mechanisms. As a result, a victim receives negligible amplified traffic from AmpPot and only in the beginning of the attack. Currently, AmpPot is able to monitor 10 different UDP services vulnerable to amplification. However, it can be easily extended to new ones.

Upon receiving an attack start event from our modified AmpPot, the DRDoS Firewall Application using POX OpenFlow-compatible Southbound API adds a rule to OpenFlow's Open vSwitch [3], which we use as the SDN Forwarding Device. This rule blocks all packets with the specified source IP address and destination port, derived from the honeypot data. It should be mentioned that adversaries often attack a whole subnetwork rather than a single IP address. In order to cover this case, we added a feature to our DRDoS Firewall Application to block an entire subnetwork. This functionality relies on a IP address partial match feature implemented in POX.

4 DEMO DETAILS

During the presentation, we are going to show an interactive demonstration of our system that can help Internet Service Providers to filter garbage DRDoS traffic out of their networks. Using the test bed shown in Figure 3, we will guide the audience through all the steps of garbage traffic removal. We will start describing the main network segments in our test bed, namely the ISP, Attacker and Victim subnetworks, and the main components within them. We will provide details about the initial configuration steps as well. Then, we will simulate a DRDoS attack in our test bed sending amplification requests from Attacker's machine to the amplifiers in the ISP Network (Host and Honeypot) with the spoofed IP address of victim. We will show that in the beginning the amplified responses will leave the ISP network and will be detected on the Victim's side. However, after the Honeypot observes from a particular IP address more packets than the predefined threshold, it issues an alert to the DRDoS Firewall Application, and the amplified responses will stop reaching victim's machine. We will also demonstrate that amplification requests are still observable by AmpPot due to a special static exclusion rule proactively added to the SDN rule table. Once the attack is over, we will exhibit how the DRDoS Firewall Application instructs the SDN Controller to remove the corresponding rule from the edge SDN Forwarding Device.

5 CONCLUSION

In this paper, we demonstrate a novel approach to filter out amplification traffic from an ISP network. It relies on data collected from an amplification honeypot to derive filtering rules. In the prototype of our system, we employ the SDN paradigm, however, other ways of packet filtering, for instance, a traditional firewall or a BGP Flowspec, can be used in our approach as well.

REFERENCES

[1] [n.d.]. *GNS3 | The Software that Empowers Network Professionals.* https://www.gns3.com

[2] [n.d.]. *How UK ISPs are charged for broadband - the cost of IPStream.* Retrieved 03/20/2018 from https://community.plus.net/t5/Plusnet-Blogs/How-UK-ISPs-are-charged-for-broadband-the-cost-of-IPStream/ba-p/1314570

[3] [n.d.]. *Open vSwitch.* https://www.openvswitch.org

[4] [n.d.]. *The POX Network Software Platform.* https://github.com/noxrepo/pox

[5] Michael Aupetit, Yury Zhauniarovich, Giorgos Vasiliadis, Marc Dacier, and Yazan Boshmaf. 2016. Visualization of Actionable Knowledge to Mitigate DRDoS Attacks. In *Proceedings of the IEEE Symposium on Visualization for Cyber Security.* 1–8.

[6] Laure Berti-Equille and Yury Zhauniarovich. 2017. Profiling DRDoS Attacks with Data Analytics Pipeline. In *Proceedings of the ACM Conference on Information and Knowledge Management.* 1983–1986.

[7] Sam Kottler. 2018. *February 28th DDoS Incident Report.* https://githubengineering.com/ddos-incident-report/

[8] Lukas Krämer, Johannes Krupp, Daisuke Makita, Tomomi Nishizoe, Takashi Koide, katsunari Yoshioka, and Christian Rossow. 2015. AmpPot: Monitoring and Defending Against Amplification DDoS Attacks. In *Proceedings of the International Symposium Research in Attacks, Intrusions, and Defenses.* 615–636.

[9] Johannes Krupp, Michael Backes, and Christian Rossow. 2016. Identifying the Scan and Attack Infrastructures Behind Amplification DDoS Attacks. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security.* 1426–1437.

[10] Johannes Krupp, Mohammad Karami, Christian Rossow, Damon McCoy, and Michael Backes. 2017. Linking Amplification DDoS Attacks to Booter Services. In *Proceedings of the International Symposium Research in Attacks, Intrusions, and Defenses.* 427–449.

[11] Jun Li, Skyler Berg, Mingwei Zhang, Peter Reiher, and Tao Wei. 2014. Drawbridge: Software-defined DDoS-resistant Traffic Engineering. In *Proceedings of the ACM Conference on SIGCOMM.* 591–592.

[12] N. McKeown, H. Balakrishnan T. Anderson, L. Peterson G. Parulkar, S. Shenker J. Rexford, and J. Turner. 2008. OpenFlow: Enabling Innovation in Campus Networks. *ACM SIGCOMM Computer Communication Review* 38, 2 (2008), 69–74.

[13] D. Senie P. Ferguson. 2000. *Network Ingress Filtering: Defeating Denial of Service Attacks which Employ IP Source Address Spoofing.* Technical Report. Internet Engineering Task Force. https://tools.ietf.org/html/rfc2827

[14] Rishikesh Sahay, Gregory Blanc, Zonghua Zhang, and Hervé Debar. 2015. Towards autonomic DDoS mitigation using software defined networking. In *Proceedings of the NDSS Workshop on Security of Emerging Networking Technologies.* Internet society.

[15] A. Sardana, K. Kumar, and R. C. Joshi. 2007. Detection and Honeypot Based Redirection to Counter DDoS Attacks in ISP Domain. In *Proceedings of the International Symposium on Information Assurance and Security.* 191–196.

[16] Yury Zhauniarovich and Priyanka Dodia. 2019. Sorting the Garbage: Filtering Out DRDoS Amplification Traffic in ISP Networks. In *Proceedings of the IEEE Conference on Network Softwarization.* 142–150.