UNIVERSITY OF LATVIA
FACULTY OF COMPUTING

Artūrs Lavrenovs

MEASURING DISTRIBUTED REFLECTED DENIAL-OF-SERVICE AMPLIFIED VOLUMETRIC ATTACK CAPACITY

Doctoral Thesis

Field: Computer Science and Informatics

Subfield: Data Processing Systems and Computer Networks

Scientific advisor: Dr.sc.comp., Prof. Guntis Bārzdiņš

RIGA 2023

ABSTRACT

In today's Internet distributed denial-of-service (DDoS) attacks play an ever-increasing role and possess a risk to any commercial or governmental entity with a presence on the Internet or anyone simply having an Internet connection. Meanwhile, there are no trustworthy estimates of the global DDoS attack capacity. To address this issue a novel methodology is proposed for the reflected amplified volumetric DDoS attack capacity measurement.

This thesis defines and implements the DDoS attack capacity measurement methodology which focuses on the individual reflectors and their properties, primarily, rate limiting, amplification, and bandwidth. This thesis presents a detailed analysis of the measured protocols and the respective calculated attack capacities: NTP – 43 Gbps, DNS – 27.5 Tbps, SSDP – 808 Gbps, SNMP – 2.47 Tbps, CLDAP – 870 Gbps. The global DDoS attack capacity for the analyzed protocols was calculated to be 31.33 Tbps. A range of new data views and visualizations based on the measurement data were developed to stimulate remediation discussion.

Limitations of the measurement methodology are identified and discussed. Comparison with the only existing alternative methodology which relies on the theoretical estimates instead of direct measurements revealed that both methodologies complement each other rather than compete. The primary advantage of protocol measurement is the ability to detect and estimate remediation.

Keywords: DDoS attack capacity, Internet measurement, Distributed denial-of-service attacks.

CONTENTS

INDEX OF TABLES

INDEX OF FIGURES

GLOSSARY OF TERMS, ACRONYMS, AND ABBREVIATIONS

Term, acronym, or abbreviation	Explanation
AD	Active Directory service by Microsoft (or compatible, i.e., Samba)
amplifier	A device providing network service on the Internet replying with significantly larger response than request, sometimes used interchangeably with reflector
AS	Autonomous System – a collection of IP prefixes under single administrative control
ASN	Autonomous System Number – unique numerical identifier of an AS, frequently used interchangeably with AS in the literature
BAF	Bandwidth Amplification Factor
BCP	Best Current Practice document of Internet Engineering Task Force
CDN	Content Delivery Network
CERT	Computer Emergency Response Team
CLDAP	Connectionless Lightweight Directory Access Protocol
CPE	Customer Premises Equipment – in the context of this thesis active devices with an assigned public IP address, e.g., gateways, modems, routers, TV set-top boxes
device	Embedded device or a server providing network service publicly reachable on the Internet. If running connectionless protocol then the device is a reflector, all reflectors are devices
DoS	Denial-of-service – a type of computing or network resource exhaustion attack
DDoS	Distributed denial-of-service – a type of network DoS attack that has many sources of the attack same time. Used interchangeably with DRDoS
DRDoS DrDoS	Distributed reflected denial-of-service – a type of DDoS attack that generates malicious network traffic using third-party network services (reflectors), usually also amplifying it (amplifiers). Sometimes in the literature abbreviated as rDDoS and therefore confused with RDDOS – Ransom DDoS
DNS	Domain Name System
ICMP	Internet Control Message Protocol
IoT	Internet of Things
IP IPv4 IPv6	Internet Protocol, occasionally shorthand for IP address Internet Protocol version 4 Internet Protocol version 6
IPS	Intrusion Prevention System
ISP	Internet Service Provider

Term, acronym, or abbreviation	Explanation
IXP	Internet eXchange Point – a physical location through which ISPs and CDNs interconnect
LAN	Local Area Network
Memcached	In-memory key-value store with a network capability.
mitigation	Activities causing a primarily temporary reduction in the DDoS attack capacity reaching victims, e.g., filtering network traffic at a victim, activating filtering in transit or client network based on a threshold
NAT	Network Address Translation
NTP	Network Time Protocol
OS	Operating System
packet	IP packet, without explicit context – IP packet with UDP payload of an abused protocol
PPS	Packets per Second
reflector	A device providing a network service that uses UDP protocol and can be abused to reflect responses to third-party (victims), sometimes used interchangeably with the amplifier
remediation	Activities causing a primarily permanent reduction in the total DDoS attack capacity, e.g., software and firmware updates, BCP38 implementation, removing or blocking reflectors
RPS	Requests per Second
RR	Response Rate (Ratio) – minimum ratio expressed in percentage of the received responses for the sent measurement requests, i.e., “RR 40%” includes all reflectors that have responded to 40%-100% of the sent requests
RRL	Response Rate Limiting
RTT	Round Trip Time
SNMP	Simple Network Management Protocol
SSDP	Simple Service Discovery Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VPN	Virtual Private Network

INTRODUCTION

Research background and motivation

Distributed denial-of-service (DDoS) attacks have been plaguing the Internet almost since its inception. The first-ever officially recorded denial-of-service (DoS) attack occurred in 1974 and was caused by a 13-year-old student David Dennis [1] which wasn't even a network DDoS attack everyone recognizes today. One of the first large-scale network DDoS attacks happened in 1999 against a computer network of the University of Minnesota by flooding it with UDP packets directly from many hosts [1]. DDoS attacks have become almost daily news and have created a large cybercrime industry offering DDoS attacks as a service as well as a huge cyber defense industry providing network filtering and attack mitigation services, software, and hardware solutions. A reasonable observer without any computer networking or cybersecurity background would assume that this issue has been and currently is being addressed properly to eliminate the issue at the root cause. The reality is that DDoS attacks have been on the rise with the increase of Internet connection speeds. Mitigation and remediation efforts have only slowed down the total growth of the attacks.

DoS attacks are still relevant to the extent of exploiting specific hardware or software vulnerabilities. These vulnerabilities can be exploited by crafting malicious payloads having a particular size or content and sending those to targeted victims using one or a few requests. Then the affected device enters the hang (broken) state and can't serve any further requests and fulfill internal functions, when the malicious request stream stops the hang state might persist until the device's power is physically reset. It usually affects embedded or IoT devices because of the ability to exploit memory caused by poor low abstraction level programming practices. It is uncommon in the public services on the Internet which are targeted by the attacks, most services are sufficiently protected from a single source DoS attacks. DDoS has become a catch-all term for all forms of resource exhaustion attacks usually enabled by network connectivity delivering attack traffic over the Internet. It operates on all 7 layers of the OSI model but does not necessarily exhaust network resources, it can exhaust also computing, memory, and storage resources of the victims.

DDoS attacks are not only affecting individually targeted services but also are one of the major causes of network outages [2]. Volumetric variant of DDoS attacks can exhaust the resources of the whole targeted network thus affecting all the connected services. [1.1.3. Reflected Amplified Volumetric DDoS attacks] is the main focus of this research, this DDoS attack relies on abusing third-party reflectors (figure 1, 2) by sending them requests containing the spoofed IP address of the victims to which reflectors are responding with the amplified responses. This attack specifically exhausts all the available network connection bandwidth of the victim thus preventing legitimate network traffic from being received.

Even unrelated networks can be affected and slowed down because of the large amount of attack traffic passing through transit routers. This creates the potential for a vast amount of collateral damage. It is not a problem of individual organizations anymore but has evolved into a national and even international issue. On this level of decision making only technical or

best practice solutions will not suffice. To address the issue properly it has to be based on hard facts and knowledge. Such comprehensive knowledge regarding total DDoS attack capacity and contributing factors is lacking. This is the main motivator for conducting current research and contributing towards both academia and solving the real-world problem at the same time.

A survey conducted by SANS Institute demonstrated that organizations are not addressing the DDoS issue sufficiently [3]. The DDoS attack landscape is volatile, new protocols are exploited while some of the old ones get remediated or mitigated but the maximum attack capacity is still growing. A DDoS mitigation solution that was set up even a year ago might not suffice anymore. Organization-level decision-makers would benefit as well from the source of information regarding the capacity of the DDoS attacks and changes occurring over time. It could be used as part of the decision-making process for new or upgraded solutions for DDoS mitigation.

Novelty of the research

The research presented in this thesis proposes a novel DDoS attack capacity measurement methodology applicable to the individual protocols abused for reflection and amplification. This methodology has been implemented and applied to the whole Internet and the most commonly abused protocol attack capacity contributions were calculated.

At the beginning of this research, there were no authoritative and trustworthy public sources providing the DDoS attack capacity estimates. Closest attack capacity estimation research [4] in simplified terms multiplied reflector counts with the average speed of the ISP.

Thesis and research questions

During the research following thesis was proposed:

- It is possible to estimate worldwide reflected amplified volumetric DDoS attack capacity with high confidence without having to rely on privileged information by measuring protocols and services present on publicly reachable devices on the Internet.

The following research questions were investigated:

- How to identify and scan for reflectors on the Internet that can be utilized for DDoS attacks?
- How to measure the properties of individual reflectors?
- What are the limitations that apply to individual reflectors?
- How to produce the total protocol attack capacity from the individual reflectors?
- Which protocols, countries, networks, and devices contribute the most to the attack capacity?
- What is the global DDoS attack capacity?
- How does the measured attack capacity compare with the existing theoretical methodology?

Research methods

The following research methods have been used in the doctoral thesis:

- A literature review was conducted to produce an overview of the DDoS research field focusing on the attack capacity.
- Quantitative measurement to identify individual reflectors and measure their properties was defined, developed, and executed in accordance with the best practice of the established research field of Internet Measurement.
- Quantitative data analysis focusing on data visualization was conducted for each of the measured protocols.
- Comparative analysis was conducted for the measured protocols and the alternative theoretical methodology results.

The scope, aim, and tasks of the research

The scope of this research is limited to a single type of DDoS attack – volumetric reflected amplified. Because of the nature of this attack, it is the most problematic to mitigate while maintaining attacked service availability. Whenever the acronym DDoS is used outside of the context describing different types of attacks, the author refers to the volumetric reflected amplified DDoS attacks. Only the Internet functioning over IPv4 protocol is defined within the research scope as IPv6 introduces additional challenges to the scanning while contributing little to the overall attack capacity that is discussed in [1.4. DDoS over IPv6].

The aim of this research is to provide an improved DDoS attack capacity measurement methodology. To achieve this aim the following tasks are outlined: review literature and industry sources related to the DDoS capacity, define DDoS attack capacity measurement methodology, implement the defined methodology for the commonly abused protocols, analyze and visualize the produced measurement results, review limitations of the methodology, explore the abused device classification possibility, compare the methodology and produced results with the existing alternatives, analyze the applicability of the results to address the persisting attack capacity issue.

Approbation and main results of the thesis

The results of this thesis and doctoral studies are presented in the following academic conferences and published in the conference proceedings, by April 2023 12 publications were indexed by Scopus and 9 by Web of Science:

1. G. Visky, A. Lavrenovs, E. Orye, D. Heering, and K. Tam, “Multi-Purpose Cyber Environment for Maritime Sector,” *iccws*, vol. 17, no. 1, pp. 349–357, Mar. 2022, doi: 10.34190/iccws.17.1.26 [5] presented by a co-author at the 17th International Conference on Information Warfare and Security, USA, on March 17-18, 2022.
2. A. Lavrenovs, E. Leverett, and A. Kaplan, “The tragedy of common bandwidth: rDDoS,” in *New Security Paradigms Workshop*, Virtual Event USA, Oct. 2021, pp. 43–58. doi: 10.1145/3498891.3500928 [6] presented by the author at the 2021 New Security Paradigms Workshop, USA, on October 26-28, 2021.
3. G. Visky, A. Lavrenovs, and O. Maennel, “Status Detector for Fuzzing-Based Vulnerability Mining of IEC 61850 Protocol,” in *Proceedings of the European Conference on Information Warfare and Security*, 2021. doi: 10.34190/EWS.21.007

[7] presented by a co-author at the 20th European Conference on Cyber Warfare and Security, UK, on June 24-25, 2021.

4. A. Lavrenovs and R. Graf, “Explainable AI for Classifying Devices on the Internet,” in *2021 13th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia, May 2021, pp. 291–308. doi: 10.23919/CyCon51939.2021.9467804 [8] presented by the author at the 13th International Conference on Cyber Conflict, Estonia, on May 25-28, 2021.
5. R. Meier, A. Lavrenovs, K. Heinaaro, L. Gambazzi, and V. Lenders, “Towards an AI-powered Player in Cyber Defence Exercises,” in *2021 13th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia, May 2021, pp. 309–326. doi: 10.23919/CyCon51939.2021.9467801 [9] presented by a co-author at the 13th International Conference on Cyber Conflict, Estonia, on May 25-28, 2021.
6. A. Lavrenovs, “Towards Remediating DDoS Attacks,” in *Proceedings of the International Conference on Cyber Warfare and Security*, TN, USA, 2021. doi: 10.34190/IWS.21.046 [10] presented by the author at the 16th International Conference on Cyber Warfare and Security, USA, on February 25-26, 2021.
7. L. Bortnik and A. Lavrenovs, “Android Dumpsys Analysis to Indicate Driver Distraction,” in *Digital Forensics and Cyber Crime. Proceedings of the 11th EAI International Conference, ICDF2C 2020, Boston, MA, USA, October 15-16, 2020.*, vol. 351, Cham: Springer International Publishing, 2021, pp. 139–163. doi: 10.1007/978-3-030-68734-2_8 [11] presented by a co-author at the 11th EAI International Conference on Digital Forensics & Cyber Crime, USA, on October 15-16, 2020.
8. A. Lavrenovs, K. Heinäaro, and E. Orye, “Towards Cyber Sensing: Venturing Beyond Traditional Security Events,” in *Proceedings of the 19th European Conference on Cyber Warfare*, Chester, UK, Jun. 2020. doi: 10.34190/EWS.20.062 [12] presented by the author at the 19th European Conference on Cyber Warfare, UK, on June 25-26, 2020.
9. A. Lavrenovs, R. Graf, and K. Heinaaro, “Towards Classifying Devices on the Internet Using Artificial Intelligence,” in *2020 12th International Conference on Cyber Conflict (CyCon)*, Estonia, May 2020, pp. 309–325. doi: 10.23919/CyCon49761.2020.9131713 [13] presented by the author at the 12th International Conference on Cyber Conflict, Estonia, on May 26-29, 2020.
10. A. Lavrenovs and G. Visky, “Investigating HTTP response headers for the classification of devices on the Internet,” presented at the 2019 IEEE 7th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Liepaja, Latvia, Nov. 2019. doi: 10.1109/AIEEE48629.2019.8977115 [14] presented by the author at the 7th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering, Latvia, on November 15-16, 2019.
11. A. Lavrenovs and G. Visky, “Exploring features of HTTP responses for the classification of devices on the Internet,” presented at the 2019 27th Telecommunications Forum (TELFOR), Belgrade, Serbia, Nov. 2019. doi: https://doi.org/10.1109/TELFOR48224.2019.8971100 [15] presented by the author at the 27th Telecommunications Forum, Serbia, on November 26-27, 2019.

12. A. Lavrenovs, “Towards Measuring Global DDoS Attack Capacity,” in *2019 11th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia, May 2019, pp. 1–15. doi: 10.23919/CYCON.2019.8756851 [16] presented by the author at the 11th International Conference on Cyber Conflict, Estonia, on May 28-31, 2019.
13. K. Podins and A. Lavrenovs, “Security Implications of Using Third-Party Resources in the World Wide Web,” in *2018 IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE)*, Vilnius, Nov. 2018, pp. 1–6. doi: 10.1109/AIEEE.2018.8592057 [17] presented by a co-author at the 6th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering, Lithuania, on November 8-10, 2018.
14. A. Lavrenovs and F. J. R. Melon, “HTTP security headers analysis of top one million websites,” in *10th International Conference on Cyber Conflict, CyCon 2018, Tallinn, Estonia, May 29 - June 1, 2018*, 2018, pp. 345–370. doi: 10.23919/CYCON.2018.8405025 [18] presented by a co-author at the 10th International Conference on Cyber Conflict, Estonia, on May 29-June 1, 2018.
15. A. Lavrenovs and K. Podins, “Privacy violations in Riga open data public transport system,” in *2016 IEEE 4th Workshop on Advances in Information, Electronic and Electrical Engineering, AIEEE 2016 - Proceedings*, Vilnius, Lithuania, Nov. 2016, pp. 1–6. doi: 10.1109/AIEEE.2016.7821808 [19] presented by the author at the 4th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering, Lithuania, on November 10-12, 2016.

All of the listed papers are publicly accessible in either final or preprint versions here: https://www.researchgate.net/profile/Arturs-Lavrenovs/research.

The main results of the research include:

- Defined, implemented, and real-world approbated DDoS attack capacity measurement methodology.
- New absolute and relative data presentations for the remediation and mitigation discussion and decision-making.
- Detailed analysis of the measured protocols and the respective calculated capacities: NTP – 43 Gbps, DNS – 27.5 Tbps, SSDP – 808 Gbps, SNMP – 2.47 Tbps, CLDAP – 870 Gbps.
- Global DDoS attack capacity calculated to be 31.33 Tbps.

Disclaimer

This doctoral thesis has been used as the source for the published academic papers listed in [Approbation and main results of the thesis] and vice versa. By default, this thesis is expected to be treated as the primary source and it does contain unmodified full sentences, paragraphs, and graphics from the listed published papers without references when fully authored by only the author himself. Textual and graphical parts of the published papers that were created by or together with co-authors are always appropriately referenced as any other reference. This doctoral thesis purposely minimizes and excludes most of the published papers where the author didn’t serve in the role of primary researcher.

Outline of the thesis

The main body of this doctoral thesis consists of 6 chapters. [1. DDoS attack and research overview] explores different types of DDoS attacks, academic research investigating the attacks and methodology, information and knowledge provided by the industry, and the current DDoS estimates. [2. Attack capacity measurement methodology] provides the proposed methodology overview and details every step in it. [3. Classifying devices] explores possibilities of classifying sets of identified abusable devices for analysis and remediation purposes and future research required to be conducted. In [4. Protocol measurements] the methodology proposed in [2. Attack capacity measurement methodology] is being applied to the individual services and protocols which are abusable for the DDoS attacks, individual protocol contributions are explored in detail, implementation and data quality issues are discussed. [5. Global attack capacity] brings together protocols analyzed in [4. Protocol measurements] and reviews trends for a one-year period comparing with the theoretical capacity, methodology limitations are explored. In [6. Remediating DDoS attacks] possible remediation and relevant actors are being discussed.

1. DDOS ATTACK AND RESEARCH OVERVIEW

1.1. Types of DDoS attacks

A detailed taxonomy of DDoS attacks and defense mechanisms is provided in academic literature and updated every few years [20], [21]. There is no reason to repeat the work done thus information in this thesis pertaining to this matter is presented in as limited fashion as possible while still providing context for the research being done.

Multiple ways have been proposed how to classify DDoS attacks in the industry and academic literature. Zargar et al. categorized attacks being either network/transport-level DDoS flooding attacks or application-level DDoS flooding attacks, further creating disparate subcategories to accommodate each type of attack [21]. Industry prefers to split off volumetric attacks into separate category to signify their importance and different requirements for mitigation. Cybersecurity and DDoS mitigation company Imperva categorizes DDoS attacks into being either volume based, protocol, or application layer [22]. Overview of DDoS attack types are presented in three categories – application layer attacks, protocol and network attacks, and reflected amplified volumetric DDoS attacks. Non-reflected volumetric attacks are discussed in the protocol and network category, this separation is caused by the prevalence of reflected amplified volumetric DDoS attacks and them being the focus of this research.

1.1.1. Application layer DDoS attacks

Applications always have less computational resources assigned to them than is available to the OS itself. Applications are far less efficient and optimized than OS network stack thus for attackers it can be easier to interrupt the service by attacking the application itself. One drawback for the attacker is that the attack software has to be adapted for specific applications and respective requests as opposed to volumetric or protocol level DDoS attacks which interrupt access to any kind of service or application running.

Websites relying on HTTP protocol are the most targeted ones for the application DDoS attacks but the principles across different protocols are the same. In the case of TCP, protocol attacks must come directly from an attacking system usually a botnet containing many compromised systems, malicious requests can be indistinguishable from the legitimate clients [23]. The most common type of application DDoS attack is flooding the application with as many requests as possible. If no protection mechanism is enabled practically no application will be able to handle even a single attacker system. By default, most server OS and application setups do not have any flooding protection as with the generic software actual use case is known only to the system owner. This type of attack is mitigatable on the network or system log monitoring layer, if all of the attacker’s bots are sending an abnormally high number of requests when individual bots reach the threshold they get blocked, in many cases after some initial attack period enough bots get automatically blocked so legitimate clients can continue using service. The blocking happens on the network level so it is efficient compared to blocking on the application level. But if the botnet is large enough and with enough

dynamic IP addresses, then flooding can cause prolonged slowdowns and inaccessibility for legitimate users.

Instead of overwhelming applications with continuous streams of requests which can be blocked by analyzing number of the requests coming from a single IP address, it is possible to abuse slow and long-term requests and responses while remaining under the blocking threshold. Applications tend to have some functionality that is slow because it is resource intensive or waits on I/O, if attackers research an application and time the responses they can send requests targeting specific slow functionality [24]. The other approach for the attacker is to be slow on the requests or reading responses to keep them alive as long as possible, one of the most prominent this type of attack is Slowloris [25]. The attacker sends request headers as slowly as possible thus overfilling the web server's active connection handling capacity, this attack has been so successful that the same approach was adopted to POST data sending attacks and even response reading slowdown [25].

With the rise of CDN prominence and server software optimizations, this type of DDoS has become less effective and prevalent. Even if the attack is successful the first time, the victim can take preparatory steps to mitigate future attacks on their system or purchase a CDN service without investing an excessive amount of resources. Simultaneously attacks are evolving as well, even when it is unfeasible to take the service down, an attacker might cause economic damage by trying to use up as much as possible of the victim's bandwidth, computational or request resources. Some CDN providers charge for those even if it is obviously a DDoS attack. This type of attack is called Economic Denial of Sustainability (EDoS), the attacker can use low-cost resources or even free ones to send victims legitimate requests that require more bandwidth and resources to respond than the attacker spent, this type of the attack is investigated by Wang et al. [26]. Researchers demonstrated that free online services, e.g., Facebook, have the functionality to request remote resources which are triggered by a user, attacker can create a large set of fake accounts to trigger that functionality thus launching reflected attacks which because of the design of large online systems is likely to be also distributed. The only way how to mitigate this type of attack is to verify that user requests are coming from an actual human, which can decrease the usability of the system and user experience. It was concluded that an attacker with a single IP address and computer can cause thousands of dollars in economic damage.

1.1.2. Protocol and network DDoS attacks

Abusing network protocols permits an attacker to exhaust the victim's computational resources without exhausting the victim's network bandwidth. Stateful network protocols and implementations are primary targets for the abuse as the victim has to keep the state of the communication in memory and process it. Possibly the most abused protocol is TCP, it has multiple features settable by the sender that affect the connection state and can be abused. Connection establishing is commonly abused, TCP requires a three-way handshake to establish a new connection, state of the connection is stored in OS memory buffers until timeout thus an attacker can send new connection-establishing SYN packets from many spoofed IP addresses (commonly called SYN flood) and exhaust victims resources for handling new legitimate user connections [27]. With the redesign of the OS network stack, drivers, and other components, and the growth of computational power and proliferation of

CDN technologies these attacks have become less effective. But Cloudflare still reports SYN being the most popular attack vector in 2022, while TCP RST and ACK packets are also being actively utilized [28].

Reflection attacks initially have been used without significant amplification. A common historic reflection attack using ICMP protocol is called the Smurf attack, it is triggered by sending an ICMP echo request to a broadcast IP address with the victim's spoofed IP address as the source, where all the generated ICMP echo replies target the victim [29]. These types of attacks are less efficient and prevalent nowadays as with the exactly same setup and resources an attacker can cause significantly larger reflected amplified attacks.

A volumetric DDoS attack (sometimes called a flooding attack) means the network bandwidth of the victim is being depleted by the attacker. Originally when these attacks first started the bandwidth came directly from systems under the control of the attacker, where the total capacity of the attacker had to exceed the bandwidth available to the victim. It was easy to determine attacking systems and take action against them on the network level further away from the victim and in the physical world by informing network owners and law enforcement agencies. With the rise of fame of reflected amplified attacks direct volumetric attacks became far less prevalent as attackers saw much higher bandwidth and effectiveness from amplified attacks while still hiding sources of infected machines under their control. One notable exception is the Mirai botnet composed of IoT devices that were able to execute different types of DDoS attacks, the direct volumetric attack was reported to be 623 Gbps on September 21, 2016, the largest publicly reported at that time [30]. It might be associated with the large count of compromised devices, providing high bandwidth and same time small loss if some devices get remediated.

1.1.3. Reflected Amplified Volumetric DDoS attacks

Reflected Amplified Volumetric DDoS attacks are the most problematic type of DDoS attacks. To mitigate it a defender has to absorb and process all the received network traffic by separating valid from the attack packets. Computationally it consumes few resources to filter out attack traffic by analyzing just network protocol and port as long as the network is not hosting some of the abused services. Otherwise, much more detailed processing is needed. The main issue is the bandwidth capacity of the attacked network. It is limited not only by contractual relations between ISP and the attacked network but also by chosen technology and network hardware.

It is prohibitively expensive for organizations to build and maintain their own networks that can receive and mitigate the attack sizes that are common today. A network connection that exceeds planned attack capacity has to be purchased as a service from an ISP. IP transit is possibly the cheapest way how to acquire large-capacity Internet connection directly from Tier I or II network service providers, prices of IP transit vary significantly even by order of magnitude depending on the service provider, location, contractual period, and conditions. Because of these factors and their commercial nature, real IP transit pricing is not public knowledge. Hurricane Electric in 2018 has advertised costs as low as 0.15 USD per Mbps on large contractual commitment [31], it is possibly the lowest price on the market or close to it. In the best case scenario to be able to receive a 100 Gbps DDoS attack, monthly IP transit bill alone would be 15,000 USD. It doesn't take into account the one-time costs of acquiring

network equipment and monthly support. Even then maximum attack capacity reported by the industry (table 1) far exceeds 100 Gbps, which means larger attacks would still bring the protected network down. This type of attack defense cost is disproportionately high compared to the attack cost.

There are two main causes for this type of attack – the ability to spoof IP addresses and network services that use UDP protocol and can produce responses significantly larger than the received requests. Both of these components are mandatory for this type of attack and proper addressing of any one of those would remediate the issue.

Spoofing issues have been addressed in the industry best practice recommendations BCP 38 [32], which states that a network should not accept packets for forwarding that did not originate in that network. Filtering can be done at the router closest to the client by hardcoding allowed ranges or close to the source at upstream ISP and IP transit provider level where it can be filtered automatically by verifying that AS of the network is announcing the IP ranges via BGP which are used as the source IP address of the packet, different filtering strategies are described in BCP 84 [33]. This requires some initial setup and possibly continuous maintenance from the network administrator, which is why there is a significant amount of networks that neglect it.

The Internet relies on many underlying network services, e.g., DNS for easy-to-remember domains, and NTP for time synchronization. Historically for speed and simplicity, these services were implemented using the stateless protocol UDP, without realizing to what extent it can be abused. Besides these useful services, there are some specialized services with limited value to the general public. In all cases, some services are misconfigured, primarily in a way that internal services become publicly accessible from the Internet by anyone. In total, there are millions of devices that are running some UDP network service, which shouldn’t be publicly accessible. An additional issue is that some of these services can produce responses much larger than the request thus an attacker can cause with its limited bandwidth significantly larger attack – amplified being reflected from these services.

Figure 1 presents a common way how academic, industry and other sources portray DDoS attacks. It has only 3 components – Bots (controlled by the botmaster, not displayed), Reflectors, and the Victim. Bots send requests for large responses (larger than requests) to the Reflectors using the spoofed IP address of the Victim. Reflectors send large responses to the Victim thus overloading its network connection so it can't serve legitimate clients. This representation is oversimplified!

A more appropriate volumetric reflected amplified DDoS attack visualization is presented in figure 2, it contains all the concepts relevant to this attack type. IP packets are simplified and contain only payload, source (SRC), and destination (DST) pseudo IP addresses. There is the Internet and five networks connected to it, all the packets are properly routed.

The client with the IP address 1.1 is requesting some useful service from a Reflector with IP 2.2. Reflector with IP 2.2 sends the response to the source address IP 1.1 and the Client successfully receives it. In this scenario, the reflector is just a server that is running a stateless network protocol (UDP) based network service that can be abused for amplification and reflection thus the attacks are classified as reflected.

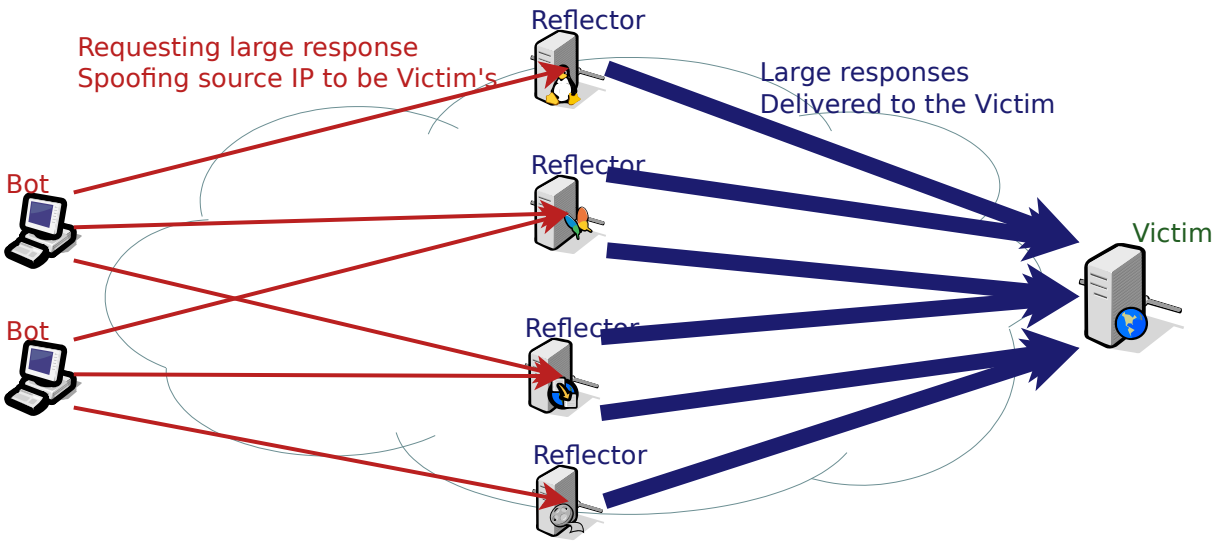

Figure 1: Common way how reflected DDoS attacks are presented

There is a botnet consisting of two compromised Client machines with IP addresses 1.2 and 3.3. When a botmaster (not displayed in the illustration) initiates the attack against the victim with IP 4.4, bots try to send as many requests as possible the with spoofed IP address of the victim to the reflectors. These requests on purpose are containing payloads that will generate significantly larger responses than requests thus amplifying the attack. Network clients are generating IP packets fully themselves, proper clients should generate proper IP packets with IP addresses assigned to them. Even if clients are misconfigured or compromised properly configured and maintained network should detect and block that, like in the case of ISP 1 which drops spoofed packets coming from its connected Client with IP 1.2.

On the contrary ISP 3 is a mismanaged network and doesn't stop packets with spoofed addresses from leaving its network. After the packet has left its originating network it becomes unviable to filter it later on down the path. Spoofed packets contain SRC IP 4.4 and reflectors will use it as DST addresses where the response is being sent. After the attack started victim's network connection was overloaded and the legitimate request coming from the Client with IP 1.3 doesn't reach the victim but is dropped somewhere along the path possibly at the victim's ISP 4 router. As the victim's network connection was overloaded the attack is classified as volumetric.

Although caused by a single bot multiple reflectors are participating in the attack thus it is classified as a Distributed denial-of-service attack. Because of the amplification one bot with a slow residential network connection can cause an attack that is hundreds or thousands of times larger than its network connection thus potentially overloading even enterprise network connection.

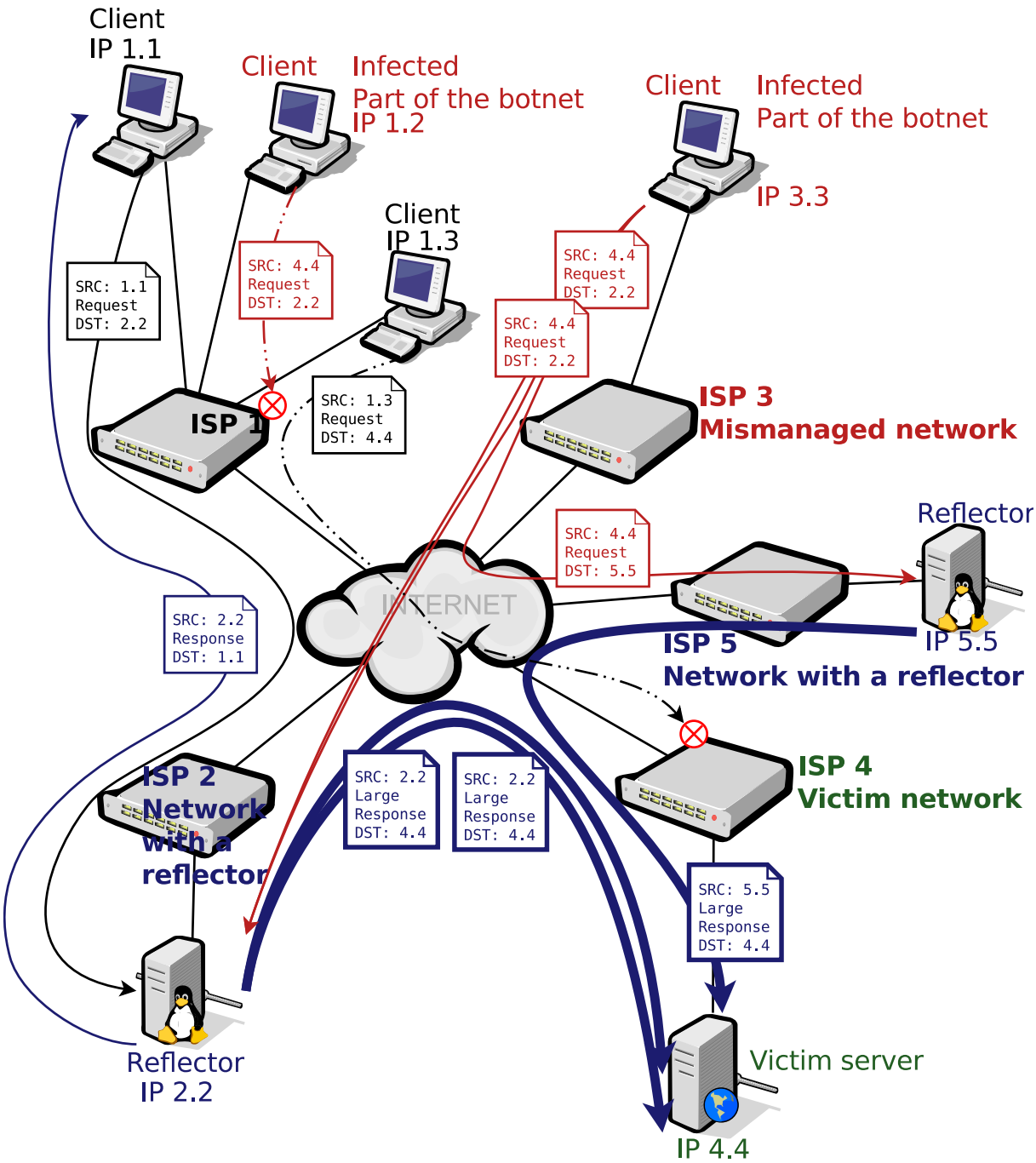

Figure 2: Detailed presentation of volumetric reflected amplified DDoS attack

1.2. Current research and methodology

DDoS attacks are widely discussed and researched in academia. Although raw data is significantly less available to researchers than to commercial and other entities that receive DDoS attacks themselves, in some cases, researchers make special agreements to access it. But in most cases, researchers have to design and set up their experiments relying on publicly available data or data available through academic channels. This section explores various approaches and methodologies utilized in academic research primarily concentrating on the reflected amplified DDoS attacks.

1.2.1. Analyzing individual protocols

Because of the nature of the DDoS, there has to be a network protocol abused especially in the case of reflected attacks. While an attacker can generate and send random payloads that make no sense from the protocol perspective and in some cases reflectors might even respond it would not produce the maximum amplification possible.

Analysis of the protocol definition documentation and source code of different implementations can allow researchers to identify new potentially abusable services. Some assumptions or previous research into the prevalence of analyzed protocol have to be made in advance to choose which of the many to pick for analysis. If the count of found devices having abusable service is low then the overall impact of DDoS attacks is negligible and malicious actors might not even bother with exploiting it. Proper responsible disclosure mandates security researchers to report discovered vulnerabilities in advance to hardware and software vendors and other parties that would be responsible for the issue remediation. In theory, it would preemptively mitigate the abuse of a particular protocol but in reality, the situation is quite different. Research publications and vulnerability reports publicly disclosed after the time period given to vendors elapse nevertheless enable malicious actors to abuse devices that were not remediated.

One of the most prominent aforementioned cases was NTP DDoS. NTP has existed since the 1980s [34] and as the Internet grew became a crucial part of it. Rossow evaluated common UDP-based protocols and identified that most NTP implementations support command to return client list that was a feature of the implementation and not defined in the protocol itself, measured amplification factor was up to 4670 which was the largest from the measured in the research [35]. Because of the potential for abuse, the researcher conducted responsible disclosure to the security community and appropriate vendors but probably related to this disclosure either directly or inferred through released software fixes, malicious actors started exploiting it in the wild. NTP DDoS attacks were first seen on the Internet on December 2013 and within a few months became the main attack vector for large DDoS attacks [36]. The outcome of the remediation campaign was successful and from January to April of 2014 identified NTP server count dropped more than 92% from 1.4 million to 110 thousand. Remediation efforts didn’t seem to be successful until the actual attacks started and real damage had been done.

When analyzing individual protocols non-standard features can be explored in detail. In the case of NTP, returned client list contained also attacked IP addresses thus allowing to discover the targets by being a simple observer and not running any honeypots [36]. It was a unique opportunity to gain full insight into all the attacks and victims for one specific DDoS variant. The only realistic way how to gain this insight for most protocols is to create a large worldwide network of honeypots that can’t be discerned from real reflectors by the attackers.

In the cases of long-term abused services the only research that can be conducted is periodical revisiting of the current situation and observing the changes, possibly also exploring some lesser investigated aspects. Anagnostopoulos et al. explored longtime abused DNS attacks from a new angle, instead of abusing resolvers researchers measured the amplification of the authoritative name servers and discovered that for 47% of servers amplification factor was above 60 [37]. Before that, Anagnostopoulos et al. measured resolver

amplification in multiple European countries and achieved an amplification factor between 37 and 44 [38].

1.2.2. Existing measurement research

At the time of writing this thesis, the only methodology to measure the overall worldwide capacity of DDoS attacks is published in the academic literature by Leverett and Kaplan [4]. Researchers analyzed only reflected volumetric UDP DDoS attacks thus closely relating to this research. More specifically four protocols were analyzed – NTP, DNS, SSDP, and SNMP. Using this methodology it was concluded that the total estimated DDoS attack capacity is 108.49 Tbps. As researchers acknowledged themselves this capacity is limited by other factors not explored in detail and in reality is significantly smaller. This result doesn't take into account the ability of the AS network to handle all of the capacity same time, device load and existing bandwidth utilization, and device computational power that might not be able to handle producing the responses to saturate the whole available network connection.

In addition to the total attack capacity estimate, additional avenues to present and visualize data for easier consumption by non-technical policymakers were explored, e.g., a map of the world with risks posed to others attached to each individual country. The aforementioned visualization allowed researchers to spot an important discovery that developed countries actually possess higher DDoS attack capacity than developing countries. This finding points to the lack of policy to, at the very least, remediate DDoS attacks or its enforcement even in developed countries. Instead of pointing fingers at developing countries, this issue should be addressed internally and on an international level.

The main drawback of the research is that it utilized ready data sets instead of conducting scans and measurements themselves. The risk to data quality without having control over it is high, but the proper scanning technique might require more effort than is available. A more significant issue is the usage of a non-public database of the network connectivity speeds. It limits other researchers to reproduce, actualize and compare the results to proposed new methodologies like the one proposed in this thesis.

Privileged access to large ISP or IXP data enables researchers to gain insights into the real attack capacity flowing through the analyzed network. Kopp et al. [39] reported their observations from a major European IXP privileged vantage point between September 23, 2019 and April 20, 2020. Although it allows researchers to gauge different protocol abuse (relative ranking) it is impossible to extrapolate this data to estimate global attack capacity.

1.2.3. Attack detection

Detecting DDoS attacks is an important part of the mitigation strategy, the quicker an attack is detected the better it can be mitigated while minimizing hindrance to legitimate users. Attack detection is a continuous field of study, as technology and science evolve the detection techniques research evolve as well, concentrating on the applicability of "hot" research topics. Real-world implementations are dealing with a humongous amount of network traffic and thus require detection to be extremely efficient. Usually, the detection is based on predefined thresholds or statistical methods, but newly proposed detection methods can be suitable for particular applications.

Mousavi et al. have investigated SDN controller susceptibility to the attacks and proposed an early detection method [40]. It relied on detecting the lack of entropy for incoming packets which was a lightweight process that could detect attacks within the first 250 incoming packets. Entropy is a widely explored avenue for efficient attack detection, Ma et al. explored the possibility to improve entropy-based detection by applying the Lyapunov exponent [41]. Karimazad et al. explored the applicability of RBF neural networks for attack detection and achieved a detection rate above 96% [42].

1.2.4. Attack analysis

Intelligence regarding individual attacks or parts of the overall attack landscape can provide insight for researchers and knowledge for the decision-making process. This type of research relies heavily on privileged data or network access.

Ordering attacks from DDoS service providers against a monitored network has been explored by Santanna et al. [43]. Researchers located illicit service providers semi-automatically by using the Google search engine and common keywords describing the DDoS service finding 102, they purchased DDoS attacks from 14 attackers but only 9 of those performed reflected DDoS attacks based only on DNS and CharGEN protocols observing a maximum 7 Gbps attack peak. This research method allows researchers to compare real-life attacks technical details of which are often kept private with public data, academic measurements, and various estimates. It allows researchers to determine which of all publicly reachable systems are actually abused for the DDoS attacks, when compared different attacks were predominantly distinct therefore tested services didn’t actually provide anything even close to the maximum possible DDoS attack capacity.

Backscatter of the TCP packets is a common side effect of the SYN flood or other TCP-based DDoS attacks, it can be detected while analyzing packets that reached the darknet. This type of backscatter allows identifying victims because they are the ones creating backscatter by replying to packets with spoofed IPs. If attackers are spoofing IP uniformly throughout the whole Internet address space it is possible to infer ongoing DDoS attacks, their size, and victims. There have been numerous investigations into backscatter caused by DDoS attacks, the most prominent was conducted by Moore et al., determining that TCP was the most observed protocol abused for DDoS visible in the backscatter, researchers were able to identify individual attack attributes like length, size, repetition and victims [44]. Produced results demonstrated that 90% of the attacks lasted less than an hour, the largest identified attack was inferred to be 517,000 packets per second, and they were able to distinguish about 4000 separate attacks per week. But nowadays with a loss of attack effectiveness and prominence compared to reflected DDoS attacks TCP attacks are less researched.

Backscatter analysis has been also explored as the research approach for UDP reflected DDoS attacks. Fachkha et al. analyzed DDoS traffic backscatter and determined that besides scanning activity also attack packets are identifiable [45]. Researchers separated attacks from scanning activity based on the observation that scanning activity for an IP address in a data set consists of a limited number of packets and sometimes requests that do not generate significantly larger responses. From the 3-month data set they were able to distinguish only 134 DNS reflected DDoS attacks. This research approach hasn’t proven to be effective as the

actual DNS attacks rarely are visible in the backscatter because it also wastes the attacker's spoofing resources without contributing to the attack.

1.2.5. Defense solutions

How to effectively defend against DDoS attacks is an open question that researchers attempt to address. Approaches to answering this question involve measuring existing solutions, proposing and validating new solutions, and exploring technical root causes which enable the attacks.

Within recent years one of the most researched topics in detecting and defending against DDoS attacks is the application of Software Defined Networks (SDN) and Cloud technologies. Contradictory relationship between DDoS defense and SDN where resource exhaustion might occur on the SDN plane while trying to defend against the attacks and additional SDN concerns are overviewed by Yan et al. [46].

Offensive security at the moment of writing this thesis was a mostly taboo topic among academia, government, and military organizations. A semi-offensive approach has been explored by Walfish et al. where bots causing application-level resource exhaustion before bandwidth exhaustion were forced to use more bandwidth [47]. It relies on previous research where botnets were comprised of individual bots having very limited bandwidth, by requiring all clients to send more bandwidth while limiting the request count per client, bot bandwidth can be exhausted thus limiting bot effectiveness for current and coinciding DDoS attacks. This approach is not viable for the most problematic case of the reflected DDoS because reflectors are usually non-compromised legitimate services, sending or requesting more data from those can impact their performance.

Filtering particular network traffic at the upstream provider level to limit attacks received by the targeted network and minimize collateral damage is a common DDoS attack mitigation technique called blackholing. Dietzel et al. analyzed blackholing data from a large Internet Exchange and determined that predominantly individual IP addresses are blackholed indicating attacks against specific services, in the rest of the cases distributed systems or networks are likely being targeted [48]. Researchers observed that the average number of blackholed network ranges is exceeding 1000 which would correspond to ongoing attacks being mitigated.

As discussed in [1.1.3. Reflected Amplified Volumetric DDoS attacks], one of the two root causes of the reflected amplified DDoS is the ability to spoof IP addresses that malicious actors abuse. Technical solutions to IP spoofing detection and mitigation have been proposed. Jin et al. initially proposed to analyze the Time-to-Live value in received IP packet headers and compare those to the real path of the IP packet to detect the difference but this approach has issues with multipath routing and other Internet peculiarities [49]. This approach has been revisited and developed further by several researchers, Mirkovic et al. managed to address the main drawbacks caused by the nature of the Internet routing thus creating a self-learning system that erroneously drops a low amount of legitimate traffic and a high amount of spoofed packets [50]. But even when deployed it would not protect against reflected DDoS, as the received attack would consist of non-spoofed packets. None of these solutions have proven to be fully effective against DDoS attacks in real-world deployment, the only proven solution is to universally apply BCP 38 [32].

1.2.6. IP Spoofing

The ability to spoof the IP address of the packets is the main cause for multiple types of attacks, including the most problematic reflected DDoS, that is why it is important to analyze sources of IP spoofed packets to better understand and address the issue.

Center for Applied Internet Data Analysis (CAIDA) based at the University of California's San Diego Supercomputer Center has been conducting research into the state of IP spoofing and continuous monitoring since 2008 [51]. When a spoofed packet is received it is already too late to search for its source, defending against received spoofed packets is not effective, meaning this issue has to be researched and mitigated at the source. CAIDA has developed a tool for popular OSes which users have to download and execute themselves, this tool sends IP packets from user computers trying to spoof the IP address in multiple ways. Data quality depends on measurement count and even more on distribution across as many ASes as possible. If there is not a single measurement from an AS, then no exact conclusions can be made, blocking spoofing could possibly be inferred from upstream AS results, but no research into this topic has been published. A similar issue is with the old data, if the latest test was conducted years ago it is unclear if the judgment about a particular network is still relevant, because there might have been remediation or less likely new misconfiguration.

CAIDA software creates various types of spoofed IP packets to determine whether they can exit the user's network, including both IPv4 and IPv6 protocols, both public and private IP addresses and parent ranges [51]. These packets are sent to the CAIDA server therefore researchers can determine that received testing packets have been routed through the Internet without proper filtering. Adjacent IP spoofing is a separate concern, if filtering is happening on a network service provider level then depending on the network's architecture, it might be problematic to filter out one client spoofing another client's IP ranges. Then it can be abused to cause DDoS attacks against neighboring networks.

CAIDA spoofer project publishes updated and historical data from their measurements, in total 22.6% of the IPv4 AS not using NAT were spoofable in July 2018 which corresponded to 14.3% of the IP address blocks [51]. In general countries in developing regions are found to be proportionally more spoofable than already developed countries. But in absolute numbers, the USA has most of the spoofable IP blocks. This finding that in absolute numbers developed countries contribute most to cybersecurity issues is similar to Leverett and Kaplan [4]. These results demonstrate that the spoofing issue is still serious and not properly addressed, even a much smaller number of spoofable networks can cause as large DDoS attacks as current ones because bots or servers using spoofable networks are not usually the bottleneck of the attacks.

Relying on end users to collect data might not be the best solution because of the discussed drawbacks. Lone et al. have proposed a new measurement technique relying on a traceroute sent from a vantage point outside of the tested network [52]. If a traceroute packet with the vantage point's IP address as the source address is sent on purpose to AS which doesn't serve the IP range of the target and is returned researchers infer that source IP address filtering is not sufficient. Their results indicated that 1.3% of network service provider ASes and 3.2% of network service customer ASes allow IP spoofing.

1.2.7. Internet connectivity outage detection

A major DDoS attack can cause local Internet outages. Inadvertently research that focuses on the Internet speed, stability, and reachability can detect also the issues caused by DDoS attacks. This type of research doesn't necessarily focus on low-level networking data, many public sources of information often are more easily acquirable. Aceto et al. while conducting a comprehensive survey on Internet outages determined that DDoS attacks are a substantial cause of the outages [2].

Banerjee et al. explored Internet outages by analyzing large data sets of mailing lists of network operators discussing causes for network outages and other intermittent issues [53]. They discovered that DDoS was the main cause in the security-related Internet outage category. Gunawi et al. investigated public information regarding major cloud provider outages [54]. Researchers determined that DDoS attacks accounted for about 5% of the cloud providers' major outages.

1.2.8. Motivation behind the attacks

The motivation behind the DDoS attacks varies significantly, actual attack executors are commonly driven by financial gains and sometimes as a form of political activism. Insights into these motivations can permit researchers and decision-makers to address the DDoS issue more appropriately. These research approaches involve analyzing public and hidden forums and websites offering and discussing the service, leaked and publicly accessible data, and attack details.

DDoS attacks have become a popular service provided by the cybercrime industry thus allowing people without any knowledge and qualification to launch attacks against any targets for as low as 1 USD and even offering a "free testing period", these services are commonly referred to as *booters* [43]. This has changed the threat landscape dramatically and all services on the Internet are threatened, even students that don't want to participate in online exams can order DDoS attacks to successfully bring the exam infrastructure down [43].

There is an interesting phenomenon that DDoS attacks are not only used against public services but also against online gamers who use residential Internet connections and thus are susceptible to network overload, leaked data from *TwBooter* DDoS service demonstrated that most of its victims are online gamers and attacks last usually less than 10 minutes, which is enough time to make the victim lose a particular online match, motivation can be as simple as winning a game without any monetary or other kind of gain [55]. Even before DDoS as a service was widely available not only individual gamers but also online game servers and related forums were commonly targeted. A special case is online gambling. System owners can escape local laws and regulations by hosting gambling systems in offshore locations but by being a lucrative business it attracts cybercriminals, extortion is the main motivation behind attacking these systems. System owners have a dilemma either lose money from being inaccessible to clients or lose money by paying extortionists who will attack again in the future [56]. All these extortion attacks can be classified as ransom DDoS (RDDoS). These and similar high-risk systems are the reason for creating new DDoS mitigation and protection industry.

Booter services can be found by clients via search engines and hacking forum posts, leaked data showed that a single DDoS service can make up to 24 thousand USD per month [57]. Before the rise of cryptocurrencies to prominence it was a promising approach to target conventional online payment systems like *PayPal* accounts of the *booter* services which could be determined by making a payment for the criminal service. It targeted not only received funds but forcing cryptocurrencies as the only method of payment would also decrease the potential earnings from not-so-technically savvy customers thus minimizing profits and motivation to provide the services.

Political activism and dissent can be placed into a different category of motivation. Not only the same approach of purchasing service is available but users can install software capable of creating DoS attacks directly from their computer, the most popular being LOIC which permits to voluntarily join a botnet [58], therefore the DDoS attack strength against a single target is proportional to the count of participants that are representing same opinion [59].

1.3. Research outside of academia

Case studies analyzing individual attacks are occasionally published online by commercial entities receiving or mitigating DDoS attacks. It happens when a new protocol is starting to be abused for the attacks or when previous attack records are broken. The motivation behind these case studies is to advertise the ability to handle DDoS attacks to gain more clients. Details in these case studies are usually very restricted so as to not reveal any commercial information or weak points in the defenses. These case studies have become the main point of reference when discussing DDoS attack capacity. When the question is raised in most settings what is the maximum realistic DDoS attack capacity, the following answer usually refers to the latest or a recently published attack case study. The DDoS mitigation industry has addressed non-volumetric reflected DDoS attacks to such an extent that they generally don’t make any news or reports anymore.

Some of the most known and referenced case studies of the aforementioned type are presented in table 1. The overall trend of the reported attacks is going up and protocols that can be exploited are exploited.

Organization	Service abused (port)	Capacity	Date reported
Arbor – DDoS mitigator	Memcached (UDP 11211)	1.7 Tbps[60]	05 Mar 2018
Akamai – CDN provider	Memcached (UDP 11211)	1.3 Tbps[61]	01 Mar 2018
Cloudflare – CDN provider	Memcached (UDP 11211)	260 Gbps[62]	27 Feb 2018
Cloudflare – CDN provider	SSDP (UDP 1900)	100 Gbps[63]	28 Jun 2017
Akamai – CDN provider	CLDAP (UDP 389)	24 Gbps[64]	04 Mar 2017
Cloudflare – CDN provider	NTP (UDP 123)	400 Gbps[65]	13 Feb 2014
Cloudflare – CDN provider	DNS (UDP 53)	75 Gbps[66]	20 Mar 2013
Cloudflare – CDN provider	SNMP (UDP 161)	21 Gbps[67]	03 Aug 2012

Table 1: Highly referenced case studies and reports published by the industry

There are four major issues with this approach and that is one of the reasons for this research:

1. Attacks that have been successful usually are not described so as to not lose reputation and reveal the weakness in defenses. There is a strong commercial motive to keep this kind of information hidden.
2. Individual DDoS attacks mostly consist of one type of vulnerable protocol being abused. Consumer of the report doesn't get the holistic picture of the total possible attack capacity.
3. It is unknown what percentage of the total attack capacity of a particular protocol was actually utilized. It might be observable by looking at the unique IP addresses in logs to determine what percentage of the total reflectors of the specific type participated in the attack. But that doesn't necessarily reflect the percentage of the total possible attack capacity.
4. These capacity values represent a single point in time. Even if these really were the maximums that included all the significantly contributing reflectors at their maximum capacity which is highly unlikely, they still have no relevancy for understanding current attack capacity.

1.3.1. Abusable network service scanning projects

Whenever a new service gets abused for DDoS attacks usually a new scanning project presenting the results publicly is created. The creators of these projects are organizations and individuals working in networking or cybersecurity fields who are affected by DDoS attacks but frequently prefer to remain anonymous. The main purpose of such projects is to advise the public in general and network owners that their networks contain systems that can be abused. It can be done by either emailing notification messages to network abuse addresses, notifying only persons that signed up for their network ranges, or enabling them to do network range searches in their database. The goal of these projects is to minimize the number of abusable devices as much and as quickly as possible.

Sometimes these projects cooperate with researchers from academia by providing them with raw data, so research can concentrate on data analysis instead of technical data gathering. On its own, the research usually is limited to scanning the Internet for all the devices using specific ports and protocols, grouping found ones by AS and geographic (country) attribute, and presenting the results in table and graph formats. If scans are repeated then a comparison can be made between time spans and device count decline trends can be identified. If scans are scheduled periodically then the current situation can be ascertained.

Many open ports exposed to the Internet are being scanned by The Shadowserver Foundation including more than 10 that are most commonly used for amplified reflected DDoS attacks [68]. The approach is to use the same scanning infrastructure, the same data processing and visualization software, and the same notification approach for every new protocol added. The only change that is required is developing and deploying the new protocol payload and port.

Open NTP project provides scan data about network time servers supporting two types of commands that can be exploited for DDoS attacks [69]. The issue with scanning projects that focus only on one type of service is that with decreased relevance or time they tend to

stale and stop conducting the scans. In this case, the latest data is from October 2016 at which point they detected 3.8 million of the reflectors.

On the opposite side, scanning activities can be detected and presented in real-time and as historical data. One of these projects is NetworkScan Mon, it aggregates data by source and destination attributes of the IP packets and presents aggregated statistics which revealed that in July 2018 there was not a single protocol abusable for DDoS attacks among the top 10 ports receiving scanning activities [70]. It indicates that DDoS is a specialized niche of cybercrime and because of the required 2-pronged execution it is less attractive to cybercriminals as opposed to most popular scanned ports which are used by services that can be directly exploited. This project has partially identified and logged IP addresses used in the current research, other academic researchers can be identified as well by the university AS names.

Open resolver project is one of the most used and referenced reflector scanning projects in the academic literature. It has monitored the DNS resolver count from 2013 till the beginning of 2017, providing statistics about resolver count and replies [71], no information about the current or future functioning of the project is known. Raw scanning data sets are being provided to verified academic researchers. CyberGreen is the best-known organization developing a DDoS monitoring project which provides some statistics and data sets to the public but the full detailed information is kept private [72]. This project relies more on calculations and estimates rather than actual measurements besides the scanning stage.

1.3.2. Attack monitoring projects

It is possible to monitor DDoS attacks and extract some of the attack attributes by either passively monitoring network traffic at IXPs or maintaining a distributed set of honeypots that pretend to be exploitable network services.

DDoS Mon project provides insight into worldwide DDoS attack statistics and historical trends, in July 2018 it reported averaging about 20,000 attacked IP addresses per day [73]. An attacked IP address doesn’t necessarily equal a single attack or target as systems under attack can have multiple IP addresses. But no deeper analysis into grouping separate IP addresses into a single target was provided, it might have a potential for separate research. In the same time period, USA and China were the most attacked countries, HTTP port 80 and HTTPS port 443 were the most targeted ports, and websites using *.com* top-level domains were targeted most often. Amplification and reflection-based attacks were the most common amounting to nearly 70% of the DDoS attacks by frequency, the most commonly abused protocols were CLDAP, NTP, and DNS. These attack statistics have drawbacks in the sense that a number of some specific abused protocol services don’t correspond with their overall bandwidth contribution to the attack which is the main property of the DDoS attack. Additionally, multiple types of DDoS botnet command-and-control servers are monitored and commands are analyzed to extract information about targets.

1.4. DDoS over IPv6

IPv6 deployment has opened new avenues for network misconfigurations exploitable for DDoS attacks by either weakened defenses or by unprotected network services utilizable for reflection.

Scanning the IPv6 part of the Internet the same way as IPv4 by fully enumerating the address space is unfeasible because of the difference in the address space size 128 bits vs. 32 bits which is 28 orders of magnitude larger. Even taking into account the fact that IPv6 address space is sparsely populated also on the network prefix level, and it is possible to extract currently employed subnet prefixes from routing tables and scan only those, the host part of the network address is commonly 64 bits large [74] which is still as unfeasible as scanning the whole address range. All of the known approaches for IPv6 network scanning are analyzed in RFC 7707 [75]. Most of the described techniques rely on scanning a particular subnet from outside or inside of the network for penetration testing purposes, for this scenario, even a partial view of the IPv6 network can be sufficient. Many of the described approaches like scanning for vendors that are known suppliers for target networks, using local broadcasts, and DHCP assignment patterns are completely irrelevant to Internet-wide scans. The most relevant method for this thesis is the enumeration of low bytes of the host part of the network address which are commonly assigned manually by system administrators for publicly reachable network services. It can get as high as 92.65% in the case of tested mail servers but at the same time client addresses rarely use this addressing scheme instead opting for randomized addresses which were introduced because of privacy concerns about the unique traceability of MAC addresses being used as part of IPv6 address [76]. Thus the described IPv6 scanning techniques can’t be applied for the whole Internet scan with the expectation of gaining a full view of the publicly reachable devices.

Gasser et al. have approached this issue from a different direction and instead of active probing conducted passive monitoring of traffic and other public sources of information and found in total 150 million unique IPv6 addresses [77]. IP addresses were acquired from an IXP and a scientific network, and because of the dominance of randomized IPv6 addresses this approach is the only one that together with methodology from RFC 7707 can produce a view of the IPv6 Internet as close as possible to the real world. The limitation of this approach is that for the full IPv6 Internet view, it would require data from many IXPs and even then some devices that have limited communication might not be identified. The factor that limits this approach is that there have to be agreements with a variety of IXPs about extracting data without violating individual and company privacy. Furthermore, there are concerns as to what extent the extracted IPv6 addresses can be used and would active probing for services on them be even permitted.

Hendriks et al. searched for open DNS resolvers that can be abused for reflected amplified DDoS over IPv6 [78] by extracting data from the DNS servers themselves. A DNS server scan was conducted over IPv4 and then the identified open resolvers were sent specially prepared queries that would be resolved over IPv6 if it was configured thus unmasking the IPv6 address. The main concern of the research was that the DNS servers running over IPv6 might be significantly less protected than IPv4 because of human factors

applying firewall rules and configuring settings. This approach is different from previous ones but it is applicable only to DNS servers.

The total available upload bandwidth of a device connected to the Internet is not limited by the Internet protocol version. But rather by technical or contractual limitations of an ISP which provides network connectivity. If the device has vulnerable network services running on both IPv4 and IPv6 protocols total contribution to the attack capacity should not exceed of that what only the IPv4 service running can contribute. There can be exceptions – only IPv6 connectivity on the device, different network transit providers for different protocol versions, configuration differences, etc. One of the possibilities is that AS could have separate physical network connections for IPv4 and IPv6, then the total attack capacity contribution of this AS would be higher for both protocols than only IPv4. Another concern raised in [78] is that configuration of running services or protecting firewalls might be neglected on IPv6 thus resulting in attack potential only over IPv6.

Because of these stated technical limitations and potentially small additional gain from including IPv6 in this research it was decided to exclude IPv6 from the scope of this thesis. Same time author acknowledges the importance of IPv6 and proposes it as future work or separate research.

2. ATTACK CAPACITY MEASUREMENT METHODOLOGY

This chapter presents an overview of the proposed DDoS attack measurement methodology without examining the technical details of the implementation, technical implementation is presented in [4. Protocol measurements]. Additionally, major issues regarding scope, legal and ethical considerations, and non-implementation technical options are discussed.

2.1. Scope of the research

The research reported and methodology proposed in this thesis focus only on measuring individual services (protocols) utilizing stateless network protocols more specifically UDP which generate amplified responses. Because of the IPv6 limitations discussed in [1.4. DDoS over IPv6] only the IPv4 protocol is being considered. The proposed methodology is not applicable to the DDoS research outside of the defined scope as scanning and measurement stages will not produce suitable data to use in the capacity calculations. Although [5. Global attack capacity] discusses totaling of different protocol measurements it is outside of the defined scope of the proposed methodology.

2.2. Proposed measurement methodology

In theory, an experiment directly measuring the maximal possible DDoS attack which is launched against the network being measured could produce a number that would represent the total DDoS attack capacity worldwide. In practice that is impossible because of the prohibitive costs of creating a network supporting the attack size and collateral damage to other networks. Even if a CDN network with distributed presence around the world would be used, the collateral damage potential would still persist.

The proposed solution is to split the experiment into smaller chunks and conduct “self” DDoS. The smallest block of the experiment would be a single device connected to the Internet providing a service abusable for DDoS attacks. Testing an individual device shouldn’t cause any networking issues in any part of the network path between the device and the measurement system. Limiting test traffic should protect the tested device from overloading or any damage. When all the reflectors of a particular abused protocol are measured, limitations that affect total attack capacity can be applied.

The steps of the proposed methodology are the following:

1. Identify abusable network services (protocols) and corresponding requests, that are exploitable for the DDoS attacks.
2. For each of those services identify all the publicly reachable devices by conducting the Internet-wide scan.
3. For every one of the identified devices send a set of exploitable requests to measure individual properties – amplification, rate limiting, speed, etc.
4. Identify limiting factors and apply those to the measured set to produce the total attack capacity of the measured protocol.

Although the idea to measure the total DDoS attack capacity is similar to research conducted by Leverett and Kaplan [4], the methodology differs significantly. There are no interactions with individual devices trying to measure respective contributions to the issue. When determining the bottleneck of total DDoS attack capacity only the median upstream connection speed of a country is used. Although not a methodological difference but no measurements were conducted as data sets were provided by third parties.

2.3. Services abusable for DDoS attacks

Any service connected to the Internet and responding to requests can potentially be abused to some extent. This thesis focuses only on services using UDP protocol as these are the easiest to abuse and produce the largest BAF. For the reflected amplified DDoS attack variant, attackers are concerned with the ability to locate abusable services easily and the amplification those services can provide. From an attacker's perspective, the perfect service to abuse is the one comprised of millions of devices having BAF in hundreds or thousands. NTP has been a great example of that but because the issue was so serious that it was affecting also reflectors it was significantly remediated in a short period of time [36].

The worst services to abuse from an attacker's perspective are the ones having very few publicly reachable devices providing insignificant amplification. But even if the number of devices is high and BAF is low or vice versa, it might still cripple the attacker's ability to launch a large-scale DDoS attack. Remediation efforts of the industry target both of these characteristics. In reality, most of the attacks are closer to the middle ground containing enough reflectors which produce average amplification.

Attackers' spoofable bandwidth is the most valuable resource for the attacks, its capacity is limited by the number of compromised systems under the control of the attackers. The higher is BAF of the abused protocol the larger DDoS attacks can be produced with the same limited resources. The larger the DDoS attack the higher the probability for it to negatively affect the victim. Attackers with very limited spoofable bandwidth might prefer protocols with the highest possible BAF, especially when a new protocol starts getting abused right before any remediation has taken effect.

The ability to easily scan for abusable services is another requirement from an attacker. It is easy to scan for services that are located on known fixed port numbers, always reply to generic requests (e.g., DNS), or respond to amplified requests when are vulnerable (e.g., NTP). Said properties are common for protocols providing some service to clients but might not be the case for client protocols. Client protocols can choose random ports as they are initiating communications and respond only in some cases. It might be possible to identify clients with appropriate ports for particular protocols using directory servers or through communicating with other clients but in all these cases it requires additional effort and resources from the attacker. For this reason, client protocols are not significantly abused for DDoS attacks, even though research demonstrates that amplification is possible and amplifiers are common [35]. When all server protocols are significantly remediated malicious parties might turn to abusing clients but currently that is less efficient.

For the purposes of this research, only the protocols already commonly abused for the DDoS attacks presented in different sources are measured. These sources include DDoS

attack monitoring projects [73], academic publications [35], [39], and non-academic scanning projects [68]. Identifying vulnerable services is outside of the scope of this research, analyzing individual protocols for the abuse potential is discussed in [1.2.1. Analyzing individual protocols].

2.4. Scanning the Internet

Internet-wide scanning is a common occurrence and is conducted by various parties – researchers, malicious actors and commercial entities but the motivation of these parties are diametrically opposite. Large-scale analysis of scanning being conducted on the Internet has been provided by Durumeric et al. by investigating network traffic received by a large darknet (network region containing no running services) [79]. By grouping together source IP addresses from the same subnets and aggregating by time and destination ports researchers were able to distinguish the individual scans. Which allowed fingerprinting the used tools, sources and speed of the scans, and targeted services. Most of the defensive research done in academia or by commercial entities was easily identifiable by network ranges, reverse DNS entries, and web pages that explained the purpose and authors of the research. Seemingly malicious actors in most cases used hosting providers known for poor security practices and scanned for recently published vulnerabilities. Active probing demonstrated that only 0.05% of the tested network ranges block subnets from which scanning activity originates which has a negligible effect on the research data quality.

The first commercial search engine that indexes Internet-exposed devices was *Shodan* [80]. It crawls the whole IPv4 address space testing for common open ports and tries to extract identifying data using appropriate protocol payloads. Competitor targeting specifically the academic community *Censys* was released later on [81]. Both have been used extensively for commercial and academic research purposes as those enable the extraction of useful data without creating a scanning setup which can have technical and bureaucratic pitfalls. Additional processing of the results after service discovery offers a data set that is much more detailed thnt a common scanner would create as an output.

There are various primarily commercial entities conducting Internet-wide scans for their own benefit and not publishing the results. Researchers can negotiate access to that data for a specific purpose. This approach can limit the researcher’s ability to be completely independent and affect the research outcome when access gets revoked.

Although mentioned sources of data have proven to be useful for research purposes in the case of the current study more control is needed especially over the age of data. When using third-party scanning results one has limited options to pick data sets by scanning time period and even if the newest data set can be selected the largest issue remains as there is no control over when the individual device identified in the scanning process was found. Individual entry can be hours or even days old. When measuring individual devices it becomes hard to separate the devices that refuse to communicate properly from the ones that have become unreachable over time – disconnected from the network or power, changed IP address or some network connectivity issues (churn) had occurred. To maximize data quality all of the individual processing and communicating must occur instantaneously after every

single individual device is discovered. It forces the author to conduct Internet-wide scans as part of this research.

2.4.1. Role of the Internet scanning

Role of the Internet scanning is simple but crucial for this research. TCP open port scanning is the easiest to conduct as there is no payload and all of the TCP connection establishments are uniform and independent of the port or the protocol used. As the scope of this research is limited exclusively to the abusable network services running UDP protocol, scanning becomes more complicated. UDP as a stateless protocol has no connection establishment thus the only way to identify service is to send an actual payload and wait for the appropriate response. Every network service running on UDP requires a different payload. Network scanners frequently provide built-in generic payloads for common services that are optimized to generate (discover devices) as many responses as possible. But as noted by Hendriks et al. default provided payloads might not be sufficient because of fake devices and network misconfigurations [78].

The goal of the scanning stage is to simply identify all the devices on the Internet running a specific network service so this information can be used for further processing in the next stage. Only then do any further communications with devices occur deciding if it is abusable for the DDoS attacks or not.

2.4.2. Selected solution

Currently, the most utilized Internet scanning tool (judging by the number of indexed publications in Scopus) among academic researchers is `zmap` originally developed by researchers from the University of Michigan. It has been designed and developed with the goal of scanning efficiently the whole Internet. Some of the design choices make it especially effective compared to the alternatives [82]:

- Randomizing destination IP address so the generated network load is equally spread against the target network (the whole Internet).
- Skipping OS TCP/IP network stack and generating raw Ethernet frames thus avoiding OS network bottlenecks.
- Not maintaining a connection state but trying to extract as much information as possible from the received responses.
- No retransmission of lost packets while still maintaining up to 98% network coverage using a single packet per the destination IP address.

These design choices enable `zmap` to complete the full Internet-wide scan efficiently in sub one hour time using only a single average computer [82]. In addition, features like the ability to launch scans from multiple systems simultaneously, blacklisting of networks, and using multiple source IP addresses on a single system make it production ready. Further improvements to the `zmap` have enabled it to utilize a full 10 Gbps network connection which allowed it to scan the whole Internet in less than 5 minutes using server-grade hardware [83]. Although not relevant to this thesis for network security researchers it has created a possibility to take near-perfect snapshots of the state of the Internet and discover systems susceptible to newly published vulnerabilities almost instantly which can be exploited by malicious parties as well.

The modular architecture of `zmap` enables it to be extended with almost any network scanning functionality without modifying the source code of the scanner core. And the modifiable output format can be directly piped into any third-party or custom tool for further data processing or network interactions. The tool over time has evolved into the ZMap Project containing a whole toolset of scanning and scan data processing software [84].

Because of the author's previous experience, discussed factors, dominance among academic researchers, and functionality that satisfies conducted research requirements `zmap` was chosen as the tool for the Internet scanning stage.

2.4.3. Alternative solutions

Currently, there are multiple competing tools suitable for Internet-wide scans, performance and functionality wise the one most closely resembling `zmap` is `masscan` [85]. Although used in the industry quite extensively it has not gained traction among academic researchers (judging by the number of indexed publications in Scopus) possibly because it was released outside of the academic setting. Interestingly both tools have been released around the same time indicating high demand for efficient Internet-wide scanning, before that another tool `unicornscan` provided similar functionality but has lost the functionality battle [86].

One of the most popular network scanning tools is `nmap` [87] which has a wide range of functionality and serves well for local network testing or remote limited network range scanning for the information gathering purposes of penetration testing. Although it has been used for the Internet-wide scanning the efficiency and speed are too low to guarantee data quality. It has been estimated that `zmap` with equivalent accuracy is 1300 times faster than `nmap` for the Internet-wide scan [82].

Any programming language supporting a network stack can be used to develop custom computer programs to scan the whole Internet for some specific purpose. This approach has been used by researchers (including the author for the previous research) extensively before the development of `zmap` but it has several serious drawbacks. The program becomes either simple but slow or fast but complex because of the required parallelism. OS network stack tuning becomes necessary to maintain produced data quality. Limited reusability of produced software between separate researches makes this approach wasteful and replicability in the academic setting becomes limited.

2.5. Measuring amplification

Amplification is an important property of the set of found devices to understand the abuse potential of the protocol, it is needed to make measurement decisions but it might not be required to calculate the total attack capacity of a protocol. If the device count is low then the total attack capacity is low even if amplification is significant. Attackers don't care about services that provide no or small amplification even if the set of devices is large, because small amplification is still affected by packet loss, rate limiting, and other factors. In this scenario, an attacker can more easily execute a direct attack from bots against a victim using spoofed IP addresses. Any device or network service that responds to the request with a larger response can be referred to as an amplifier. Amplification in the case of DDoS attacks is a

property of the attack that describes the attacker's ability to utilize the bandwidth available to them but cause more bandwidth to be received by the victim.

The commonly used term to describe amplification size is bandwidth amplification factor (BAF), it was coined by Rossow and defined as "bandwidth multiplier in terms of number of UDP payload bytes that an amplifier sends to answer a request, compared to the number of UDP payload bytes of the request" [35]. The researcher knowingly disregards Ethernet, IP, UDP headers to preemptively address changes in the IP protocol version and keep BAF future-proof. In some scenarios, this calculation can produce a wrong impression because network connection speed is defined as the ability to move the amount of bits on the lowest physical level of the network link. Even if the UDP payload was 0 it still would consume some bandwidth of the network link to send or receive such a packet, UDP header would be 8 bytes [88], IP protocol header would be at least 20 bytes [89], and in the case of Ethernet frame (header, CRC, preamble) at least another 26 bytes are used [89]. Totaling at least 54 bytes of the network bandwidth consumed for transferring a 0-length UDP packet. If a payload causing the amplified response is very small (a few bytes) and the calculated BAF is a multiple (e.g., 3-10), then the real link layer amplification the victim receives can be multiple times smaller than the BAF. In real life attacks request payload and BAF is usually large enough to disregard the overhead for estimation purposes.

In the same research [35] term packet amplification factor (PAF) was proposed as the ratio of received packet count to sent packet count. This term hasn't been commonly adapted because modern network devices computationally can handle small network packets up to the bandwidth capacity of the network port. Thus the network port usually has to be exhausted volumetrically by bandwidth to cause DDoS before computational resources of packet processing are exhausted. In most cases, PAF is far less relevant than BAF when analyzing DDoS attacks. Usually, PAF is not even mentioned when discussing case studies of attacks or in academic research.

Amplification can be caused by the two types of commands – standard (protocol) and non-standard. Non-standard commands are not defined in the protocol specification and can be debugging or implementation features. These features can usually be safely disabled without affecting the service functionality. But when these non-standard features propagate on the Internet it is caused by the default software configuration distribution over a long period of time, at some point attackers can discover those and abuse them, after that, it takes months and even longer to remediate those services. In the case of DNS, amplification is caused by standard request command which generates a large response, large responses are defined in the protocol and are always useful for DNS functioning, e.g., many alternative IP addresses for a single domain request. This feature of the protocol can't be disabled, this is one of the reasons why the DNS is a longtime abused service. In the case of NTP, the opposite is true, abused command is debug feature that is not defined in the protocol specification and does not affect NTP functionality, it spread across many platforms because it was enabled by default in the software implementation. It got significantly mitigated by releasing software updates with this feature disabled, only leaving behind devices that are not updated and managed.

Measuring amplification is one of the steps of the research methodology proposed in this thesis. It can be done by sending a single request that is known to cause amplification and waiting for the response and measuring the total bandwidth and packet count received. Real-

world amplification per every potentially abusable device has not yet been measured and published in other research. Amplification testing payload has to be developed and tested for every investigated protocol. There are two variants of commands that are abused for amplification – parameterless or parameterized. Measuring amplification for requests without any specific parameter is easier, it provides the maximum result for every device regardless of other factors. Parameterized requests, e.g., a specific type of query for a specific domain name in the case of DNS, affect response size based on the input parameter. This variant has to be carefully selected from real-life attack reports and tested that it is still valid, or manually crafted, wrong parameter selection will not produce maximum amplification which is a risk for data quality for this research.

Measuring amplification in theory should be a safe action as it is using standard features of the devices, it should neither cause any issues to the device nor the network. In reality, there are anomalies detected by the author and other researchers, there are amplifiers that in response to a single packet send multiple gigabytes of the response while sustaining hundreds of Mbps [36]. This not only endangers data quality but also the tested device itself, the network it is located in, and other networks on its path to the measurement network.

When discussing the amplification of a set of devices usually some average number either measured or estimated per protocol is used as the amplification factor, e.g., provided by US-CERT [90] or Rossow [35]. Generally, BAF measurements are simple scans utilizing optimized amplification-causing payloads and are calculated for each received response (single response per device), if this payload discovers a sufficient number of devices then BAF property can be extracted from the standard quantitative scan. Although the implementation of the proposed measurement methodology relies on this simplified BAF in [4. Protocol measurements], the methodology proposes also optional whole protocol BAF which can be expressed as the ratio of all received (0 to measurement count per device) response payload bytes to all sent measurement payload bytes. This metric is meaningless without a universal limitation applied across all the measured protocols, e.g., device speed threshold, rate limit threshold. If an attacker relies on similar limitations then the produced protocol BAF can correspond to the real-world DDoS attack amplification and can be used to determine the required spoofable bandwidth.

The purpose of this methodological step is to establish to what extent the protocol is abusable compared to the reported theoretical maximum or initially measured BAF values. Decreasing BAF is one of the remediation approaches which is usually not remeasured after the introduction and proliferation have occurred, e.g., configuration bundled with a software package that is automatically updated.

2.6. Detecting rate limiting

Attackers abusing network services rely on that they don’t have any rate limiting. Academic and industry research usually stops at identifying the devices or estimating amplification, there is no published research regarding real-world rate limiting among the identified potentially abusable devices. The lack of aforementioned research is the reason why rate limit detection is part of the proposed methodology for this thesis.

For measuring amplification one request packet can suffice but for detecting rate limit many identical requests have to be sent and answers received and measured. Every measured protocol (implementation) can have its own specific rate limit that needs to be tested, e.g., default configuration value for software distribution. If it is not possible to identify trustworthy rate limit values then more aggressive testing might be required but only for a limited set of devices. Because of the count of sent requests, rate limit measurement is much riskier than amplification measurement, no harm should be caused to properly designed and engineered devices. In some rare and specific cases, a burst of packets might overload improperly designed devices, causing a slowdown, hanging, or reboot, but no physical damage should occur. The assumption for this thesis is that these devices are publicly reachable and already receive traffic that would cause potential issues often, and in these cases, no real damage is caused.

Rate limiting can be caused by software implementing RRL to mitigate the consequences of abusing stateless protocol for DDoS attacks [91]. RRL can be implemented in different ways over different time intervals, by using the burst response measuring method actual rate limit per second is not known. For approximation in this thesis, it is assumed that the measured rate limit is per 1 second but the primary concern of this methodological step is to detect if any rate limit is present. If a need to measure RRL more precisely arises in the future it can be accomplished by using a continuous stream of requests for multiple seconds instead of the single burst.

Rate limiting might also be an unintended consequence of the limited computing or network resources. A significant portion of the reflectors is CPE, networking, and IoT devices with low computing power which might also have a low-speed network connection. It is possible that the resources of a device at the moment of measurement are utilized for some legitimate task or even for ongoing real DDoS attacks to the extent which will produce false positive rate limit detection, it is an acknowledged risk for data quality but it is not addressed within the scope of this thesis. Causes for rate limiting might be a worthy separate research question but because of the complexity is not investigated in this thesis. For the purposes of this thesis, there is no differentiation between different causes of rate limiting.

Technically rate limit measurement can be implemented in two ways – by sending a burst of packets and verifying the count of received packets or by analyzing every pair of response and request packet sets. Because the measurement requests are identical for most of the protocols it should produce identical or very similar response packets count-wise. By using packet count from the amplification measurement step, it is possible to divide the number of received packets with PAF to roughly estimate if the resulting value is close enough to the number of sent requests, if it is then there is no rate limiting or it is above the selected threshold, otherwise resulting value approximately corresponds to rate limit.

More precisely rate limiting can be measured by mapping sets of response packets to each appropriate request. To an extent it allows to differentiate packet loss from rate limiting, as rate limiting is implemented per response basis, it potentially allows to identify exactly from which request responses stopped coming. This method is also suitable for measuring rate limiting that is not per second basis by detecting at which request number responses stop and at which restart. This type of measurement technically can be implemented in two ways. The easiest way is that every request uses a different source port number and every response

packet set will be received by a different port. But in DDoS attacks usually all of the reflected packets target a single port. A more challenging way is to use the same port but try to differentiate between responses which depending on the tested protocol might be unfeasible because all the sent requests have to be the same. Different protocols possibly might produce better data using different measurement methods, from a methodological perspective it does not matter which approach is implemented as long as for every tested protocol implementation advantages and disadvantages are considered.

A more advanced way how to detect rate limiting could be sending the same amount of burst packets from single IP and a set of IP addresses, and comparing the results. If the set of IP addresses receives significantly more traffic than the single one then it is highly likely that rate limiting exists. The main requirement for the set is that IP addresses are from different subnets as some rate limiting is implemented on per subnet not per IP address basis. This approach is not explored further within this thesis but has the potential for future research.

It is highly advantageous for attackers to abuse only the network services that don't have rate limiting, otherwise, resources of the devices with the ability to spoof IP addresses are wasted as reflectors receive requests but don't send responses to the victims. Published research doesn't analyze the activities of the attackers regarding rate limit measurements. The author has attempted to identify if attackers are measuring RRL on DNS servers by creating honeypots and logging all the requests but the produced data were inconclusive and thus not analyzed in the context of this thesis. This measurement has to come directly from the machine under the attacker's control without IP spoofing and the testing pattern should be observable – a burst of requests from a single IP address in a short period of time (testing stage) and then after possibly significant delay following a large amount of continuous requests from different spoofed IP addresses (attack stage). This open question regarding attacker activities before the attack execution warrants further research.

2.7. Identifying bottlenecks and applying limitations

After detecting rate limiting on every tested device it might be tempting to continue the same stream of requests and measure the total upstream bandwidth of the device, then sum together all the measured results to produce total DDoS attack capacity. This approach has the potential to negatively impact some of the tested devices and services they provide, furthermore, all the devices are connected to the networks which have limited bandwidth available to them. The sum of the speed of every device located in the network might exceed that network's upstream capacity, this is one of the potential bottlenecks that might need to be applied to the set of identified abusable devices, another bottleneck being the maximum upload speed of an individual device.

The maximum real network speed of a specific network connection can't be reasonably measured from a remote observer's vantage point. It might be possible to locate a sufficient number of network services located in some of the measured networks to generate enough response traffic to fill the whole upstream bandwidth. It would negatively affect tested networks potentially triggering counteractions, likely be illegal, and use a significant amount of resources without guaranteeing precise results. Bandwidth in this scenario can be affected by many factors that can't be properly addressed, e.g., network connection saturation, network

path overload at any point, measured device load, activity on the network, time of the day, etc. Because of these limitations remote network speed and bottleneck measurements are not feasible, to produce a DDoS attack capacity estimate other information sources have to be used for calculations.

Academic research into the bandwidth of individual networks is lacking. Leverett and Kaplan solved the bottleneck data issue by using the average upload speed of the Internet connection per country as the maximum speed of individual devices, then applying more precise network speed to the devices which could be extrapolated from the M-Lab data set [4]. This approach doesn't answer the question if the networks themselves are a bottleneck or if they can handle all the amount of the outgoing attack traffic without any issues.

The speed of the network is not a value that can be easily estimated. It can be even hard to draw precise border where individual network connection is, AS as a network differentiator would be the easiest solution. In reality, a single AS can provide a multitude of network connections to different clients with different network speeds, this information might be disclosed through the whois systems but it is not universal.

The main purpose of this methodological step is to address the issue that bottlenecks are present on the network level, it provides no perfect solution how to achieve that. One option is to use external data sources for the limitation and capacity calculations, the choice of these sources is part of the implementation but has to be the same for all the investigated protocols. The alternative option implemented in this thesis and presented in [4. Protocol measurements] is to rely on the measurement itself as the fully self-contained data source, it can provide individual device speed estimates and by excluding rate-limited reflectors the network level bottlenecks can be essentially avoided. Defining and applying these limitations permits to generate the measured protocol's total attack capacity and protocol level BAF.

The goal of this step's implementation is to achieve modularity where implemented bottleneck calculation can be easily replaced by a different one, probably joining with information from other sources. These better sources can arise in the future thus making DDoS capacity monitoring system future-proof, even previous results can be recalculated to be more precise in those cases.

2.8. Notifying network owners

Notification of the network and system owners that devices under their control are abusable for DDoS attacks is one of the main goals for academic and industry scanning projects. This is the practical way how the issue is currently remediated, besides notifying also networks that allow IP spoofing. Commonly notification is done by sending an email containing a list of found devices to the abuse email addresses published in the whois. Whois service is supported by Regional Internet Registries (RIR) and most Internet Routing Registries (IRR) [92]. Depending on the network provided information could even contain the contact email address of the end client using the Internet connection.

Notification can be a part of the methodology with the research goal to not only provide information and knowledge to the decision makers but also to try to contribute to the remediation of the issue on the individual network maintainer level. The proposed measurement methodology can be more efficient than previous attempts because for every

network (AS) it is possible to estimate contributed attack capacity which can be presented to the network maintainer as the amount of wasted bandwidth thus possibly motivating them more to remediate the issue. This thesis doesn't implement a notification stage, the potential for the notification applicability is discussed in [5. Global attack capacity] and [6. Remediating DDoS attacks].

2.9. Ethical and legal considerations

All academic research strives to be completely legal and ethical. Oftentimes cybersecurity researchers cross into the gray zone and sometimes even commit illegal acts. The legal basis of cybersecurity research is not stable and is still evolving around the globe. There are three main aspects to this thesis that require clarification from legal and ethical standpoints to mitigate potential issues:

- Scanning the Internet to find abusable devices.
- Interacting with the discovered abusable devices.
- Publishing the results that might be used for nefarious purposes.

Scanning the whole Internet from a single IP address sufficiently quickly (in 1 day or less) will generate abuse emails to the abuse contact information associated with the IP range. Most of these emails are generated automatically by the systems serving the network IDS role when a predefined threshold is exceeded or a pattern is matched [93]. These emails describe network events corresponding to scanning, might contain part of the network log, request a solution to the event, and demand an answer. Conducting initial scanning from a single IP address clearly established that every single time one or more abuse emails are received. These abuse emails mandated responses. The organization managing the network and responsible for abuse email resolution has to be notified about the activities, before that permission to conduct scanning activities should be acquired as it violates terms of service for most of the networks. One efficient way how to minimize abuse count and other technical risks is to use as many IP addresses as possible and spread outgoing requests equally across them. Most IDS systems will process network activities per IP address thus in most cases notification threshold level won't be exceeded.

Developers of `zmap` and related projects have provided recommendations on how to conduct scans without negatively affecting network operations and maintaining ethical standards for the research [81], [82]. These recommendations are:

- "Coordinate closely with local network admins to reduce risks and handle inquiries.
- Verify that scans will not overwhelm the local network or upstream provider.
- Signal the benign nature of the scans in web pages and DNS entries of the source addresses.
- Clearly explain the purpose and scope of the scans in all communications.
- Provide a simple means of opting out, and honor requests promptly.
- Conduct scans no larger or more frequent than is necessary for research objectives.
- Spread scan traffic over time or source addresses when feasible." [82].

These recommendations have become the de facto standard for most academic research involving Internet-wide scanning and even crawling, but neither they nor any other scanning publications investigate the legal aspects.

Interaction with the devices is separated from the scanning aspect. Scanning for devices using TCP protocols is straightforward as only protocol communication is established and no payload is being sent. UDP scanning and TCP communication are already more intrusive and risky as the devices are instructed to do something and reply but it is still classified as scanning. These instructions usually are safe and normal which generate proper response in all cases in theory. In practice because of different implementations and other anomalies some fringe devices might be negatively affected. From an ethical and legal perspective, no malformed or other payload interfering with the normal functioning of the device can be used.

The interaction stage in academic research usually involves the extraction of additional data with as few commands as possible. The risk is increased in comparison with the scanning stage as there might be multiple requests which are implemented even worse that the scanning ones. There is no detailed research into affecting devices but this risk is acknowledged for some protocols with poor implementation, e.g., DNP3 [94]. The measurements conducted in this research involve dozens or hundreds of identical requests, these requests are the same or similar to the scanning ones and are as safe.

Scanning involves risks, data flowing all around the world, multiple jurisdictions, and fields of law that a researcher has to adhere to. Further interaction has a theoretical potential to overload some individual devices. The author has consulted with legal professionals regarding the legal ramifications of this research. The received feedback has been dissatisfactory as there is no clear established legal practice or precedent regarding this type of research. Thus the author hasn’t taken any additional actions addressing the ethicality and legality of this research.

Publishing the results that can be used by malicious parties for illegal purposes is a serious ethical consideration that has prevented and delayed the publishing of academic security research before. This research contains both methodology and actual data. Malicious parties are already conducting scans to determine which devices are abusable and use those to conduct the attacks. The methodology provided in this thesis can allow readers to gain a better understanding of the contributors to the attack capacity. It might enable malicious parties to achieve larger or more effective attacks if implemented. But, likely, some measurements for this purpose are already being conducted.

Current research doesn’t disclose raw data to the public and identified IP addresses of the devices that are deemed to be significant contributors are hidden as well. Information is presented as a statistical overview and can’t be directly abused.

3. CLASSIFYING DEVICES

Device classification has an important role and numerous uses in both academic research and industry discussed in [3.2. Classification research]. Understanding what is the type or ideally specific model of an unknown device that is present on a network or even more worryingly sending data to the outside is crucial in the modern security context. Albeit there are methods for device classification that can achieve high precision for small sets of devices there were no suitable solutions for the large sets such as millions of DDoS reflecting devices at the start of this research (c. 2017) therefore the author has explored this topic and published contributions to this field. This chapter provides a short overview of the published research [8], [13]–[15], [17], [18] where the author was the lead researcher and its relevance to the DDoS research in the context of this thesis. Refer to the original published papers for a more thorough view.

3.1. Relevance to the DDoS research

After the primary research question “which protocols are contributing the most capacity to the attacks” is answered then a secondary research question arises – “what are these devices”? If it is possible to identify sets of highly contributing reflectors can we classify those and act upon this new knowledge? Can the DDoS capacity issues be addressed from the device angle is discussed in [6. Remediating DDoS attacks].

Although not a part of the currently defined capacity measurement methodology, the classification of the identified highly contributing reflectors can provide valuable insight into the issue. The current methodology doesn’t get into depth analyzing what every device actually is besides only the identified and measured protocol. This protocol might serve only an auxiliary function of the device, e.g., a web server running an NTP server because of some specific requirements or a misconfiguration. Or a residential Internet router that operates a DNS server on both external and internal interfaces. This knowledge can be highly valuable to address the issue at the root cause for every category of found devices. For example, if determined that a significant amount of residential Internet routers are responsible for substantially contributing to the DDoS attacks, while the abused network service might not even be needed to fulfill its functions but is enabled by mistake on the external interface, policymakers can force manufacturers to address it in the newly produced devices and through firmware upgrades.

It is possible to make judgments about every single individual device by analyzing other ports and services running. There are two major issues that caused classification to be excluded from the scope of the proposed methodology. The first issue is balancing between scanning as many ports for every device as possible while not getting blocked or causing issues to the targeted device. The more ports of the device are scanned the more data for classification is extracted. Data quality plays an important role in this research and getting decreasing number of results because of getting detected and blocked would cause more complications than gained insights. The second issue is that creating a large classifier is very

meticulous and time-consuming work and could easily fill a separate doctoral thesis on its own. Without proper training set this task can't be handled using machine learning.

3.2. Classification research

A variety of individual fingerprinting techniques are explored in academic literature and industry products. One of the simplest and oldest ways is to passively analyze TCP/IP stack default settings to determine the operating system [95]. Information disclosure using version identification requests implemented by a protocol running on the device is one of the most common ways to classify the devices but it is usually used against already specific device categories, e.g., scanning the Internet for industrial control system (ICS) devices [94]. Because of the differences between how various protocols and implementations operate and the overall protocol count, it hasn't become the way how all devices on the Internet are classified. Not only operating systems or running software can be identified but physical devices can be fingerprinted as well and identified by their skewed clocks if they are synchronized with time servers [96]. But nothing from the existing body of work can be readily adapted for the current research purposes, a more integrated approach is needed by utilizing the most effective fingerprinting techniques together.

Scanning the Internet for specific devices or protocols is an established practice in security research. This type of research in itself has no novelty in the classification aspect. Assumptions can be made that a device with a known open port corresponding to a non-generic protocol is serving a role that could be easily classified. Further validation by executing protocol communications can be conducted and data potentially useful for classification extracted. This methodology is effective for locating high-impact devices that are running specific protocols (commonly ICS) for the purpose of disabling public access. Mirian et al. scanned the Internet for common industrial protocols while identifying the discrepancy between open ports and the ability to handle respective protocol handshakes [94]. Dahlmanns et al. explore the security issues for the publicly reachable industrial protocol OPC UA [97]. Feng et al. automated IoT classification rule generation [98].

A privileged observer can identify traffic passing through network routers. The basic properties of port and protocol communication can be similar, while active communication requires sophisticated fingerprinting. This approach might allow identifying devices that are not publicly reachable but are actively communicating while at the same time, it might miss devices that are not actively sending packets. Nawrocki et al. utilized IXP and ISP vantage points to identify common industrial protocols while still being challenged by traffic classification [99].

The research into AI classification consists of the same two vantage point approaches. The main challenge becomes identifying features and labeling sufficient training sets. Yang et al. identified and classified ICS and IoT devices extracting features and fingerprints from multiple communication layers [100]. Augmenting this with automated rule generation saved a significant amount of work for labeling the training set. Privileged network observer classifiers commonly are trained on labeled data either from a laboratory network [101] or a campus network [102], [103]. Yadav et al. provide a systematic categorization of ML-augmented techniques for fingerprinting IoT devices [104].

Due to the fact that many AI models follow the black-box approach in terms of result transparency, research in the explainable AI domain has evolved drastically in the last few years. Multiple frameworks such as LIME [105] and SHAP [106] have been developed, aiming to facilitate the implementation of AI in different domains, by providing transparency and trust in underlying models. Explainable AI solutions are already employed in the IoT domain, where low-cost sensors incorporated one in a decision support system [107] and an IoT system [108] generating explanations about the knowledge learned by a neural network from IoT environments.

3.3. Feature selection

Albeit feature selection is a vast research direction in itself in this section only the features relevant to the reflector classification are discussed. As this research addresses only remotely reachable devices the possible feature set is limited to what can be extracted from the communications initiated by scanning or measuring tools.

Network fingerprinting is an old technique of applying static rules to the non-payload component of the communications which may be either passively observed or actively initiated by the fingerprinting tools. This type of fingerprinting primarily relies on the limited quantity of different TCP/IP stack implementation choices. With time variety of devices and their respective network stack implementations have grown exponentially thus decreasing fingerprinting result quality. All of the measured protocols by the scope definition are UDP based thus no fingerprinting can be achieved. Therefore this approach is not suitable for the desired classification.

Most feature-rich sources are protocol communications. A reachable device on the Internet might have multiple protocols running which might or might not serve a useful function just as reflectors. The main issue is that in this research measured protocols don’t provide enough identifiable data in their amplified responses meaning that no classification is possible. Devices might serve other protocols that reveal or leak information that is sufficient for classification and sometimes even pinpointing the device manufacturer and model. But it always requires additional one or more protocol requests for each potential protocol being tested at the time of measuring.

HTTP is a simple generic way of how different kinds of embedded devices can provide an interface (graphical, REST). HTTP features that are suitable for the classification are investigated in detail in [14], [15], [17], [18]. HTTP protocol is feature rich and many of these features individually can serve as a clue to what the device could be – status, protocol version, body, and headers [109]. Although response status codes are standardized the corresponding messages can be customized which distinguishes large groups of embedded devices. If a device responds with an old HTTP/1.0 version response when a modern version request was issued it indicates that this is either a legacy software or a resource-constrained device, in almost all these cases these are low-power low-cost low-resource embedded devices.

HTTP response body can contain keywords revealing the manufacturer, model, or purpose of the device in either the HTML body section visibly or in the HTML head section not displayed in a web browser. Even nondescript textual and graphical elements or external resources enable precise grouping. HTML tree hash can group large sets of embedded

interfaces having textual differences, e.g., interface displaying different languages in different regions or exposed version number, or the current date. Although numerous similarity calculation algorithms can be applied to the textual part or HTML tree the false positives are a guarantee and require significant expert intervention. External resources (e.g., user tracking script) linked in the response body can identify everything else besides embedded devices as these almost never utilize such functionality.

HTTP response headers can reveal extensive technical information that enables users' web browsers to properly process and visually display that response. Although a subset of most common headers are standardized custom headers can be defined (e.g., for debugging purposes) which enables grouping. Lack of, presence and combination of headers can indirectly identify or indicate the use case of the device, modern headers generally are not present on the embedded devices. The server header can directly reveal an HTTP server software but the same embedded software solutions are regularly reused across a wide range of devices. WWW-Authenticate header sent by the server defines the properties of the authentication [110] which only embedded devices and internal web sites use. Cookies can serve a more complex authentication role and commonly are named identifying software names. The date header can identify low-power devices lacking battery-supported clocks.

Redirects to other URLs can identify devices with a unique landing or authentication path. Redirects or direct requests to HTTPS enable to additionally extract data from SSL certificates. Web sites generally have a valid certificate while embedded devices have self-signed ones which are commonly generated using the manufacturer's information.

Features from external sources have proven to be useful. The utilized ones are geographic country and AS mapping extracted from the GeoIP database [111] using the device's IP address. Other potential external features include forward and reverse DNS records which require executing DNS lookup at the time of scans or having access to a DNS historic records database.

3.4. Classifying devices on the Internet

This section provides an overview of the published research [13] where device classification is explored focusing on the web interfaces which are present on various types of devices, from low-impact residential devices to high-impact industrial devices affecting whole regions. In this research, primarily HTTP protocol features are utilized which are discussed in [3.3. Feature selection] - response headers and their values, AS name, HTML tree hash, body title and keywords, SSL certificate issuer, and subject.

3.4.1. Classes of the devices

Distinct sets of device classes have been proposed [100], [103], [112] as every early exploratory research defines classes from scratch. It is expected that with maturity there will be formalized sets of device classes that would allow easy comparison between research and industry tool outputs. For this research, a small set of 10 classes was defined where every class is selected based on role, impact, size of the reachable device set, and historical prevalence based on expert input and previous exploration of the device classes [113].

Device class set definition is a balancing act as these can be viewed from the user, functionality, impact, and observer perspectives. Creating more classes would require a larger and more precise labeled training set without guaranteed improvement of the total overview. Indistinguishably similar behavior even within the small class set is observed because of the HTTP protocol genericness requiring a special class for these devices. Same time some of the proposed classes have small subsets of devices that vary drastically in their behavior and specific purpose. Although the labeled set is significant and proportional to the whole data set, it is not sufficiently representing various rarer devices and subclasses to train the classifier. Which combined with hard-to-distinguish protocol responses would introduce even more uncertainty. These issues can be mitigated by augmenting data sets with features from other protocols.

ICS class contains the most impactful devices which can affect not only individual users but potentially whole regions. It includes industrial control systems, SCADA, and building automation devices. The role and software vary drastically for devices in this class. Through significant scanning and notification efforts, the number of reachable devices has fallen.

Network devices are classified as the NET class which includes all the wired and wireless devices used in individual residential installations and most of the devices serving a more significant role on the network, providing connectivity to organizations and other networks, primarily these are routers, switches, and firewalls. The impact of attacks on these devices cannot be overstated as not only detectable network interruption but hidden MITM attacks can be executed. Other devices in this class include network storage, television, and streaming set-top boxes. INFRA class encompasses data center infrastructure devices affecting the physical properties of the server hardware. These are high-impact devices providing server control panels and virtualization solution control panels.

Although a variety of IoT devices from the serving role viewpoint is significant, all of these are placed in one IOT class. The ratio of IoT devices connected to the Internet versus those directly reachable is lower than for most of our other classes. It can be explained by the different ways different devices are connected to networks.

Historically prevalent device classes PRINTER, IPCAM, and VOIP are kept separate. These classes had historic public mass attacks that negatively affected a large number of people, e.g., wasting toner on printing unwanted documents, leaking private video feeds. Thus their reachability should have decreased over time. IPCAM class includes not only IP cameras but also DVR and NVR devices providing recording and viewing functionality. PRINTER class includes printers and network print servers. VOIP class includes phone sets, conferencing solutions, and VoIP gateways.

It is possible to determine with a high likelihood that a specific device is not a generic web server. Features like unsupported HTTP protocol version 1.1, the wrong clock which starts to count time from Unix 0 seconds, and the lack of any headers indicate custom or outdated server software that is usually an embedded device and only in rare cases serves a generic web server role. If response features are insufficient, these devices are classified as UNCLEAR. This class also includes manufacturers that are represented in multiple classes, but no clear dominant class is established, and it is not possible to distinguish the device class from the response alone, e.g., the same web interface is re-used across classes. All the

remaining cases, where it is not possible to confirm that the device is not a generic web server, are marked UNCATEGORIZED.

From the security research perspective, generic web servers hosting various web applications are often the least exciting class of reachable devices. These devices are much more often properly managed and automatically updated as they are usually deliberately reachable. The most vulnerable parts of these devices are web applications themselves, not the HTTP servers, but these applications in most cases are reachable using the domain instead of the IP address, which involves a different kind of scanning. There are web applications that are configured to process requests received without the domain name, but quantity-wise they are a minority. All generic web servers, web applications, and services related to these, e.g., CDN are placed into the WEB class.

3.4.2. Classifier

Four data sets were created by scanning the Internet using scanning tools commonly used for research: `zmap` and `zgrab`. Both HTTP default port 80 and common alternative port 8080 were scanned in December 2018 and one year apart in December 2019. Up to three redirects are being followed to any port including HTTPS, in which case TLS negotiation is being saved as well. For the standard port in 2018, there are 54,811,827 elements, and in 2019 there are 57,131,825 elements. For the alternative port, there are 7,792,077 and 8,100,201 elements, respectively. An element is a single response or response redirect chain corresponding to a single request that contains at least one proper HTTP response.

The labeled set consists of 171,791 elements. It was created from random elements of the 2018 port 80 data set and therefore is unbalanced across classes. There are 132,562 WEB, 22,002 NET, 9561 IPCAM, 711 INFRA, 265 VOIP, 243 ICS, 218 IOT, 153 PRINTER, 4175 UNCLEAR, and 1901 UNCATEGORIZED devices in the labeled set.

Two models were trained - one with the full labeled data set (large) and one balanced model (small). Comparing their accuracy (about 87% for the small and 97% for the large data set), by randomly sampling the classified output of the whole data set it can be noticed that the small model performed better due to the bias in the large data set. As the full labeled data set primarily consists of WEB devices, the classified output is significantly skewed towards classifying devices as WEB. To avoid bias of overrepresented classes in the labeled data set, a balanced labeled training set is employed (in total 11,479): ICS:243, INFRA:711, IOT:218, IPCAM:1,999, NET:2,000, PRINTER:153, UNCATEGORIZED:1,901, UNCLEAR:1,999, VOIP:265, WEB:1,999. The labeled training data set was divided into a training set (5,628), a validation set (2,413), and a test set (3,447). The test accuracy is 0.87277. Neural network selection justification, model training, and workflows are outside of the scope of this thesis and are discussed in detail in [13].

3.4.3. Results

The model was trained using the 2018 standard port labeled data set and applied to the 2019 standard port data set and also port 8080 data sets for both years. Although the reachability of devices has been recognized as a poor and high-risk management practice, there is an increase in the data set size in 2019.

The standard port 80 classification results are provided in figure 3. As expected from labeled set WEB devices are the most prevalent ones. What was not expected is that the UNCLEAR and UNCATEGORIZED devices will be so numerous, but that can be explained. UNCLEAR and UNCATEGORIZED devices often have a small set of rare features extracted from the HTTP responses, which makes classifying them even manually challenging (requiring aggressive service enumeration combined with using external sources) and in many cases impossible. Although while creating the labeled set many of these devices were categorized, it was done through numerous weak rules utilizing only the available features. These features might be rare and unique enough that are not applicable to the whole data set in which case HTTP response data on its own might not suffice for accurate classification.

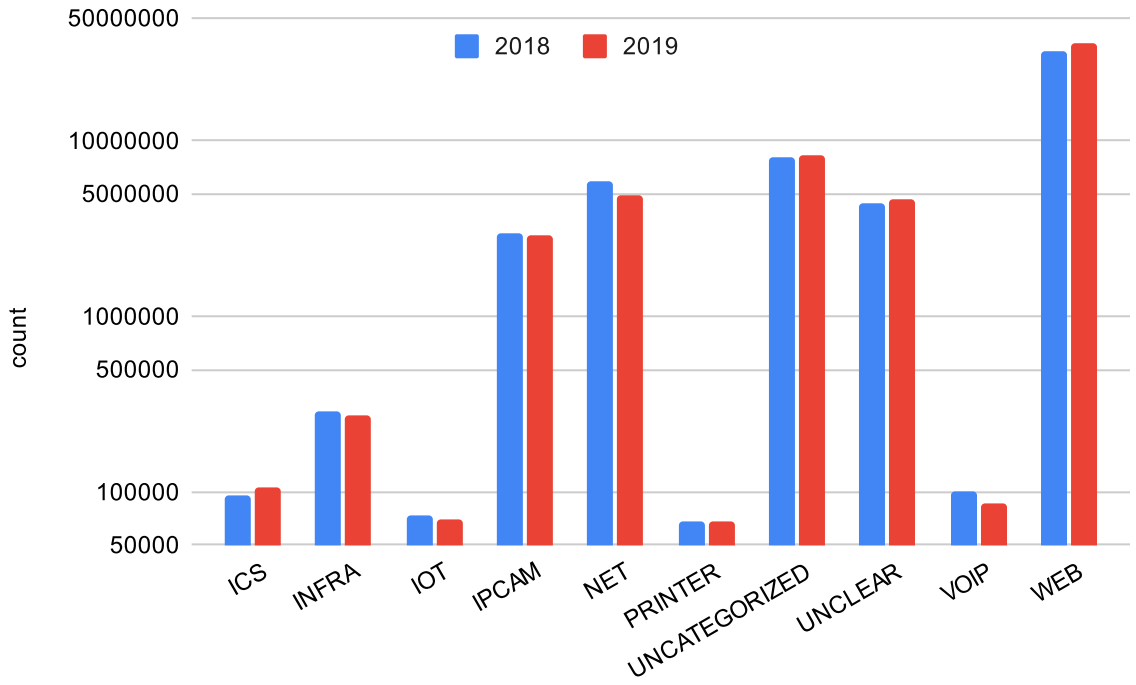

Figure 3: Distribution of device classes for port 80 for 2018 and 2019

We can observe a slight decrease in reachable INFRA and IOT devices in 2019. As the number of IOT devices is growing significantly, it would be expected that the number of these reachable devices would grow over the one-year period. However, this class of devices is the only one from the defined one that historically rarely could be connected in a way that made them reachable. A more significant decrease in VOIP could be explained by changes in the way these types of devices are deployed and managed on a vendor level.

From the publicly well-known attacks targeting IPCAM and PRINTER devices, it could be expected that the number of reachable ones would decrease significantly as sufficient mitigation strategies were employed, but no such trend is observable. One explanation is that the number of newly added reachable devices closely matches the ones that were mitigated. It is currently not clear what portion of these almost 3 million IPCAM devices have to be reachable for remote surveillance and recording purposes.

A large number of NET devices were expected. In a residential Internet connection, the device can expose the control panel to the Internet even if the initial setup is done by the ISP

technician. A significant drop in these devices might suggest that the device life cycle could be playing a role in older ones getting replaced and newer ones having a better configuration.

The alternative port 8080 classification results are presented in figure 4. As expected, the WEB devices are proportionally smaller class than in the port 80 cases as generic web sites usually reside on port 80. UNCLEAR and UNCATEGORIZED being the largest two classes and having significant growth over the one-year period might suggest that the feature difference is significant enough between the two ports that the model needs to be augmented with the alternative port data as well. We can observe much more significant proportion changes among the classes on the alternative port.

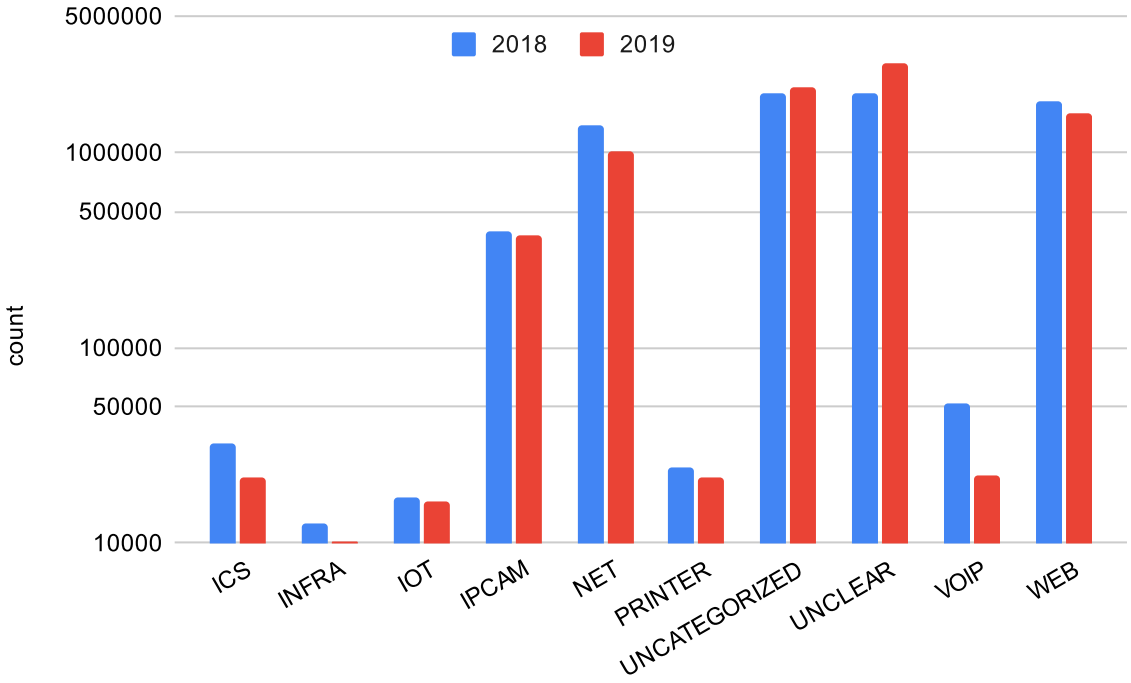

Figure 4: Distribution of device classes for port 8080 in 2018 and 2019

The relative class distribution for all four classified data sets is presented in figure 5. This view enables us to make a comparison between the utilization of different devices on different ports. There are other discernible differences besides already identified WEB, UNCATEGORIZED, and UNCLEAR classes. INFRA devices are proportionally about four times less prevalent on the alternative port, and it could be explained by the fact that there are a small number of manufacturers whose devices were identified and labeled on port 80. These devices might be using the default port setting, and there might be unidentified INFRA devices defaulting to 8080 port.

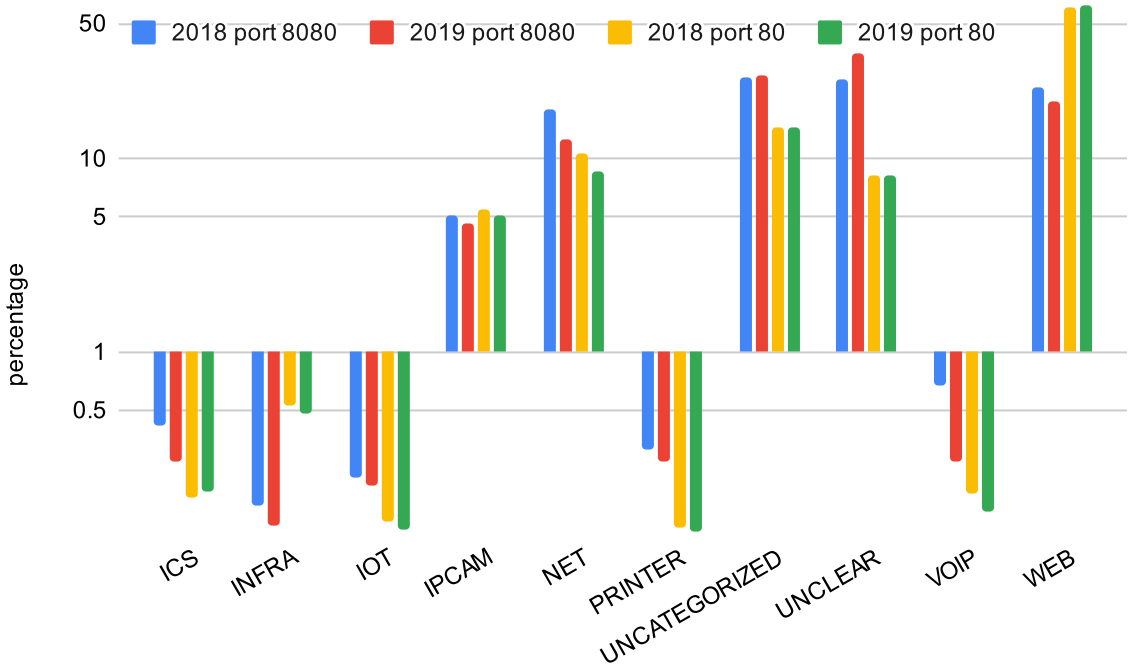

Figure 5: Proportional distribution of devices for port 80 and 8080 in 2018 and 2019

Interestingly IPCAM has almost the same proportion across the ports with the same decrease over the one year. Proportionally there are significantly more PRINTER devices on the alternative port, and that is explainable by the high variance of device models and default configuration even within a single manufacturer. VOIP, ICS, IOT, and NET devices are proportionally more represented as well on the alternative port. This might be the result of manufacturers' concerns about creating port conflicts on a single device. Especially valid this concern is for NET devices which are handling networking traffic and possibly forwarding the port 80 to another device.

3.5. Explainability

This section provides an overview of the published research [8] focusing on the explainability of the device classification which builds upon [13] overviewed in [3.4. Classifying devices on the Internet]. Classification explainability in the device context enables bidirectional knowledge exchange – experts can formalize their professional intuition into additional static rules for rule engines by reviewing classification decision weighting, while simultaneously learning new indirect ways how non-straightforward devices can be classified.

3.5.1. Classifier

The data set was prepared in the same way as described in [3.4.2. Classifier]. For the standard port, there were 51,118,537 elements, and for the alternative port, 8,343,898 elements. Labeled sets and class definitions were re-used as well.

The LIME framework was selected for the explainability of the implemented Naive Bayes classifier. It provides rational numbers and associated features as text, which enables

the human interpreter to understand if the feature was weighted positively or negatively for each device. The test accuracy is 82%.

Neural network and Naive Bayes comparison, implementing explainability, and technical considerations are outside of the scope of this thesis and are discussed in detail in [8].

3.5.2. Understanding classification

Explainable classification can increase the precision and furthermore transfer the new knowledge back to experts. In this subsection, a randomly selected device from each class is evaluated in an attempt to understand the classification and to evaluate options for improving it. The calculated prediction of classes and the most impactful weights of the features determining the likely classes are presented.

Cisco IP telephony device classified as VOIP is presented in figure 6. While an expert would focus on keywords “Cisco” and “SPA”, the classifier selects “spa” as the highest weight feature and disregards “cisco” manufacturing a large variety of NET devices. While authentication headers are more indicative of other lower power and cheaper devices and have negative weight in this case, it is counterweighted by a slightly more complex and secure variant instead of plain text.

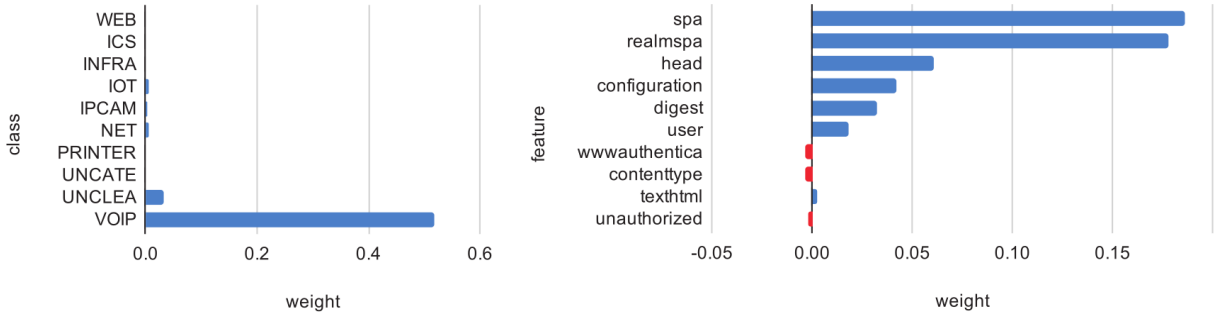

Figure 6: Class predictions and feature weights for the VOIP device

The PRINTER device is presented in figure 7. The highest weight features “hp” and “officejet” correspond to one of the most common printer series covered by most static rule sets. Feature “broadband” is weighted negatively as is more expected in the context of networking devices. The “finance” keyword is part of the network name feature, which is not common in other randomly reviewed devices.

Figure 7: Class predictions and feature weights for the PRINTER device

Smart home automation device from LOXONE classified as IOT is presented in figure 8. While none of the high-weight features is identifying the vendor or model which is the way an expert would write a static rule for this device, the highest weight has “webinterface”

keyword of the interface and response headers. Coincidingly security headers are weighted negatively, indicating that the model expects IOT devices to have lesser security features. Interestingly network name feature consisting of “austria” and “telekom” indicates that the manufacturer based in Austria has a high presence in Austrian networks. While this can be intuitively recognized by an expert, the variety of devices and complexity of the rule has prevented this from being implemented in static classification rule sets.

Figure 8: Class predictions and feature weights for the IOT device

VMware Horizon device classified as INFRA is presented in figure 9. By definition of the class, most VMware solutions match the INFRA, thus keyword “vmware” having high weight is expected, as well as all other classes assigning a negative value to it. The product keyword also is expected, static classification rule sets might contain a simple rule matching these two keywords together.

Figure 9: Class predictions and feature weights for the INFRA device

From the randomly selected devices, the spectrum analyzer has the least features and is classified as ICS and presented in figure 10. ICS devices can have the least properties of the responses that can be extracted as features. Rich response features typically weigh heavily against the device being classified as an ICS. While the combination of “spectrum” and “analyzer” can be evident for humans, these are treated as separate features and spectrum weighting against this class while weighting heavily in favor of some other classes. It identifies an issue of introducing network names as a feature, in this case, likely the large ISP named Spectrum, suggesting that the network name feature should be treated differently from the response features.

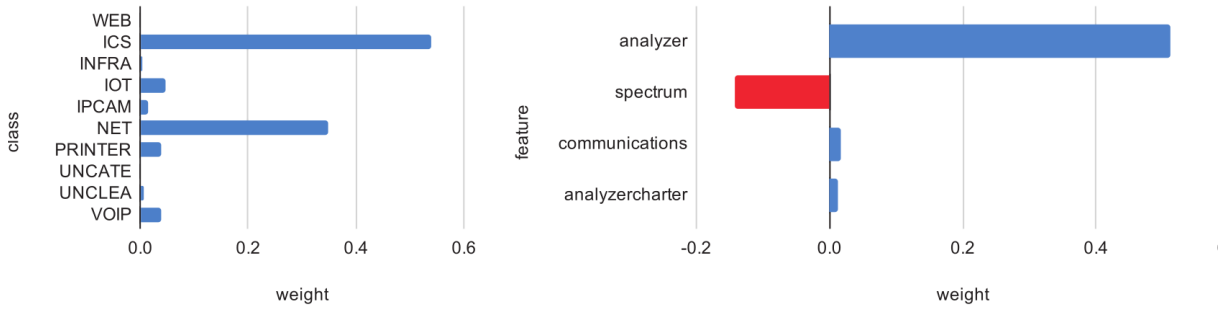

Figure 10: Class predictions and feature weights for the ICS device

For the network device presented in figure 11 (NET class) the second highest weight feature “gateway” is a classic keyword even in static rules. The feature set and raw response confirm that this is an unbranded residential network gateway for which even an expert is unable to extract more information without active probing. This feature is not unique to the NET class. It might correspond to gateway functionality in an application protocol sense or display configuration debugging information for any networked device. In this particular case, this feature is weighted in favor of only the VOIP class. Most of the remaining determining features consist of authentication interface keywords, including the highest weight feature “incorrect” indicating failed authentication. The way how an authentication interface is presented has a high weight in determining the class.

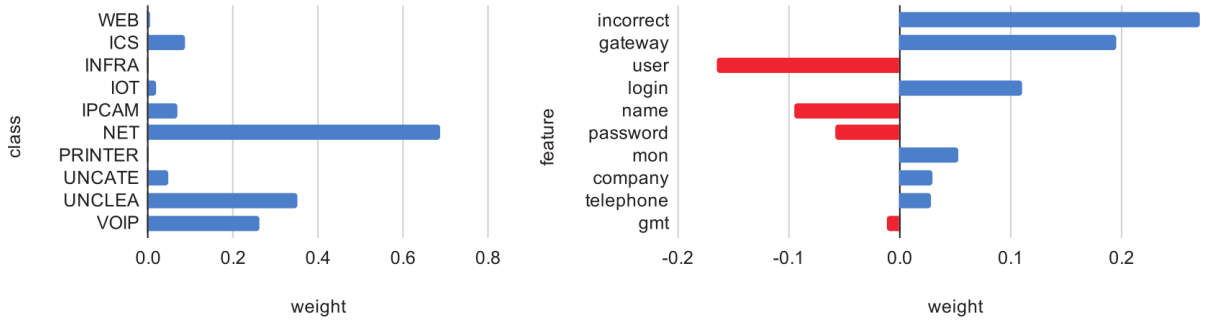

Figure 11: Class predictions and feature weights for the NET device

Hikvision networked surveillance device classified as IPCAM is presented in figure 12. The “dnvrswebs” is a software version unique to IP cameras and video recorders and thus is weighted heavily. In general, it is weighted negatively against all other classes. Most static rule sets have this as a simple match rule to reliably classify IP cameras.

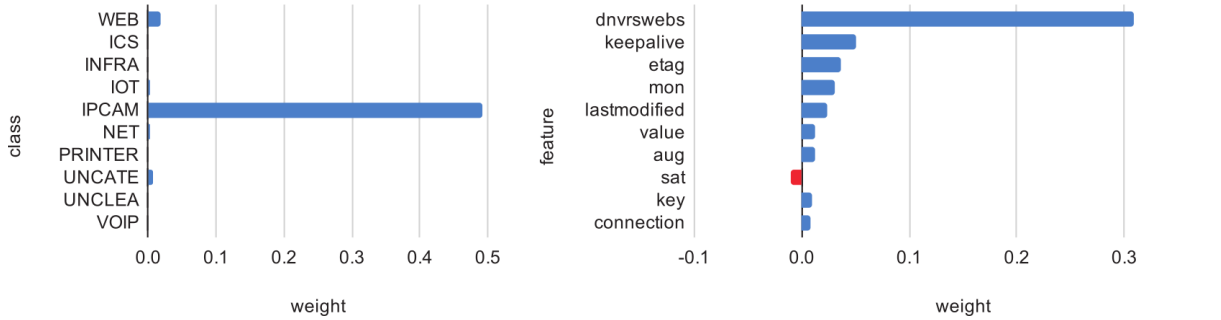

Figure 12: Class predictions and feature weights for the IPCAM device

The WEB device presented in figure 13 is an Apache Tomcat interface allowing deployment and management of web applications. While the feature “apache” has a high weight in determining the class it is not always the case otherwise a blanket static rule would

suffice. In general, it has a negative weight on the UNCLEAR class where no web sites are expected. The keyword “restricted” generally associated with web interface authentication has a significant negative weight.

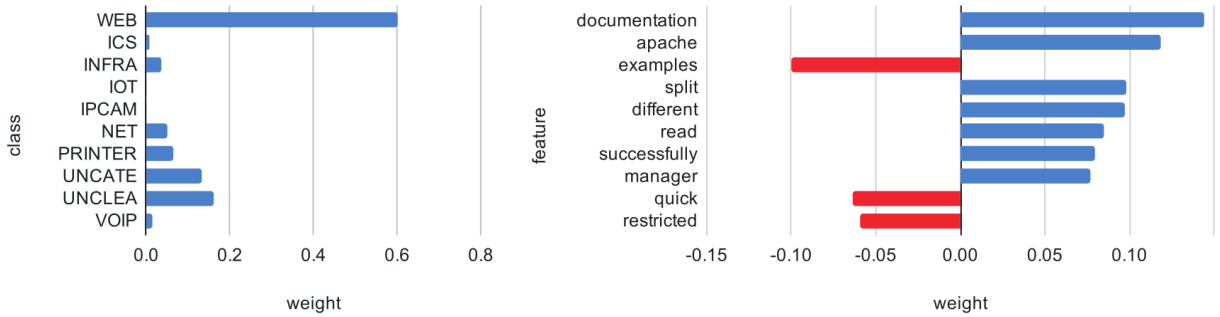

Figure 13: Class predictions and feature weights for the WEB device

A device classified as UNCLEAR likely being an embedded one without a determinable functionality but definitely not a generic web site is presented in figure 14. Keywords related to HTTP basic authentication and the displayed message are weighting in favor of this class. While the presence of the Server header revealing software name and version is weighting against as it is often a high-weight feature, in this case, it is a generic embedded software having many uses.

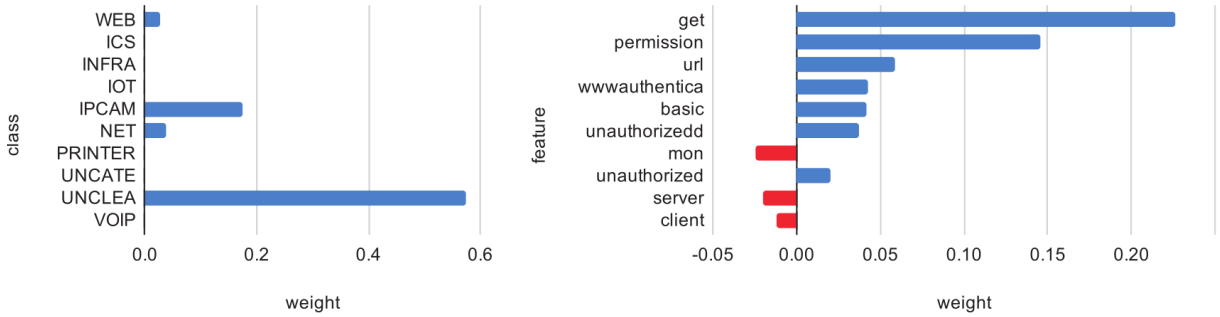

Figure 14: Class predictions and feature weights for the UNCLEAR device

The UNCATEGORIZED device that can’t be determined to be a part of any class is presented in figure 15. This device has a small feature set (all being generic) but not as small as the most basic embedded devices. The generic response headers are sufficient to be also of a web site or service not handling the default request. In general, plain text content type, which is the heaviest weight feature, corresponds to unformatted output mostly short error messages. From this set of features, an expert is not able to reliably determine the class either.

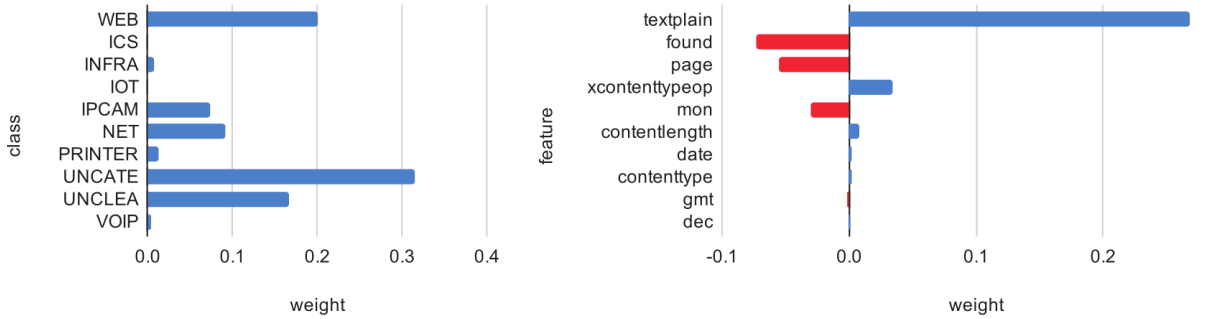

Figure 15: Class predictions and feature weights for the UNCATEGORIZED device

While there can be identified cases that are common and covered by static classification rule sets even within these few random examples more complex classification matching

expert intuition can be seen. These types of cases individually can be classified by an expert but defining all of that into static rules is not feasible, not only by the sheer number of rules but also by the complexity which would require statistical calculations to formalize the intuition.

3.5.3. Class distribution trends

The relative class distribution is presented in figure 16, the Naive Bayes classifier results developed in this paper are prefixed NB. The remaining classification data are based on neural network results from [13], the raw scan data from the same source is used to test Naive Bayes classification for years 2018 and 2019, while the 2020 data set has been created specifically for this research.

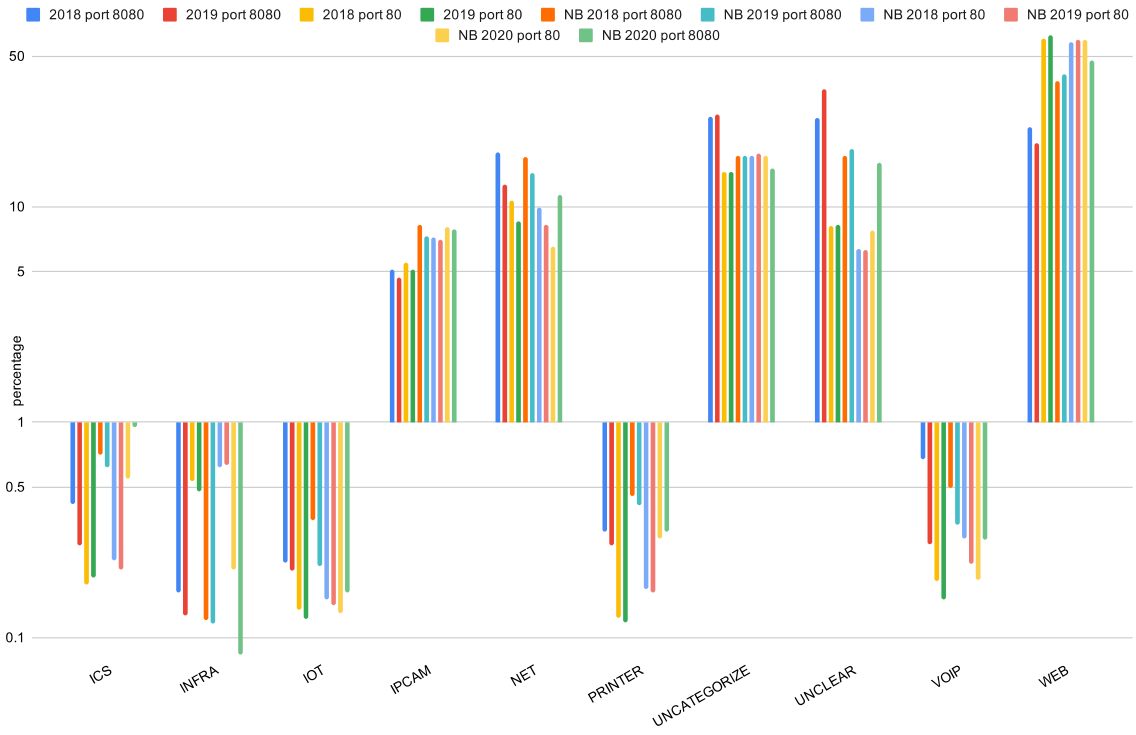

Figure 16: NB classification applied for 2018-2019 and neural network classification

While classification differences can be easily observed, they are explained by varying accuracy ranges between different methods. Although the goal of this research wasn't to analyze the classification main trends of Naive Bayes classification can be observed. The increase in ICS devices is unexpected in light of worldwide efforts of disconnecting these devices from the Internet, most likely these are new deployments of low-impact automation devices. The decrease in INFRA devices is expected with a shorter lifecycle of deployments and new deployments following better security practices. The stable proportion of IOT devices is positive, considering the increasing number of new deployments. IP cameras which often require remote reachability see a slight increase, in contrast, to NET devices which don't see a significant decrease. IP telephony devices experience a stable decrease.

3.6. Conclusions

Although [3.4. Classifying devices on the Internet] and [3.5. Explainability] provide a functioning device classifier and furthers understanding of the classification decisions and therefore has contributed to the overall device classification field with the respective publications [8], [13], the produced classifier is insufficient to achieve high precision results for the reflector data sets which is needed for the analysis and remediation discussed in [6.2.3. Devices and regulation]. The main reasons for this are that only a small portion of the reflectors have an HTTP interface, reflected UDP packets lack features suitable for fingerprinting, and the classifier's inability to utilize other protocol banner grabs.

The author anticipates universal device classifiers meeting the requirements of this research to be released in the coming years and gathers data to the extent of not affecting primary scans. This is done during the scanning phase and only when devices respond to the measured protocol requests. Banner grabbing is conducted using `zgrab2` [114] which is a part of `zmap` network scanning toolset [84]. Currently, only a small subset of commonly used protocols that have the potential of information leakage (banner grab) via the first and only request is interacted with: FTP, SSH, Telnet, POP3, HTTP (including alternative ports 8080, 8000), and HTTPS (including alternative port 8443). When the required classifier functionality level is achieved it will be trivial to apply it to the existing data sets providing historical views and changes over time. The main risk is that the awaited new generation of classifiers might require more features (e.g., wider coverage of protocols being banner grabbed) to be accurate which won't be retroactively collectable.

4. PROTOCOL MEASUREMENTS

This chapter describes how the methodology proposed in [2. Attack capacity measurement methodology] is applied to real-life scenarios and discusses technical implementation details. Individual protocol measurements and produced results are explored as well, for each protocol a single measurement is analyzed. Changes over time are reviewed in [5. Global attack capacity].

The protocol set selected for the measurement and analysis in this chapter primarily represents the ones that historically have been abused and are still relevant in the current DDoS attack landscape. Selecting NTP, DNS, SSDP, and SNMP enables comparing produced results with the only other existing methodology [4] addressing global attack capacity in [5.4. Measured vs. theoretical capacity]. Picking additional CLDAP and Memcached protocols results in having 6 out of the top 7 most abused protocols observed in an IXP [39]. This list is not meant to be exhaustive thus protocols that have lost their prominence (including short-lived recent ones) in the academic and industry sources are not measured.

4.1. Implementation overview

This section provides an overview of the implementation and setup used to produce measurement data for the selected protocols in this thesis. It doesn't cover the evaluation of different technical decisions made over the years, conducted testing of dedicated hardware vs. different virtualization platforms, or evolution of the code base. Initial published measurement results [10], [16] were based on the data from a geographically different measurement point which had a different technical setup and an older code base. Making these older results incomparable to the ones presented in this thesis.

4.1.1. Technical setup

For the scanning and measurement purposes various virtualization and hardware setups, OS, and network setups were tested. The measurement results presented in this thesis have been acquired on a dedicated server as it proved to be the most stable and reliable setup. The network setup consists of a dedicated 1 Gbps uplink and 16 non-continuous IP addresses. Network and ISP are known not to automatically filter out network traffic that resembles DDoS attacks. ISP was informed and it accepted the use of its network for this research. The dedicated server had 32 GB of RAM and 4 dedicated CPU cores using Ubuntu 18.04 LTS OS. Server resources were monitored and if were unexpectedly exhausted (e.g., anomalous network loops exhausting storage or filling incoming network link) the affected data sets were discarded which is discussed in [4.9. Data quality].

While conducting scans small number of automatically generated abuse e-mails were received by an ISP but because of the company policy, those were not forwarded uncensored, and thus no blacklisting action could have been taken by the author. The author implemented scanning best practices discussed in [2.9. Ethical and legal considerations] and has published the research explanation containing an easy way how to manually opt out from further scans. In the time period of the final network setup from May 2020 to March 2023 surprisingly only

10 manual opt-outs requests were received. Opt-outs totaled 8455 IP addresses which is an inconsequential count compared to the scanned Internet IP address space thus no changes in calculations were made to address those. It indicates that selected scanning speeds are sufficiently low not to trigger a widespread manual review by network operators.

Data extrapolation from opt-outs and possibly network bans might be an interesting research question if received opt-outs or detected network unreachability would be significantly more common. It exceeds a simple technical solution as one could reasonably argue that networks that are monitored and take action against scanning might also address the reflectors on their own networks.

Properties of the measurement network and the closest ISP and IP transit provider are crucial for the measurement activity. The primary concerns that are easily addressable are firewalls and Intrusion prevention systems (IPS) on the network, which might detect the UDP traffic and classify it as an attack, therefore, blocking a single IP or the whole protocol. The preferred way is to route network traffic directly from the measurement system to the Internet without any intermediary security devices. Issues with the Internet providers are complicated, most ISP have experienced DDoS attacks and therefore have some mitigation in place, IP transit providers handle DDoS attack traffic continuously and often are tasked with mitigating it. The primary goal of the measurement is to cause a small network flow using the exact same requests as real-world attackers and receive apparent DDoS attack responses. This measuring can resemble patterns of real attacks and trigger an automated response from the ISP or less likely IP transit provider. Not all commercial networks can be used for this type of research as some have aggressive policies towards UDP traffic which gets limited or dropped without even triggering any thresholds, intermittent mitigation efforts can affect data quality which can't be easily addressed but can be detected as discussed in [4.9. Data quality].

OS network stack is being used for the measurement stage and thus requires some tuning albeit minimal compared to what TCP would require. Linux network buffer settings are quite low by default and depending on the load UDP receive and send buffer sizes might need to be increased. For the purposes of the current research, only the following changes (16MB send and receive buffers) sufficed in every tested and experimented scenario, these settings are stored in `/etc/sysctl.conf`:

```
net.core.rmem_default = 16777216
net.core.rmem_max = 16777216
net.core.wmem_default = 16777216
net.core.wmem_max = 16777216
```

4.1.2. Scanning

For all the measured protocols scanning step is implemented the same way:

1. `zmap` is launched with the appropriate port, settings, and payload fine-tuned for the measured protocol.
2. Responses from `zmap` are piped in real-time into the custom measurement script written in python.
3. The script determines if an initial response from `zmap` satisfies the minimum conditions for the measuring.

4. Script sends amplification-causing requests, and logs or processes responses to detect amplification and rate limiting. Amplification and rate limiting measurement can be completely joined into a single action.

For the scanning with `zmap` IP source address range is being passed. Arzhakov et al. suggest that the speed of scanning has created a new challenge for the generated data quality as it has created a situation where scanning is so fast that it can be detected and blocked by IPS [115]. To address that a conservative rate limit of 45000 PPS is selected which allows one to scan the whole Internet in approximately 23 hours. One thread is being dedicated to the scanning to meet the performance requirements and leave the remaining CPU cores for measuring and logging. UDP module and appropriate port are being used, protocol payload has to be supplied, and output format containing not only IP address and port but also received data response and destination IP address is being used. Responses from `zmap` are piped into the load balancer which in turn pipes them to separate measurement scripts (one for each used source IP address). It enforces association between the scanning and measuring source IP address but more importantly, it permits to use simple python parallelism concepts that are bound by global interpreter lock to a single CPU core. As the launched measurement scripts are completely independent processes then as many CPU cores as the number of processes can be utilized without any added programmatical parallelism complexity. Example scanning and measurement script launch command:

```
/root/opt/sbin/zmap -i $INTERFACE -S $IPS -T $THREADS -r $RPS -b
/root/DDoS-Capacity/zmap.blacklist -M udp -p $PORT $PROBE -f
saddr,data,daddr,classification,dport -O csv -o - 2>
$OUTPUT_DIR/zmap.stderr | python3 $DIRECTOR $IPS python3 measure.py
$MEASURE_PROTO $OUTPUT_DIR
```

Bandwidth utilization is protocol payload dependent and therefore can be easily estimated as `(overheads+payload)*45000` and is essentially constant for the whole duration of the scan. At the selected PPS rate outgoing scanning consumes approximately 80 Mbps for the largest used payload of 192 bytes for the NTP protocol.

One request per every IP on the Internet is sent, if the request packet is lost or the response is sent but not received then that specific device is not discovered or measured as all of the measured protocols use UDP. It is possible to optimize network utilization and scanning speed by sending packets only to prefixes visible through the BGP table but the scanning is already designed to be slow and in this scenario, no gain is expected. This optimization would be beneficial only for a fast snapshotting scan. Developers of `zmap` state that the number of lost packets are less than 2% [82]. The goal of the scanning and measuring step is to eventually be automatically scheduled and executed thus even some temporary network issues will not negatively affect the overall measurements over time. The scanning stage relies on the default `zmap` behavior to randomize scanned IP addresses to minimize detection and to provide output of the detected services distributed evenly across the whole scanning time thus enabling simplified measurement implementation without buffering and queues.

The code of `zmap` is extremely efficient by design and consumes all the CPU cycles available on the system. This creates contention for the CPU and can cause completely unnecessary delays for the software running in parallel on the same system which usually is none. But in this research on the same system as `zmap` at the same time measurement script is being executed which expects no delays for timing calculations. The solution for this is to use

`zmap` thread limit and leave at least one free thread for the measurement meaning system has to have access to 2 dedicated CPU cores to produce reliable data. For the technical setup and selected configuration, it suffices to limit `zmap` to a single thread (core) as the scanning speed is low. Every new technical setup and rate limit combination must be tested to guarantee that the selected thread limit is sufficient.

Measurement scripts are written in python as it is a universal language allowing to implement algorithms easily and computing overhead is not an issue as the amount of processing that has to be done is significantly lower than full Internet scanning which is done by `zmap`. Measurement starts instantly without any artificial delay or queuing. That can be achieved because discovered host per scanned IP ratio is low, the scanning speed is artificially limited and destination address randomization is employed.

Snapshotting the state of the Internet would provide the best view of the situation, and a good approximation is possible [83]. Issues with scanning are discussed in [2.4. Scanning the Internet], for this research primary requirement for the full processing is defined as it must be completed in less than 24 hours.

4.1.3. Rate limit detection and measuring amplification

Considerations regarding rate limit detection are discussed in [2.6. Detecting rate limiting], these range from easy implementation producing low data quality to high data quality which requires complex implementation. For the purposes of current research middle ground is implemented the same way for all the measured protocols. In the implementation port association with the tested IP address and request number doesn’t have to be explicitly maintained, it is implicitly implemented.

A specific non-privileged port range is selected and for every measured protocol maximum number of requests needed for the measurement is determined. The port range is split into sets of ports equal to the count of required requests (50 or 100 requests) for a single reflector measurement. In these smaller sets ports are consecutive which allows to send requests from these ports consecutively, it allows to determine which request didn’t receive the response. These associations are implemented through simple arithmetic operations.

An important aspect is to select a sufficiently large port range for measurement activity that doesn’t overlap `zmap` scanning activity. By default `zmap` evenly utilizes port range 32768-61000 [116] for the scanning stage, this large range permits matching between sent and received packets even when the IP addresses of received packets differ from the targeted ones. For the purpose of this research port range 2000-32000 is being used for the measurement stage. It must be ensured that these ports are not already in use by other software on the measurement system.

Whenever a load balancing process receives a new IP address from the scanning process it pipes it into the appropriate measurement process based on the response destination IP address. The selected measurement process verifies that it has to be measured, measuring starts instantly by sending a burst of request packets as fast as possible. The time the burst takes is short and while being affected by the system load and other factors it is proportional to the number of packets being sent. Burst takes less than 1 millisecond for 100 packets sent and for the calculations is assumed that it is instantaneous.

A potential drawback of the implemented measuring solution is that a different source port is used for every single request, although rate limiting works per IP address basis not per port, attackers in most cases still choose to target specific service ports [73], which actually is a theoretical concern in a sense that this is a difference between how attackers usually operate and how measurements are conducted. The author didn't observe any noteworthy differences in the testing stages for the different port utilization.

A timeout of 10 seconds for every reflector measurement was selected through testing and empiric observations. In the local network testing setup all measured responses were received in less than 0.1 second. In most real-world cases 1 second could suffice but to cover borderline cases, anomalies, and intermittent network issues larger value is used. Timeout has to be balanced against scanning instance count and speed of scanning so as not to overload measurements, at any given point in time no more than concurrent reflector measurements can occur:

$$maximum\,concurrent\,measurements = \frac{instance\,count * port\,range}{ports\,per\,test}$$

For the most common value of 50 measurement requests with 16 instances (measurement IP addresses), `30000*16/50=9600` concurrent measurement can be executed. Either as a burst or averaging `9600/10=960` measurements per second. That would mean `960/45000=2.13%` hitrate from the scanning stage is the limit (within the sliding window of the timeout) that can be handled by the measurement stage. None of the protocols measured in this thesis approached this limit.

Alternative implementations would complicate processing as separation by the source IP address then would be required. If responses were always perfect and received on time from the IP addresses they were sent to, overload wouldn't be an issue and the measurement rate could be as fast as technically possible. But the Internet is not perfect, besides routing anomalies, backscatter traffic, and delays, there are scenarios in which responses are coming from different IP addresses than requests were sent to, e.g., multiple IP addresses per single device. Without overload, it can be assumed that the IP address mismatch doesn't affect measured results. With overload happening only effective strategy is to exactly match the IP address of the sent packets with the IP address of the received packets.

Cyclical use of the port range significantly simplifies measurement implementation by removing the need for any queuing. For debugging purposes notification is provided when overload happens so fine tuning can be conducted or data discarded if it is tainted enough. To an extent undesired received responses from the timed-out reflector measurements are mitigated as well because the average rate of active measurements is significantly lower than the maximum measurement rate. Within this time buffer old response packets are accepted but are actively discarded.

As for the measurement stage IP address association from the scanning stage is maintained and there is no explicit delay between scanning discovery and the measurement initiation, the rate limit counter might be already active on the device. Meaning the measurement response count per reflector will differ by 1 if the rate limit is present and triggered.

The selected design choice allows the author to combine this step with amplification measurement easily. As the requests used for rate limit detection are numbered it is possible to

determine the reply for the first request sent. This first response is measured in bytes and packet count, for additional debugging and investigation purposes these first response payloads are saved. There are various scenarios in which packets get lost or don't generate maximum amplification in the first response but these are rare and therefore not handled in any particular way.

Choosing the output format and amount of data to preserve from the rate limit detection and amplification measurement is a balancing act. A raw network traffic dump produced by `tcpdump` is useful for any kind of network research in the initial stage including this thesis. A common way how to use `tcpdump` for Internet measurement is to conduct random sampling to debug measurement software and investigate detected anomalies at the network level. For this research `tcpdump` is used in real time to pre-process incoming measurement response packets as it has proven to be more efficient and reliable than implementing raw sockets in the python measurement code. And to save raw packets for post-processing. But a single full Internet scan dump would contain at least the sent more than 3 billion packets, all the received replies, and backscatter. It is not rational to save these full dumps for all scans. Only incoming measurement packets are saved instead. The unsaved data can be partially recovered from various logs generated throughout the whole process if needed.

The primary goal of the output is to be easily processable for analysis. Some processing of the raw data is being conducted to simplify the processing and minimize the output size which is generated in JSON format. The selected JSON format allows to import data in any programming language and most data analysis tools which leave options open for exploration and testing.

For every measured reflector an object is created, it has the IP address, start time (seconds since epoch as floating point number, precision is system dependent), and an associative array containing all the measured responses. An anonymized single reflector measurement output in the JSON format is presented in appendix A. Keeping packets from mismatched IP addresses while still verifying IP addresses simplifies further processing and enables anomaly investigation when needed. For every response IP address packets and bytes are counted for every measurement request while preserving sequence, allowing aggregating statistics per IP address and investigating response patterns. The time of the first and the last packet received is also preserved. For every response IP address, all the response payloads for the first request are being saved by transforming those into hex for safe handling across different stages of processing. These payloads permit the investigation of responses on an application level, network level properties of the packets are not preserved in the JSON data set.

Selected storage formats have met the requirements for the data processing presented in this thesis. Simultaneously it can take too much space (more than 100 GB for a single DNS measurement uncompressed) for archival storage and not be detailed enough to debug a single anomaly occurrence. There is no perfect solution suitable for every use case.

4.1.4. Verifying data

The focus of this research is data that have not been previously acquired, processed and analyzed. Thus it is of utmost importance to be confident that a specific protocol measurement has been properly executed. There are a multitude of scenarios that can affect the

measurement. There might be anti-DDoS defenses kicking in somewhere along the network path be it the direct uplink or a transit provider, a major IXP, or a large end-user network. There might be local network issues or global scale outages, even if these do not split the Internet still the changed routing paths can become suboptimal and overloaded thus dropping UDP packets and altering the scanning and measurement results. There could materialize unexpected network anomalies. How the data quality is verified is discussed in detail in [4.9. Data quality].

4.1.5. Data processing

After the scanning and measurement stages are completed the generated measurement output JSON files (single line of the file presented in appendix A) are used for the final processing. These JSON files are the only output used in the processing stage, all the remaining logs and packet dumps are used for debugging, anomaly investigation, and data quality evaluation. The processed data set is augmented by geolocated country and AS information. For this supplementary data GeoLite2 database from MaxMind is being used, this database is free but less precise than the commercial alternatives [111].

Aggregating by the AS and country is the first and simplest processing being done. Although circumstances surrounding different IP address ranges might vary drastically in a large AS, AS still is the authority that is responsible for the network. If patterns arise regarding a specific AS then conclusions regarding poor administrative practices can be made. Not only absolute numbers but also relative to the total IP address count of the AS can be calculated. The country view can indicate which countries lack regulations, where it is not enforced, or where industry best practices are not followed.

The burst of the measurement packets is instantaneous from the network perspective of the sender (measurement system). In perfect network conditions receiver receives these packets after a short network delay proportional to the distance between sender and receiver, processes it immediately, and responds to the sender with a similar delay. In the real world, this is affected by network issues and capacity limitations across the whole path up to the end device connection, processing capacity, and other limitations. By preserving the start and end time of the received responses some conclusions can be made. Individual measurements of high power, not rate or bandwidth limited reflectors have demonstrated that they generate perfect data – there is no packet loss, there is negligible difference between the time of the first and last received packet for every response, the first response arrives after expected RTT and last response arrives insignificantly later than the first (this type of measurement is presented in appendix A). This time difference isn’t directly proportional to the RTT but rather is a sum of all the delays related to putting packets on the network on both the client and server side, processing on both client and server side which is further increased by varying delays across the network path.

Large delays (larger than RTT) between the first and last packet of a single request are potentially telling. Generally, the response is completely generated before the sending starts. If the difference is significant for most of the responses then it would indicate that the network is likely the limiting factor and the contributing attack capacity is low. Likewise, a large time difference between the first and last response might indicate the same. But in this

case, another limiting factor might be the reflector's physical inability to generate responses as fast as the requests are being received.

If the scanning stage uses a different payload than the measurement stage the first measurement response might have an initial delay caused by a different work or remote lookups being required to fulfill the request. Further responses might be cached and therefore handled much faster but it completely depends on the protocol and implementation. If the request processing is happening in parallel then most or all of the responses can be delayed. These delays are sufficiently addressed by using the time of the first and last received packet from a particular reflector and disregarding request orders and initial delays.

There is a potential for data quality improvement by extracting AS association from the global routing table in real-time, augmenting the data set with the real-time reverse DNS entries, and augmenting the data set with the connection speed and type estimates.

4.1.6. Low-level network data

Exploratory stages of this research analyzed also low-level networking data – ICMP responses. For example, the August 2018 NTP measurement data set contained 104 million ICMP responses that were advising of an inability to reach the host or other errors. The most common were UDP port 123 unreachable messages in about 50% of the cases and host unreachable in about 30% of the cases. The port unreachable message is generated by either host or inbound gateway, the host unreachable in most cases is generated by the inbound gateway [117]. 16% of the messages were time exceeded in transit informing that the sent packet's TTL went to 0, almost 3% were ICMP network unreachable, and the remaining were an assortment of the other less common ICMP messages.

These ICMP responses are extremely valuable for Internet-wide scanning research. They identify hosts that are reachable and functioning but not servicing measured protocol and hosts that are explicitly down. The technical setup and design choices used in this thesis did not permit reliable receiving and storing of the ICMP packets, therefore, no further analysis is conducted.

4.2. NTP

Precise time is extremely important for the functioning of most electronics in use today. Data logging, synchronized protocols over the network, timestamping and secure communications are some of the most common applications. Most of these devices have cheap and imprecise clocks, but some of them don't even have any persistent (battery-supported) clock and the time is lost every time power is reset. The aforementioned design choices contribute to modern consumer electronics being so cheap and widely utilized. The solution to precise time in imprecise hardware is constant synchronization over the network.

Network Time Protocol (NTP) provides the solution for time synchronization and is one of the historical network protocols that the Internet relies on, currently used version has evolved from RFC 958 proposed in 1985 which is based on even older time protocols [118]. NTP uses UDP port 123, it has a hierarchical structure having the most precise clock sources (e.g., atomic clock) available to mankind at the very top of the hierarchy and distributing it throughout the hierarchical layers to the end users [119]. Client requests can be as simple as

sending empty client mode packets, server response contains estimated errors, received, sent, and reference timestamps. NTP has been designed to take into the calculations network delays thus providing high precision time even over low-quality network links to low-end devices.

NTP is so commonplace that most of the network-enabled devices have the capability to use it already built-in. Some of those devices even come preconfigured with an NTP server and by just powering those on and connecting to the Internet precise time can be acquired without any action from the end users.

NTP is designed to operate over any network including isolated private networks but in reality, most of the clients use it over the Internet. The protocol and infrastructure are so universal that it is used in a uniform way starting from miniature low-impact IoT devices providing inconsequential services and ending with high-impact enterprise systems providing banking or cryptographic signing.

4.2.1. How protocol is abused

NTP standard responses are generally small in size providing little amplification and are not appealing to the attackers. Rossow in 2013 revisited network protocols that could be abused for DDoS attacks and identified that NTP can be abused with enormous BAF up to 4670 [35]. The researcher conducted responsible disclosure to the responsible parties but the large-scale attacks started at the end of 2013 and beginning of 2014 before remediation efforts produced a significant effect.

The functionality that caused NTP DDoS attacks is `monlist`, it is a debugging command that returns a list of recent clients with additional information like NTP version and request count [35]. It provides insight into the DDoS victims to a remote vantage point observer without access to any network logs and honeypots which is uncommon for the DDoS attack analysis. Whenever an attack is started against a victim, the victim's IP is added to the client list by the NTP server and the request counter is increased corresponding to the number of received spoofed packets. If amplification per NTP server is measured then by multiplying the request counter it is possible to establish the specific NTP server contribution to the global attacks (both bandwidth and targets) and against individual victims. To achieve that snapshot of the `monlist` has to be taken periodically and the difference processed. If all the NTP servers are snapshotted then the whole picture of the NTP DDoS landscape can be produced – all the abused servers and victims can be identified, and the overall attack capacity against all and individual victims can be calculated. Czyz et al. analyzed `monlist` responses and provided various statistical insights into the attacked victims, the targeted ports, and networks without snapshotting [36]. These results are not real-time but correspond to the overall history which varies by the server.

Additionally, the abusability of the version command was explored by Czyz et al. determining that after `monlist` remediation efforts took place distribution of NTP servers replying to the version command wasn't significantly affected [36]. The version command provides information regarding OS, software version, and stratum (level in the protocol synchronization hierarchy) as the response. This information is useful in the case detailed classification of the devices is being conducted as neither default time requests nor `monlist` provides any detailed information about the system. Version command provides much lower BAF than `monlist` but has a higher amplifier count. Information to what extent version

command is being abused for real-world DDoS attacks is not known as attack statistics and reports usually don’t go deeper than the port and protocol, only analyzing contents of the attack traffic if something previously unseen is discovered.

The client table might be small due to a specific NTP server not having clients, being recently rebooted, or not being abused for the attacks. An attacker might fuel the attack by increasing the BAF of NTP servers that provide small `monlist` responses by sending spoofed requests with different IP addresses to fill the client table with fake clients besides the existing real clients and victims.

As with some of the other abused protocols, NTP servers are useful and important for the Internet, and they have to be publicly accessible to the clients. A sew thousand public NTP servers would satisfy the whole world's time synchronization demand, in reality, the number of the NTP servers that openly and purposely serve the public today is similar to it. These servers have to be set up knowingly and added to the server pools or hardcoded into clients’ settings. Researchers have demonstrated that there are millions of publicly reachable NTP servers on the Internet [36], most of which are caused by common issues – firewall misconfiguration for legitimate use or poor default software configuration.

Significant remediation efforts have been conducted for the NTP protocol by trying to minimize the count of the publicly reachable servers and distribution of configuration that by default has an enforced rate limiting and disabled abusable functionality. The newer protocol version specified in RFC 5905 addresses various other security issues including sending *Kiss-o'-Death* responses instructing intelligent clients to follow the access settings of the server [120].

4.2.2. Special considerations

Different implementations and versions of NTP server software treat the `monlist` command differently. The vast majority were observed to completely disregard the request without any reply and that is the way how it is expected to detect abusable functionality. But multiple other types of responses stated that command is not supported. These responses are not useful for DDoS attacks as they are almost always smaller than the request itself. But as these responses are received by the `zmap` process it pipes those IP addresses to the measurement script. Measuring those servers doesn’t provide any useful data and only wastes resources of measurement and server, and increases the risk of detection potentially degrading future data. For this reason, the measurement stage has to filter these types of responses out.

NTP has a difference compared with the other measured protocol in the payload properties caused by how the different implementations handle it. This research uses the `zmap` payload of 192 bytes [121] optimized for version compatibility but it has been reported that a payload as short as 8 bytes can be used to cause `monlist` responses [35]. In general size difference is caused by zero padding the payload to a common length expected by the implementation. In this case, only the first 4 bytes of the payload `0x1700032a` have a substance the remaining payload is padded with additional 188 null bytes. There is no published data on what is the implementation support for the various padding length therefore it is not possible to estimate the average payload length that would translate into averaged BAF.

4.2.3. Scanning for abusable NTP reflectors

Although `zmap` provides both NTP discovery and `monlist` discovery payloads [121], for the scanning stage only the `monlist` command is being sent to the UDP port 123. Meaning most of the NTP servers are not going to be detected and logged therefore the ratio of abusable servers vs. total reachable servers is not calculable. The reason is that the abusable command is not part of the protocol standard and therefore NTP implementations not supporting it can't be abused. Sending many measurement requests to the reflectors not supporting it might be considered poor practice from a networking and research ethics standpoint.

The version command is not reported to be widely used for DDoS attacks and thus hasn't been implemented for NTP scanning and measurement. To obtain a more complete picture of the potential NTP contributions to the DDoS attack capacity it is planned to be implemented alongside `monlist`. It requires separate scanning and measurement but all the aspects of the implementation remain unchanged, if devices are found supporting both commands, then only the one with the larger BAF should be counted towards total capacity contribution. To minimize detection and load on the target networks and devices it is not recommended to conduct both measurements at the same time. Thus making the intersection of both data sets less precise.

4.2.4. Measuring amplification and detecting rate limiting

For the amplification and rate limit detection, the 16 byte payload is being used, it differs from the scanning payload only by the number of padded null bytes. The author has observed it being preferred by malicious scanners in the network traffic received by honeypots. There is no commonly known explicit RRL configuration, to detect the rate limit the measurement count is set to 100.

After the initial DDoS attacks abusing NTP servers at the beginning of 2014 one of the remediation solutions was to change the default software configuration to disable the `monlist` command. Detected `monlist` supporting systems indicate that they are either legacy systems or have legacy configuration, or are misconfigured.

4.2.5. Measurement data

This and the following section present a single measurement of the NTP protocol conducted on May 1, 2020. From the `zmap` scanning with the `monlist` payload, the measurement stage received 656,923 responses. Of these 488,273 had the undesired small-size payloads which were excluded from the measurement in which 168,650 reflectors were actually measured. These unmeasured responses were 8 byte various error and version mismatch payloads from 304,445 reflectors and 48 byte standard properly formatted NTP v2 payloads from 183,770 reflectors containing clock synchronization data, many of which seemed to be synchronized. These reflectors likely are running the same implementations or versions of the software that is not handling unknown requests properly.

There was a small subset of the reflectors that respond with non-empty but small packets. As with the 8 and 48 bytes responses most of these seem to be implementations having custom error payloads or unexpected behaviors but their prevalence is low and they

were not filtered out from the measurement stage to determine if amplification is provided for further requests. Through multiple measurements, this was confirmed not to be the case and all further NTP `monlist` measurements use 80 byte response payload (`monlist` with 1 result) threshold to decide if an individual reflector must be measured which significantly reduces the number of unnecessary burst requests and consequently potential blacklisting.

The most important devices are the ones responding properly to the monlist command and providing a client list. Therefore for the attack capacity analysis data set is filtered to have only reflectors that have responded at least to 1 of the 100 requests with at least 80 byte payload (`monlist` with 1 result), there were 22,222 measurements matching this requirement. Other measurement results are not supporting `monlist` and therefore didn't generate any significant amplification even as a bug or error message that could be abused. Sent `monlist` requests include 16 bytes of the payload, meaning the BAF cut-off, in this case, is 80/16=5.

Some of the smaller responses which are fast and not rate limited could be potentially abused, it makes no sense for an attacker to abuse reflectors with BAF below 2 even when no rate limit is present because packet loss and other issues will decrease real attack traffic reaching the victim. The BAF 2-5 is low and is rarely exploited, attackers prefer much larger BAF and there are no known published attack cases that use NTP with so low BAF. The most common response within the range of 32-79 bytes is a 48 byte standard synchronization payload containing time data, the remaining responses in this range are insignificant. There were 183,770 synchronization responses but none of those were measured to detect RRL. The 48/16=3 BAF is borderline useful but it might be possible to trigger the same responses with shorter requests thus increasing the BAF. The set of these servers is noteworthy but it can't be used to create record-breaking attacks or anything even remotely close to what continuously abused protocols produce. For the NTP protocol, BAF up to 10 can be considered low because it is not known to attackers if the identified abusable reflectors actually support the smallest amplification payload of 8 bytes.

4.2.6. Attack capacity

The proposed measurement methodology entails RRL detection and implementation purposefully maintains the association between sent requests and received responses. Figure 17 presents the count of received responses for every one of the 100 sent requests while preserving the sequence. The trend is clearly downwards pointing. If any implicit or explicit rate limiting is happening it might be affecting the total number of responses without a clear pattern. Data didn't allow the author to observe common patterns of rate limiting when comparing individual measurements or even sets of those. The cause might be that it is not known in which order packets are received by the reflectors.

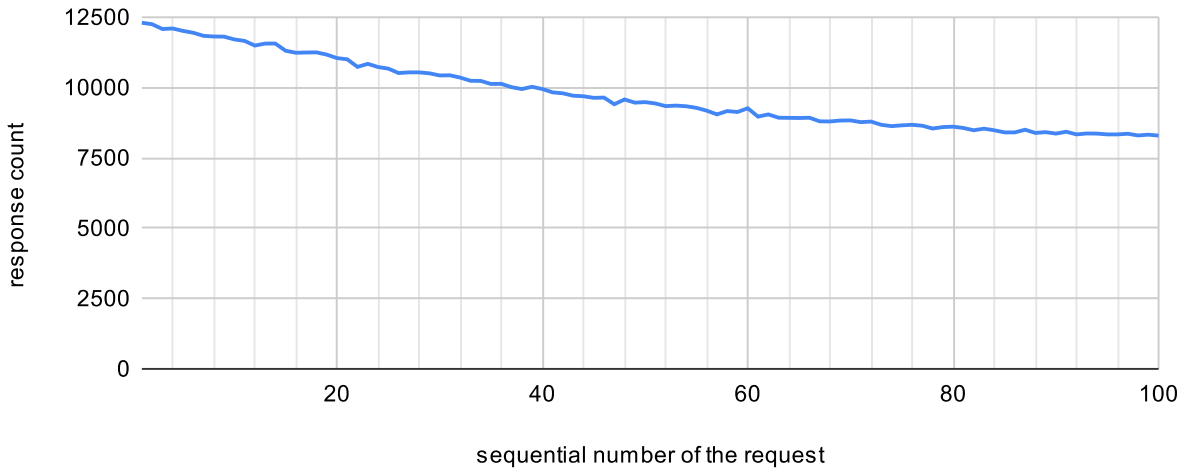

Figure 17: Count of responses for every request number

Aggregated number of NTP reflectors per count of responses presented in figure 18 portrays a much more compelling picture. Noteworthy are the clearly observable spikes around 10, 20, 30, and 40 responses. There is nothing in the measurement implementation or networking setup that relies on increments of 10 for sending or receiving packets. It indicates that some kind of rate limiting is present. It is not necessarily explicitly defined in the configuration file of the NTP server software. It might be a hardcoded limit inside the software or the system itself, especially for low-power embedded systems. This limit might be also present outside of the reflectors, it is possible that some rate limiting might be enforced by intermediary network devices in general or possibly targeting response payloads known to be used primarily for DDoS attacks. It is not enforced by measurement network ISP otherwise the full response spike would not be so significant. It is unlikely that this limit is enforced by a major IP transit provider, and it is unlikely that end-user networks apply these limits manually. Another possibility is that some network security solutions apply these limits automatically. Midsize ISPs are most likely candidates that would create this type of limiting policy manually. It is a compelling question for future research but is not further investigated in this thesis.

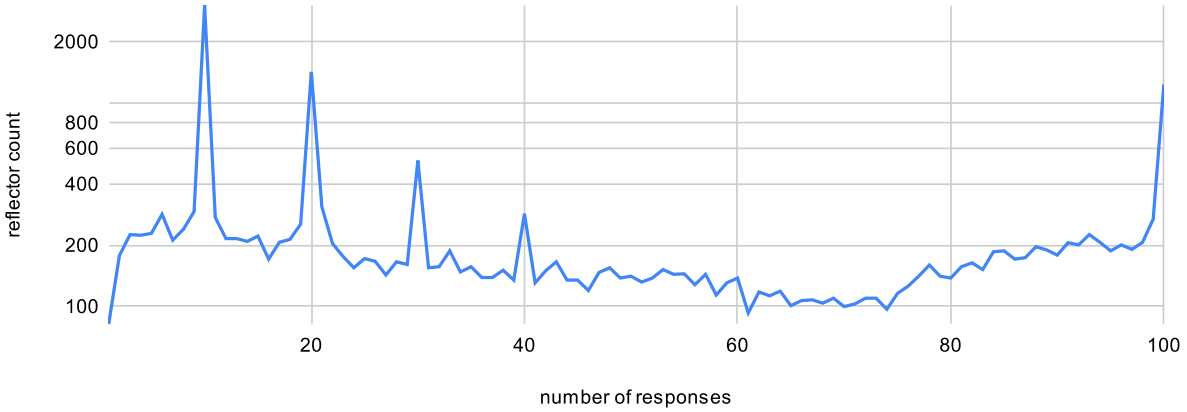

Figure 18: Count of NTP reflectors per number of received responses

NTP server distribution per average response size is provided in figure 19. Average is calculated over received responses, if a single response is received then its size will be the

average. Most common values are displayed individually and uncommon are aggregated together, the highest values are the most significant ones. With 80 byte payload which is the most common response size responded 4742 reflectors, all of those are `monlist` replies containing a single client entry. There were only 545 reflectors that provided the maximum possible response of 100 packets with 440 byte payload totaling 44,000 byte responses without packet loss.

A single client entry uses 72 bytes of the `monlist` reply. Responses containing 2 clients are 152 bytes long and were measured for 3639 reflectors, 3 clients are 224 bytes for 2588 reflectors, 4 clients are 296 bytes for 2503 reflectors, 5 clients are 368 bytes for 1733 reflectors, 6 clients are 440 bytes for 756 reflectors. Single response packets contain no more than 6 clients, if there are more than that for every 6 clients a full packet is sent with an additional partial response packet containing the remaining clients. About 27% of the reflectors responded with more than 1 response packet per request, from this data it is impossible to establish what percentage of the remaining reflectors are capable of producing more than 1 packet, for that artificial filling of the client table would be required. Early testing data (not reviewed here) indicated that one of the remediation approaches was to limit NTP software to a single packet response.

A small response is not necessarily limiting the total contribution to the attack but it is definitely increasing network resource requirement from the attacker. If no implicit or explicit rate limiting is present then the reflector can utilize all the upload bandwidth available to it.

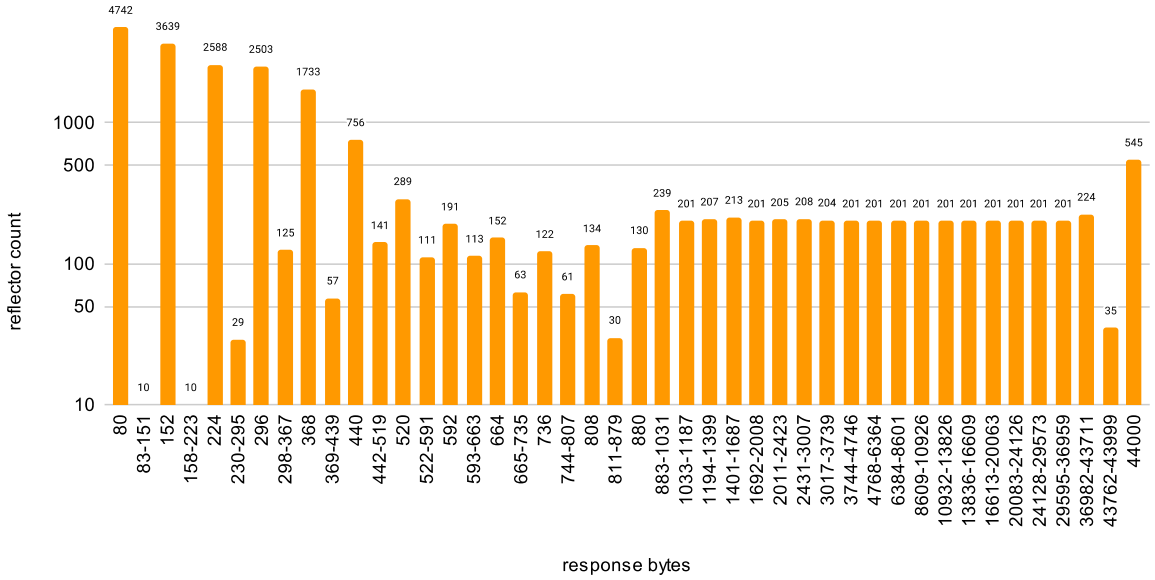

Figure 19: NTP average response payload size distribution

The geographic distribution of NTP reflectors is presented in figure 20. Most NTP reflectors that replied to the `monlist` command were located in China – 5424, USA – 3171, Brazil – 3090, Vietnam – 1387, Spain 1294, Turkey – 1092, Indonesia – 1021, Malaysia – 691 and Taiwan – 675. The USA has been commonly disproportionately represented in many scanning research, which might be surprising but it is related to the historic availability of the Internet and a high number of legacy systems, in the early testing measurements the USA had a significant lead over China in the number of NTP reflectors. The whole continent of Africa has very few amplifiers, about half of the countries have none. With the limited bandwidth

and high costs of the Internet in Africa, it is expected that the contribution to the total attack capacity is insignificant. Asia is a high contributor to this and many other network issues which are caused by the fast proliferation and growth of the Internet in these developing countries. Large connection count and fast speed coupled with a lack of regulation and enforcement, and general disregard for the best network management practices cause countries of Asia to be breeding grounds for cyber security issues. But as highlighted by existing research [4] pure reflector count is not a good metric when estimating the contribution to the total attack capacity, the count has to be balanced against the upload bandwidth.

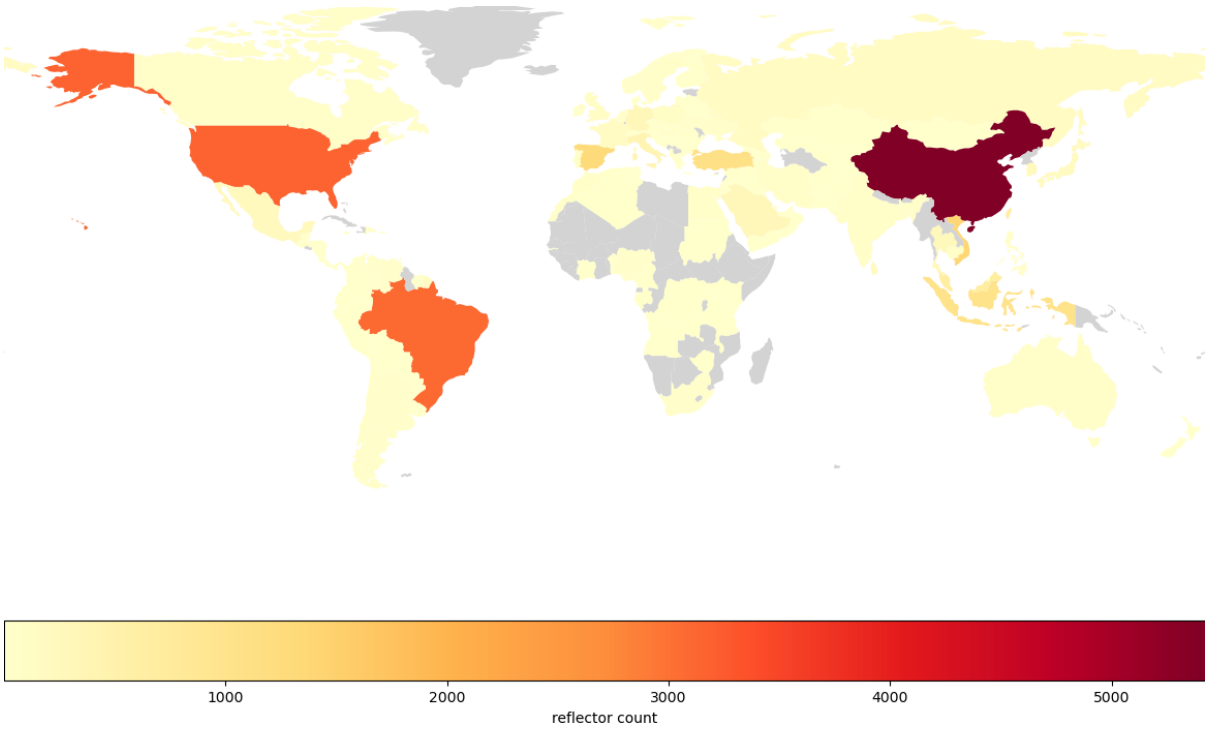

Figure 20: NTP reflector geographic distribution

There might be multiple causes for the large device count in some networks not just poor network management. ISPs can supply end client network devices possibly having some network services enabled by default. Some vendors supply specific regions with preconfigured devices having abusable services enabled by default and bundling network connection often wireless with the delivered solution. It might be unfeasible to disable these existing services remotely.

Logged timestamps enable the exploration of additional aspects. The first thing that can be extracted is the first or more precisely shortest time before the response is received, it corresponds to RTT. This is the way how ping functionality works but some additional time is added because the NTP response has to be generated. The minimum time can be loosely associated with the distance between the measurement network to the measured reflector. Reflector distribution by the time of the first received packet is presented in figure 21, excluded from this visualization are 393 reflectors that had RTT between 1 and 2 seconds, and 143 reflectors with RTT above 2 seconds (up to the 10 second measuring interval).

Reflectors with RTT below 200 ms are located in Europe or nearby countries as the measurement system is located in Europe. Almost 54% of the reflectors responded within the 200-400 ms range which corresponds to a good or average RTT to other continents. RTT

above 400 ms in many cases can be considered as poor network performance caused by suboptimal network routing. But the values above 1 second are not acceptable as normal network performance.

For these values above 400 ms two main causes besides poor network routing can be identified. First, poor network connection often is caused by mobile or other wireless connection technology where technology affected by the low signal quality creates a slow connection. Second, an endpoint can be overloaded either on a network level (link can be saturated) or the device can be overloaded or has extremely low processing capacity available. Some of these devices even if measured to be contributors might contribute very little to the overall attack capacity. To assess that packet loss and the time difference between the first and last received packet has to be calculated.

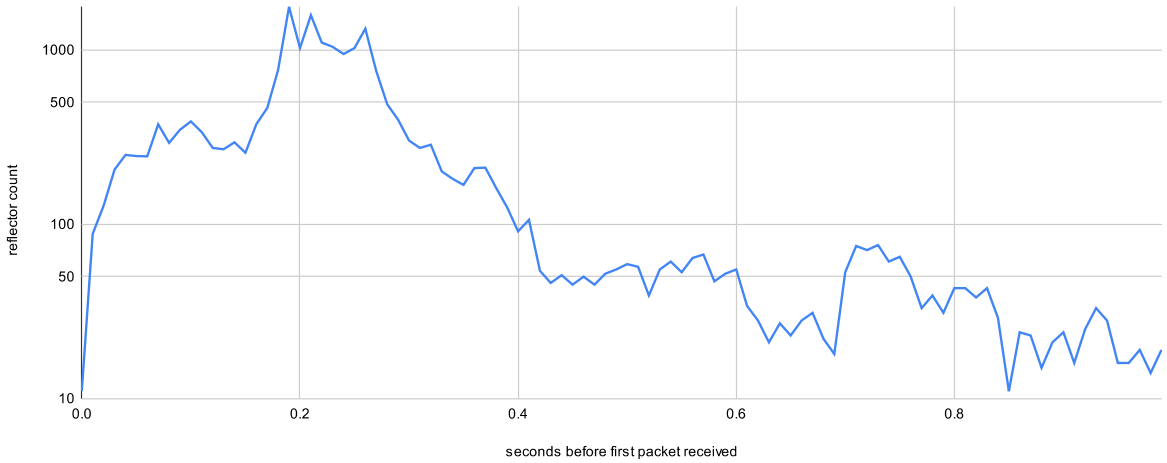

Figure 21: NTP reflector distribution by time before the first received packet

Can individual reflector speed (bandwidth contribution) be estimated from the timestamps as proposed in the measurement methodology is one of the research questions of this thesis. If for start and end time the minimum and maximum values are selected then a single delayed packet skews the calculation significantly. Speed is calculated by adding up all the received payload sizes and 42 bytes as transmission overhead for each received packet and dividing it by the time difference between the first and last received packet, responses with one packet are ignored (as are incalculable). But even that is not enough as indicated in figure 18 rate limiting is present therefore the capacity calculation should always take that into account to be reliable. As UDP is an unreliable protocol the response rate 100% (RR 100%) is not guaranteed even if the measurement system and reflector processes 100% of the requests. Therefore a reasonable threshold has to be selected to accommodate some packet loss not associated with RRL, the author for all the protocols measured in this research selected the minimum response rate (RR) of 80%.

Summing all the calculated average speeds of the 5028 reflectors having RR 80% produces **43 Gbps** NTP attack capacity. This speed is rather an estimate, there might have been competition for the bandwidth with ongoing real DDoS attacks, average speed decrease due to the distance, and intermittent or permanent network issues. Calculated attack capacity for the different RR and respective reflector counts are presented in figure 22. Selecting the minimum calculable 2 response RR would produce an unrealistically large 291 Gbps attack capacity ignoring all RRL. While selecting RR 100% would produce only 15 Gbps attack

capacity excluding all reflectors that lost even a single response. RR 90% with 31 Gbps might be even more realistic but to maintain comparability with the other measured protocols that have lower data resolution (less measurements) the RR 80% is maintained across this thesis.

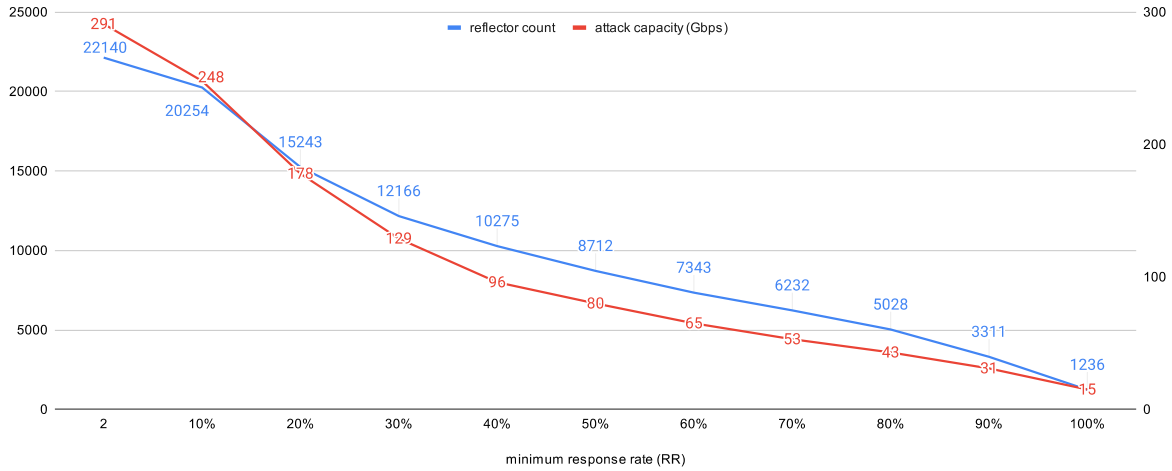

Figure 22: NTP reflector count and attack capacity for different minimum response rates

NTP reflector speed distribution (RR 80%) is presented in figure 23, excluded from this visualization are 17 outliers having speeds between 147 and 503 Mbps. There are only 29 reflectors having measured speeds above 100 Mbps, most of which are identified as data center networks rather than residential Internet connections. These are the highest individual contributors to the attack capacity if no RRL (higher than the measured) is present. 76% of the reflectors responded with a speed below 10 Mbps which likely indicates a combination of slow network connection and low-power devices. 97% of the reflectors had measured speeds below 50 Mbps.

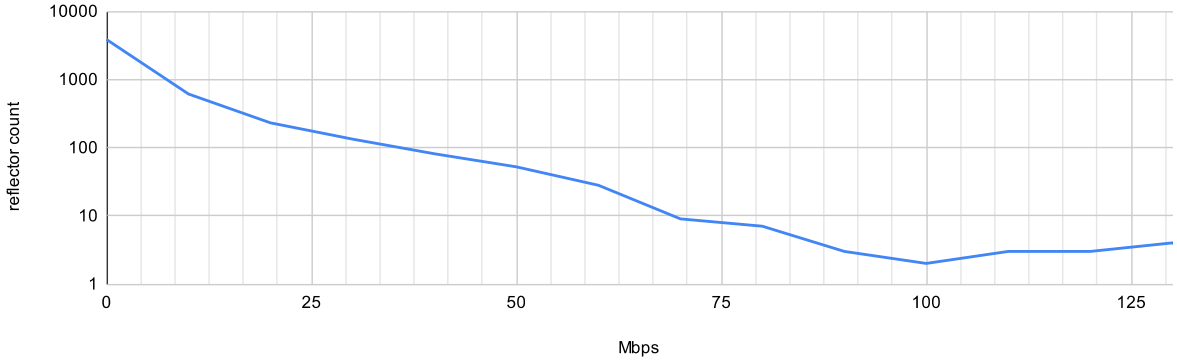

Figure 23: NTP reflector speed distribution

Reflector (RR 80%) distribution having a speed below 0.5 Mbps is presented in figure 24. These reflectors either have a low response rate and respond slowly, or have a high response rate and take multiple seconds to respond from the first to the last packet. Random sampling indicates that a significant portion of these devices are utilizing slow-speed wireless connections to the Internet. In total 578 reflectors responded with a speed below 0.5 Mbps, while these individually are insignificant contributors to the overall DDoS attack capacity separate BAF calculation targeting only this set has to be completed before determining whether to exclude those from the overall attack capacity calculation.

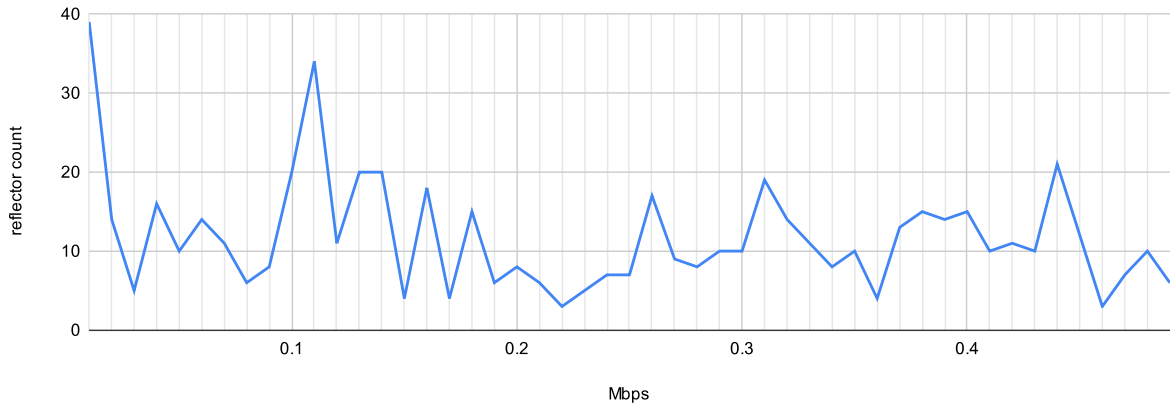

Figure 24: NTP reflectors responding with the average speed below 0.5 Mbps

Real measured BAF for the 43 Gbps (RR 80%) capacity can be calculated by dividing total received bytes with 100 sent payloads multiplied by payload length and reflector count. In this case, the real total measured **BAF** was `596563454/(100*16*5028)`=**74.2** which is impressive but significantly below the standard maximum of 2750. If an optimized 8 byte payload would produce the same measurement data then doubled **BAF** of `596563454/(100*8*5028)`=**148.3** would be achieved. If an attacker would disregard reflectors having large packet loss and small response payloads then significantly larger BAF could be achieved but only with the bandwidth far below the calculated total attack capacity.

Bandwidth contribution is a much more important metric than the reflector count. The geographic distribution of the attack capacity (RR 80%) is presented in figure 25 where all countries that were measured providing less than 1 Gbps of attack capacity are excluded as insignificant contributors. Compared to the count (figure 20) significant differences can be observed. Only 6 countries exceeded 1 Gbps contribution. China was the largest contributor with 20.3 Gbps or 47% of the calculated capacity. The question becomes if this is a country (or region) issue or if there are individual responsible networks?

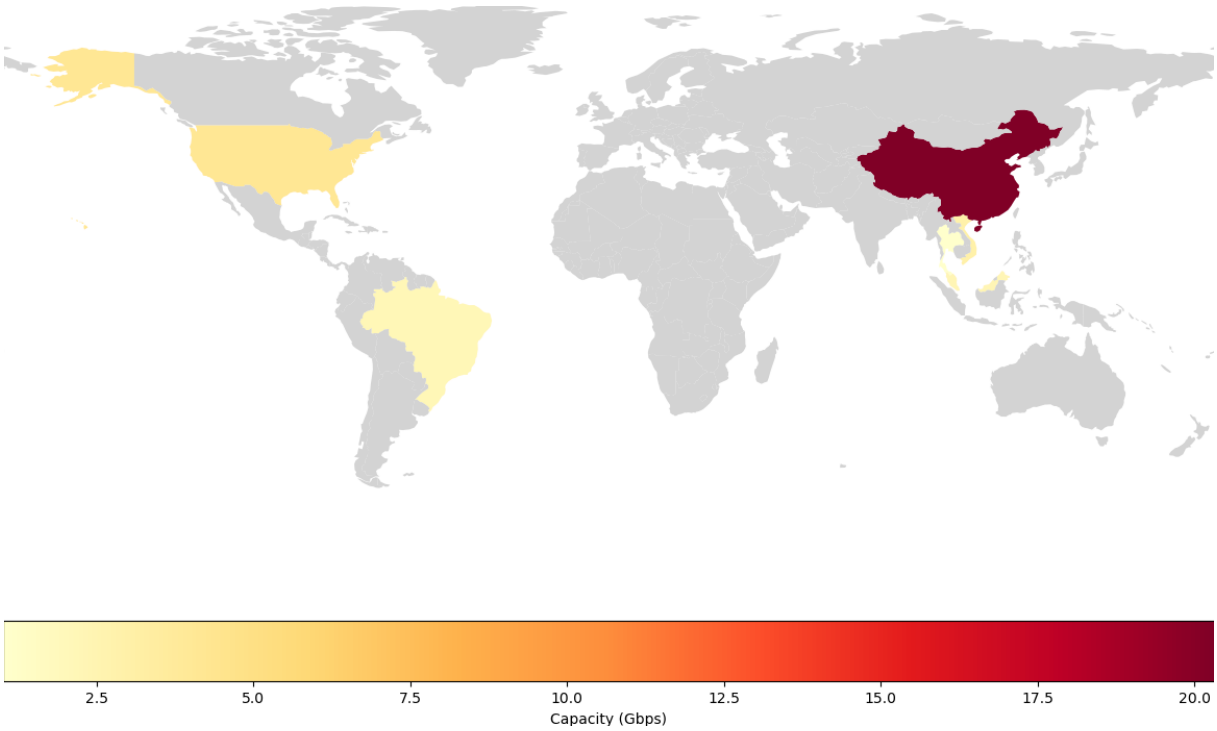

Figure 25: Measured capacity geographic distribution of the NTP protocol

The top 10 networks contributing most to the attack capacity are presented in table 2. Chinanet alone contributes 38% of the total attack capacity but because of how the Internet in China is built it is impossible to discern a specific use case of the network in this case. In the case of the remaining providers, it is not clearly discernible from the measurement data if the measured servers are located in a data center, or are connected using a business or residential Internet connection except for the hosting provider DREAMHOST. These 10 networks contributed 63% to the overall attack capacity which signals that these contributing devices might be ISP-deployed end-user devices having NTP service enabled by default on the Internet-facing network interface. This measurement data view can provide insight into the actual bandwidth that ISPs provide to attacks which is a waste with the real-world associated bandwidth cost, which might motivate network maintainers to address the reflector presence.

ASN	AS name	Country	Gbps
4134	Chinanet	CN	16.4
58466	CHINANET Guangdong province network	CN	2.2
4788	TM Net, Internet Service Provider	MY	1.8
7552	Viettel Group	VN	1.7
26347	DREAMHOST-AS	US	1.2
4837	CHINA UNICOM China169 Backbone	CN	1
4230	CLARO S.A.	BR	0.9
7713	PT Telekomunikasi Indonesia	ID	0.7
45899	VNPT Corp	VN	0.7
9534	Binariang Berhad	MY	0.6

Table 2: Top AS measured attack capacity contribution for the NTP

Overall NTP can be considered a highly remediated protocol. Only a small subset of the globally reachable NTP servers support the `monlist` command and furthermore, there seem to be present aggressive RRL configurations that considerably limit global attack capacity. Only a small number of networks contribute most of the capacity which simultaneously

indicates that this is not a country-level issue but rather individual networks and devices deployed by the networks themselves might be the culprit.

4.3. DNS

Domain Name System (DNS) is one of the cornerstones of the Internet, it enables Internet users to rely on easy to remember words as addresses instead of typing the long hard to remember digit strings. While new protocols that get discovered to be abusable for the reflected DDoS attacks often get significantly remediated within months and sometimes even become irrelevant in the overall DDoS landscape, the DNS case is the complete opposite, it has been abused for the DDoS attacks since the early 2000s and the remediation efforts haven't been successful.

DNS servers have two modes of operation – authoritative and resolver, UDP protocol on port 53 is being used [122]. Authoritative servers serve specific domain zone and are administered by the zone owner. Resolvers are servers usually close to end users that communicate with the authoritative servers recursively to fulfill user queries. Recursivity is required because only the root DNS servers are known to the resolvers by default. Root servers point to the server of top-level domains, e.g., *.com*, then the appropriate top-level server points to the DNS server that manages the required domain which in turn can either respond or redirect further to a subdomain DNS server. This approach is too complex and too slow to be executed on every user system, the selected approach was to create separate DNS servers in the user network so the responses get cached and become faster for all the network users, and fewer requests are being sent to the authoritative servers.

4.3.1. How protocol is abused

A resolver usually serves clients of some specific network, this service should be only available to the clients connected to that network. There are very few exceptions when the resolver should be available publicly to anyone by design – stability, censorship circumvention, alternative zones, and malicious domain blocking. Many of these reasons are based on the DNS evolving from a purely technical task into a governmental and other third-party desire to control users [123]. The author estimates that a few thousand publicly accessible resolvers would suffice for the whole Internet. In reality, there were about 10 million resolvers publicly available in January of 2017 which is still a significant improvement over about 22 million in March of 2013 [71].

There are multiple causes for the large number of open DNS resolvers on the Internet. One is the legitimate resolvers serving specific networks which by network administrator error are accessible from the Internet. In 2018 there were close to 400 thousand assigned AS numbers [124]. Even if assumed that every AS has this kind of mistake it still doesn't come close to the total number of resolvers.

A more common reason is CPE and other reachable devices at the end client locations. These devices are often mass-produced cheaply and neglect security aspects, even if addressed later on as a firmware update these devices are rarely updated in actuality, making them a security risk till their end of life. The majority of these devices provide useful network services like NAT, DHCP, and DNS. Even if the DNS servers provided by ISP are fast,

caching results on these devices make it even faster but the issue is that vendors and ISPs configure DNS server settings improperly thus instead of providing service only to clients directly connected to the LAN, it is provided to all network interfaces including the Internet. The third category is software-based, historically hosting your own domain zone was common practice but today with development of the CDN, DNS hosting, and the rise of cyber threats, it makes little sense to host your own authoritative server. OS distributions, software packages, and common configuration practices tended to include DNS servers for this reason, when left unconfigured these servers often acted as resolvers thus completely unnecessary and unused services consumed bandwidth and resources of the system and contributed to the DDoS problem. With better security practices DNS server software is not commonly installed or enabled by default anymore and their numbers dwindle as the old servers are decommissioned. But poor configuration practices by inexperienced system administrators still cause new resolvers which are completely unnecessary and not used for any legitimate purposes.

Authoritative servers can also be abused for DDoS attacks if they are not rate limited although no major DDoS attack using this approach has been reported. Attackers would need to crawl the web or otherwise acquire a large list of domains and then recursively determine appropriate authoritative servers. Then each server has to be tried for common queries to pick the one that generates the largest amplified response for a particular domain and rate limiting has to be measured, abusing open resolvers requires none of these steps and still most likely provides a larger attack capacity. The actual real-world capacity for this type of DDoS is an open research question that lies outside of the scope of this thesis, Anagnostopoulos et al. have explored this potential for abuse [37].

A noteworthy fact about the DNS system is that it is not only abused for DDoS attacks but also is a common target for the attacks, by disrupting the DNS server of a specific domain attacker can disable its resolving for the users. By disrupting root DNS servers attackers could bring down the whole Internet DNS, these servers have been first attacked in 2002 [1]. There have been occasional major attempts since the first attack and one of the largest registered was at the end of 2015, it employed valid DNS queries from the spoofed IP addresses and therefore had to be processed regardless of the RRL, for every one of the 13 root server groups increasing request count by 5 million per second knocking out a few of root server groups but because of the resilient root server infrastructure attack managed to only slow down some of the clients' requests [125].

Resolvers are optimized for speed by being located close to the end users and reusing locally cached responses for all the clients often without any limitations. Besides blocking access from the Internet another proposed practice is RRL per client IP [91]. It should be done mandatory for the resolvers that are purposefully available on the Internet. It would be beneficial if the RRL was configured by default in DNS software distributions and disabling RRL would require manipulations from the system administrator. In this scenario, even misconfigured devices would slowly propagate with RRL enabled eventually replacing old publicly reachable servers having no rate limiting. It would not solve the issue completely but would have a significant positive effect on overall remediation efforts.

4.3.2. Special considerations

DNS protocol has a special property rarely implemented in network protocols and not implemented in any other abusable protocol measured in this research – utilizing both UDP and TCP protocols for the same functionality, in this case, the selection is made based on the size of the response. RFC 1035 specifies that "Messages carried by UDP are restricted to 512 bytes (not counting the IP or UDP headers). Longer messages are truncated and the TC bit is set in the header." [122]. Meaning that all DNS servers that have implemented protocol specification correctly should never send responses larger than 512 bytes thus amplification available to attackers is severely limited. When truncation happens it is the responsibility of the client to initiate a TCP connection to the server to receive the full response, which will not occur in a DDoS attack case. The motivation behind this design choice is to balance speed and reliability, if the DNS response fits within a single IP packet then the speed is chosen otherwise reliability is preferred.

Extension mechanisms for DNS (EDNS0) described in RFC 6891 permit DNS packets to be significantly larger in size (maximum 65535 bytes) before truncation, a common value is about 4096 bytes [126]. The client has to set a specific flag in the request that it supports the large responses and then if a DNS server supports this feature it can send the enlarged response over UDP. As an attacker is spoofing the requests, it can set this flag resulting in victims receiving larger responses thus significantly increasing BAF. Rijswijk-Deij et al. confirmed that queries generating large responses, primarily DNSSEC, can be abused for the DDoS attack by also using 4096 byte response size [127]. Rossow observed that open resolvers supporting these larger responses proportionally are far less common than authoritative servers [35] but detailed insight into causes for it is still lacking. As is lacking the real-world DDoS data presenting to what extent this type of query is abused. It is possible that these open resolvers are embedded or network devices running outdated software but without classification, it is not possible to verify that.

Support of both TCP and UDP protocols potentially simplifies the solution to DNS abuse by moving to TCP as the default transport mechanism. RFC 7766 provides implementation guidelines on how to achieve DNS over TCP performance similar to the currently used DNS over UDP [128]. This migration would not only remediate DNS contribution to the DDoS attacks over time but also more easily implement currently lacking security features like encryption.

4.3.3. Scanning for the DNS open resolvers

Hendriks et al. noted that default payloads for DNS scanning cause a high rate of fake resolvers possibly under the control of China's government-related entity to be detected [78]. The purpose of this kind of fake resolvers is an open question. For this reason for scanning payload custom domain under the control of the author is utilized and A entry is queried. Only if the response is correct the specific DNS server is functioning properly as a resolver and is used for further measurements. If the received answer is wrong or contains an error the DNS server might be authoritative, misconfigured, censored, or have transient errors. For the purpose of this thesis, it is assumed that a wrong answer means also that significant amplification will not be possible in responses to attackers' queries, so it is not efficient for an

attacker to abuse it and if abused then these reflectors will not be highly contributing factor to the overall attack capacity. Future research might be mandated for the root cause of different behavior of resolvers, actual contribution to the attack capacity, and real-life use in attacks.

For the scanning purposes standard query without EDNS0 is being used. Researchers have noted that many resolvers do not support EDNS0 [127]. All servers should support standard query, it queries A record which even poor implementations should handle without any issues.

4.3.4. Measuring amplification and detecting rate limiting

During the testing stage, it was identified that DNS resolvers implement truncating in two different ways. Either sending as many responses as possible that is equal to or under the allowed DNS response size or always sending DNS reply without a single answer with just the truncated flag set. This difference has caused the need to adapt the measurement approach by introducing another internal stage of precheck, it is not possible to extract the required information from the scanning responses. That is why before the actual measurement a single EDNS0 packet is being sent and the response analyzed, if the response larger than 512 bytes (even if truncated) arrives within 5 seconds then EDNS0 measurement requests are sent, otherwise standard requests are being sent even if the precheck response was not received. That way set of open resolvers can be split into the ones supporting EDNS0 and not supporting it which can be used for further investigation.

For the amplification measuring stage either standard or EDNS0 queries are being sent. The maximum size of the packet is set to 4096, the consideration being that this value is lower than the maximum value and is the most common EDNS0 size value [127]. For this step, the primary objective is to produce requests that generate responses larger than 512 bytes if possible. Two separate TXT entries are being used, one generating 500 byte standard response and another one generating about 4000 byte EDNS0 response, the maximum value is not used as the response size can vary by other factors. For EDNS0 entry multiple entries are being used to generate truncated responses containing partial data if the resolver implements that.

For the initial testing, the used authoritative server peaked at around 7300 RPS generated by the resolvers. Initial attempts at the full Internet measurement intermittently stopped producing measurement results while scanning continued unaffected thus producing useless results. Detailed investigation revealed that the uplink provider of the utilized ISP had automatic DDoS attack detection and filtering triggered by the measurement TXT queries but not by the scanning A queries. These occurrences were the motivation behind the decisions made in the [4.1.1. Technical setup] and [4.9. Data quality].

The maximum response size for DNS is known to be either standard 512 bytes or larger if servers support EDNS0. Measured amplification is not the maximum possible, as it requires a special as short as possible domain not used in this thesis. Maximum amplification in the case of 512 byte packet limit is about 22.3 [127]. In these cases measured amplification can be disregarded and 22.3 BAF can be used as the fixed value for the calculations instead.

RRL as a proposed solution to remediate the issue has been implemented in most of the popular DNS server software. It often is implemented differently from the other measured protocols and usually is calculated for the same query per client IP address because the DNS has a critical role in the functioning of the network and effects of the individual malicious

users have to be minimized. Other network services commonly calculate RRL as per all requests from a client. The main issue is that RRL is not enabled by default as default settings might affect out of the box functionality. Default limit settings and calculations vary as well thus decreasing the effectiveness of the measurement creating the need to send more packets to cover a wider range of default limits.

Microsoft Windows DNS server default RRL is quite low at 5 RPS [129]. BIND DNS server software has different configuration rate limiting options for authoritative and resolver modes, rate limit although recommended for authoritative servers can be used also resolvers. BIND offers multiple methods how to mitigate the negative effects on users from the same network or IP addresses by creating small truncated responses and responding occasionally thus forcing legitimate clients to use TCP connections which are affected by the rate limiting differently [130]. These patterns can be observed in the measured raw data because of the chosen implementation.

ISC developing BIND originally suggested RRL at 5 RPS [131] but it doesn't take effect without explicitly stating it in the configuration file and default software distribution is shipped without this value provided and further package distributions usually don't add it. Meaning administrators configuring the RRL have to acquire RRL from some source of information but various books and guides suggest inconsistently different values. Different DNS server software have different limits therefore for the DNS measurement single dominant RRL can't be established. For the purposes of this research, 50 requests are being sent to measure responses. All the devices should be able to handle that load as publicly reachable DNS servers are already abused for DDoS attacks.

4.3.5. Measurement data

DNS is a more frequently used service compared to the NTP which results in a much larger data set. This and the following section present a single measurement conducted on May 18, 2020. The first thing that is checked is the payloads of the 8,415,951 responses received by `zmap` in the scanning stage. There were 4,378,444 reflectors that responded without providing the expected IP address for the A request, these responses were ignored and not measured any further. About 74% of these responses were REFUSED messages. The majority of the remaining were different DNS error codes like domain not found, not authenticated, and server error. A small number of servers fulfilled the response but provided the wrong IP address, these are filtering, censorship, and misconfigured servers. A large number of these are authoritative servers and are not measured in the scope of this research.

4,037,249 reflectors responded with the correct IP address for the A query meaning they are open DNS resolvers, for all of these precheck was executed and the full measurement was conducted. Around 13.5% of the reflectors didn't respond to the precheck EDNS0 request causing the individual measurements to fall back to standard queries, this behavior justifies the optional precheck stage in the measurement methodology.

Although while successfully fulfilling scan requests some resolvers were unable to fulfill TXT requests and responded with small or empty packets usually containing no responses but having error payloads (commonly 27 bytes long). As the measurement request is 29 bytes for standard request then the produced BAF is below 1 which is unusable from the attackers' point of view. For the attack capacity calculation produced measurement data set

has to be filtered to exclude low BAF reflectors, based on the response size distribution and pragmatic attacker preferences the author has selected this cutoff to be 135 byte response payload (at least one per measurement per reflector) or BAF of around 4.7. Therefore the attack capacity in the following section is calculated based on the measurement data set containing 3,618,577 open resolvers.

4.3.6. Attack capacity

DNS reflector average response payload size distribution is presented in figure 26. The most common payload length of 493 bytes corresponds to the expected standard query response which was received from 2.9 million or 72% of the reflectors. This standard response length can vary based on the DNS resolver software implementation adding or removing some protocol properties into the response payload besides the TXT value. A significant number of lower averages are caused by the servers' occasional inability to respond to the request properly, instead returning errors or empty packets. Individual responses significantly smaller than the standard are invalid and can't deliver the full predefined TXT response.

All responses between 512 bytes and around 4000 bytes are expected to be EDNS0 which are often truncated to common values. The second most common response payload length was 3627 bytes received from 5.4% of the measured reflectors, a significant number of reflectors had similar lengths. MTU value around 1500 causes the response packet to be cut short by fragmentation resulting in invalid responses for which other fragments to properly reassemble the packets were not received. About 1% of the responses contained 1464 or 1472 byte payload. This is a potentially interesting research question – where and to which size the packet length is cut, as DNS resolvers are very common then most of the world's AS can be covered.

There is a significant number of reflectors having payloads above the expected EDNS0 response size of around 4000 bytes. 2.4% (97,274) of the reflectors had a response payload size above 4206 bytes. These are misbehaving reflectors most commonly resending responses multiple times for a single request. Although these do not behave the same way as the routing loops discussed in [4.9.4. Anomalies], these still might be routing loops handling packet TTL properly or erroneous software implementations. The most common response values having 24,650 byte length (83.8% of the large response reflectors) were simply separate standard 493 byte responses looped 50 times. As 50 is also the measurement request count and most of these measurements had only one response (instead of 2-50) consisting of 50 payloads it is believable that a poor implementation improperly caches the response port number (which should never be done) causing the generated responses to be sent to a single port on the measurement system instead of 50 different ones. An embedded device software that skipped testing consecutive DNS requests (repeated from the different client ports) could have made it into production. This behavior was not widely observed for the other protocols measured in this thesis further decreasing the likelihood of it being a routing loop. As this is purely the author's unconfirmed speculation these cases are processed normally as the ones having RR 1 and therefore excluded from any capacity calculations. Device classification is required to confirm that this is either a single maker, model, firmware, or software, potentially requiring special handling of these cases for the capacity calculations for the DNS measurements only

which contradicts the proposed measurement and calculation uniformity for all the protocols. The cause and the calculation solution of this issue remain an open research question.

Although it is not explored which responses are anomalous and which are proper it doesn't matter in most cases. Attackers care about BAF and the capacity which the measurement methodology produces without analyzing the payloads. DNS response data is easily decodable as long as it is not cut short by fragmentation or generated by an erroneous implementation, this might be an interesting research question to extract values of the EDNS0 options for analysis.

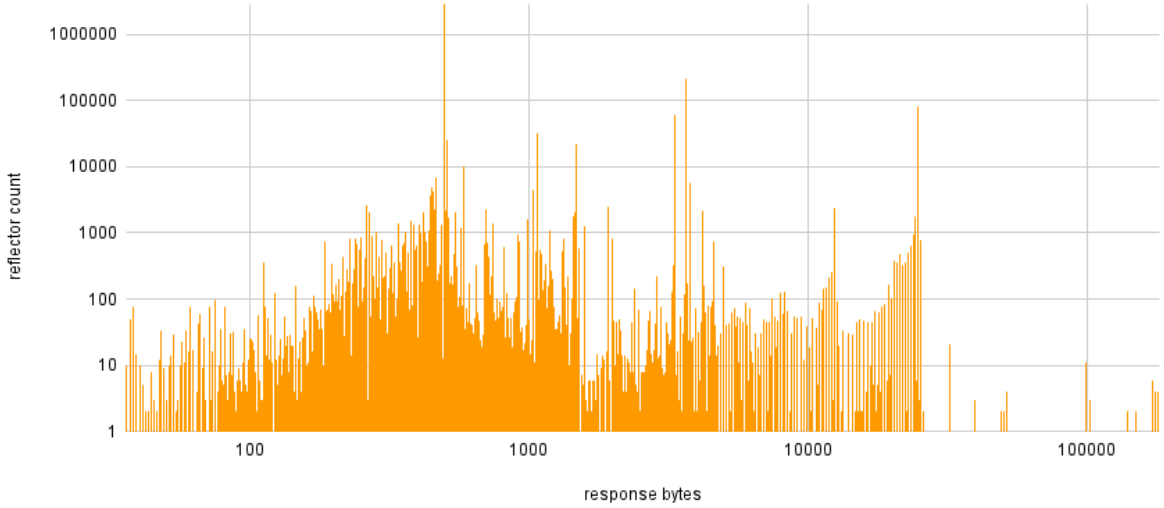

Figure 26: DNS average response payload size distribution

Figure 27 presents the count of received responses for every one of the 50 sent requests while preserving the sequence. Although there is a clear downward trend reconfirming this data presentation's limited usefulness for all the measured protocols, DNS is the only protocol that presents a noteworthy drop. It is present right after the first response which could indicate an extremely aggressive rate limiting.

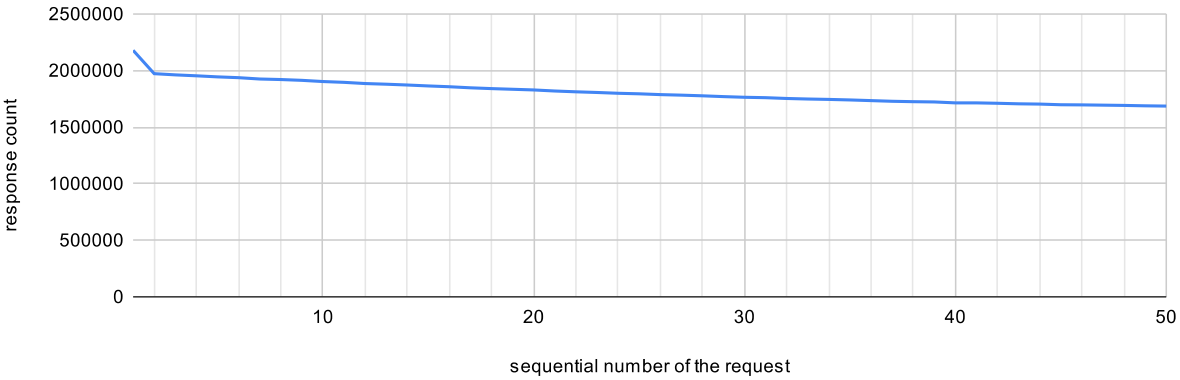

Figure 27: Count of DNS responses for every request number

Aggregated number of DNS reflectors per count of responses presented in figure 28 confirms that there is indeed a rate limiting present after a single response for the 419,474 resolvers. Most commonly reflectors responded to all of the requests indicating no measurable RRL (>50) and secondly only to a single request. As there is a single scanning request plus for the DNS extra precheck request, then these 2 additional requests can be increasing identified

RRLs depending on the RRL implementation and other factors outside the author's control. Therefore the observed aggressive RRL could be a singular value 1, 2, or 3 or even spread across these. The second highest rate peak was observed around 9-11 from 635,777 reflectors with the extra requests would be RRL 9-13, because of the spread across multiple values likely multiple different RRL values and implementations are present around the round number "10" developers and system administrators might prefer. There are smaller spikes that would indicate a human-caused rate limiting around 20, 25, 30, and 45 requests. Measurement data doesn't distinguish if these RRLs are present on end devices or somewhere in a major transit provider's infrastructure as automated mitigation because it is expected that if the measurement is affected by the RRL then the real DDoS attacks will be as well.

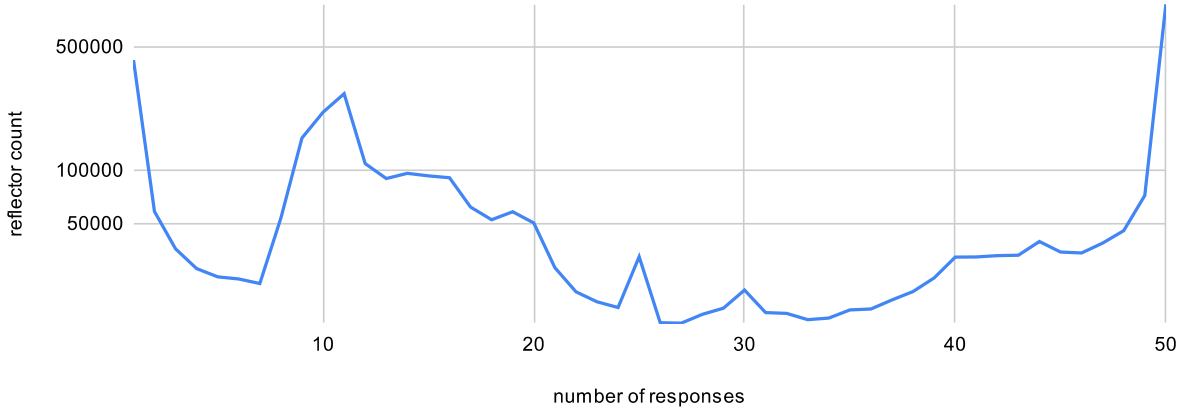

Figure 28: Count of DNS reflectors per number of received responses

Calculated attack capacity for the different RR and respective reflector counts are presented in figure 29. To accommodate UDP response loss of a few packets per individual measurement capacity calculation in this thesis selects RR 80%, for the DNS protocol attack capacity was calculated to be **27.5 Tbps** which is the highest of any measured protocol in this thesis. The author estimates that DNS has been the highest sustained global attack capacity contributor since the DDoS attacks began purely based on the open resolver count. The reflector count dropoff was much more significant than the capacity around RR 40% which closely corresponds to the identified RRLs of 20 and below. It indicates that these 1,534,096 reflectors were already contributing insignificant attack capacity. The calculated capacity was generated only from 1,262,640 reflectors representing only 15% of the scanned results and 31% of the ones responding properly as open resolvers.

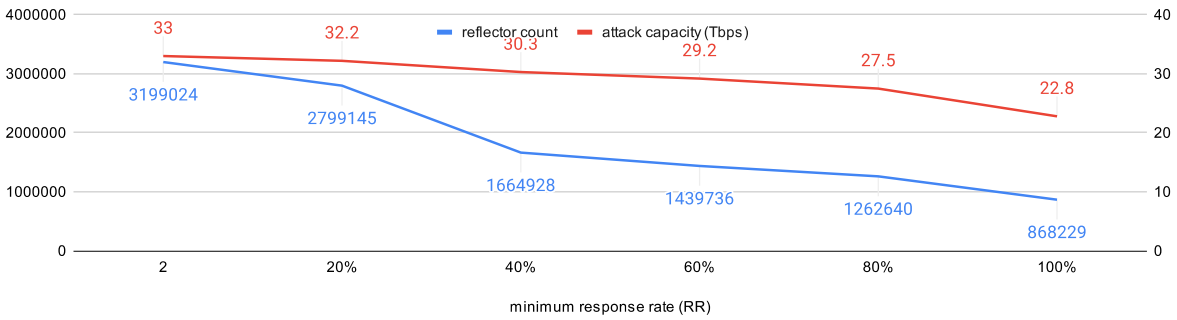

Figure 29: DNS reflector count and attack capacity for different minimum response rates

DNS reflector speed distribution (RR 80%) is presented in figure 30. 59% of the reflectors responded with speeds below 10 Mbps, these could be CPE and other devices having residential or other low-speed network connections. Around 13% of the reflectors responded with speeds below 1 Mbps, these could be excluded in a more complex capacity calculation but still present an interesting classification challenge to determine if these are especially low power, having some limiting or having a low-speed connection, e.g., wireless. There is a reflector count increase just below the 100 Mbps which corresponds to both the physical limitation of Fast Ethernet and a common ISP connection speed offering. Speeds in hundreds of Mbps are achievable on physical gigabit network connections and with sufficient processing resources generally high performance servers in data centers, only 1.9% of the reflectors contributing to the capacity had a measured speed of 200 Mbps or more. Speeds above 900 Mbps were calculated for an insignificant number (0.017%) of reflectors, these are primarily servers having multi-gigabit network connectivity located geographically (and network-wise) close by the measurement system, the measurement system's 1 Gbps network connectivity is the limiting factor for these reflectors.

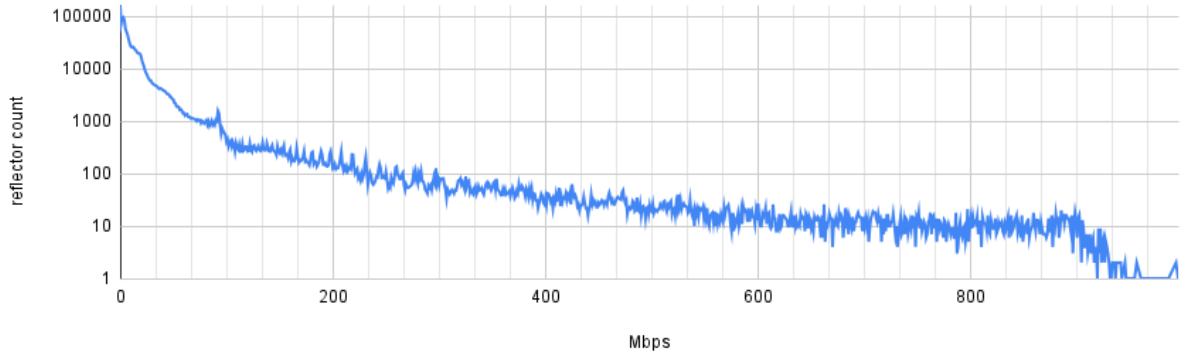

Figure 30: DNS reflector speed distribution

The geographic distribution of the measured attack capacity (RR 80%) is presented in figure 31 where all countries that were measured providing less than 1 Gbps of attack capacity are excluded as insignificant contributors. The largest attack capacity contributors are China with 5.6 Tbps, the USA with 4.3 Tbps, France with 1.4 Tbps, Russia and Brazil with 1.3 Tbps. These top contributors are not unexpected and they correlate not only with the other measured protocols but also with the generic quantitative scanning and the Internet measurement research. The USA is a country with a disproportionate network device presence because of its historic development and China has provided Internet access to a large population.

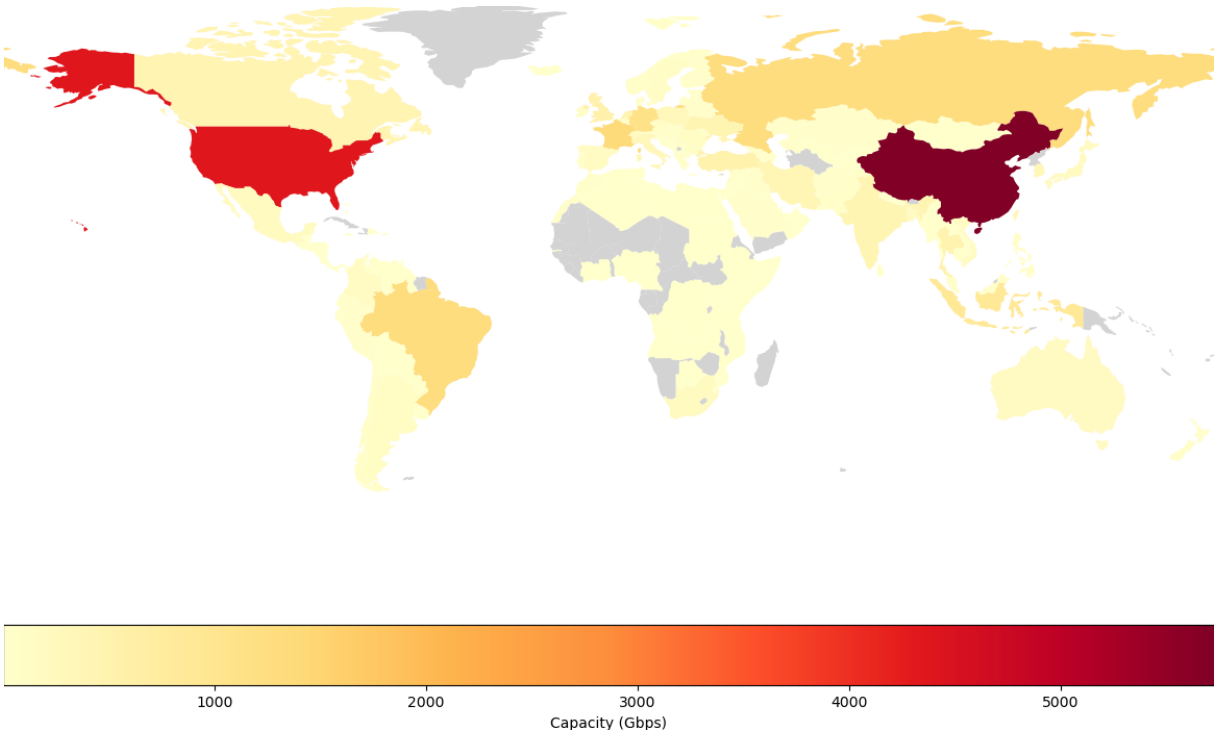

Figure 31: Geographic distribution of DNS attack capacity in Gbps

The top 15 networks contributing the most to the DNS (RR 80%) attack capacity are presented in table 3. China expectedly dominates this list with the top network contributing almost 9% to the total attack capacity but because of the Internet being under governmental control, it is not possible to extract from this data what kind of end clients contribute the most. AS networks can be physically separated thus the geolocation enhancement creates a repeating network in the list of contributors. Cloudflare as one of the world's largest CDN and DDoS mitigators is unexpected in this list, likely these DNS resolvers are part of the infrastructure providing public service and probably have implemented RRL which is higher than the measured (50), because of the high-speed network connectivity limited number of these servers produce large capacity. The same is true for the reflectors located in OVH and Hetzner data centers which are one of the world's largest but these might be clients' servers that are misconfigured. Large ISPs (AS209, AS3462, AS23969) that provide Internet connectivity to every kind of customer might contain CPE and other low-power devices providing the capacity through a large number of limited contributors but the device classification is required to confirm that.

ASN	AS name	Country	Gbps
4134	Chinanet	CN	2490
13335	CLOUDFLARENET	—	1708
4837	CHINA UNICOM China169 Backbone	CN	1271
16276	OVH SAS	FR	1114
209	CENTURYLINK-US-LEGACY-QWEST	US	788
4847	China Networks Inter-Exchange	CN	418
4538	China Education and Research Network Center	CN	325
13335	CLOUDFLARENET	US	305
24940	Hetzner Online GmbH	DE	300
8075	MICROSOFT-CORP-MSN-AS-BLOCK	US	268
3462	Data Communication Business Group	TW	257

16276	OVH SAS	CA	225
23969	TOT Public Company Limited	TH	203
7922	COMCAST-7922	US	194
6724	Strato AG	DE	194

Table 3: Top AS measured attack capacity contribution for the DNS

Overall DNS has the largest attack capacity of any measured protocol but because it has been so widespread and abused for so long there have been attempts at remediation besides the reflector elimination. Aggressive rate limits indicate effective widespread remediation and mitigation either on a device level or somewhere in a network, to determine that a device classification and extraneous Internet routing and network peering data are required for analysis. The author suspects there might be other common RRLs above 50 and it is justified to rerun this measurement at least once using 100 measurement requests and review the produced capacity and RRL at RR 90%. During this research author has observed undocumented transit provider behavior mitigating suspected DDoS traffic containing DNS packets, it is likely that there are other unpublicized mitigations being deployed not detectable in the presented data views or measurements not reaching the activation threshold. That would explain the lack of record-breaking DNS DDoS attacks targeting individual victims being reported nowadays.

4.4. SSDP

Universal Plug and Play (UPnP) is a set of standards and network protocols allowing devices with various functionality from different manufacturers to inter-connect automatically. Simple Service Discovery Protocol (SSDP) is part of the UPnP and is partially built on top of HTTP/1.1 and operates over UDP port 1900 [132]. SSDP is an older protocol defined in 1999, it permits automated network service discovery without an arbiter and configuration from end users by relying on multicast discovery [133]. SSDP and UPnP implementing products primarily are targeted towards home and small business users where end users are not expected to have the ability to configure the device besides plugging in a network cable or setting a Wi-fi password, and no in-person installation support is available. These types of devices suffer from a large number of security and privacy issues that stem from a lack of automated updates, insecure default configuration, and a lack of management.

4.4.1. How protocol is abused

In 2013 an industry report raised serious concerns about UPnP including a large number of devices readily exploitable using known vulnerabilities and 81 million devices exposing SSDP service to the Internet [134]. Although the protocol is older the SSDP based DDoS attacks started being commonly observed in 2014 [135]. In 2017 Cloudflare reported mitigating the SSDP DDoS attack exceeding 100 Gbps capacity [63]. From the IXP viewpoint, SSDP was the third most abused protocol for DDoS attacks [39].

SSDP payloads containing the `M-SEARCH` command are almost always static and it requests supporting UPnP devices to respond with the list of services it provides [133]. This response format is predefined to include meta information and redundant data – repeating response headers (see appendix B). These repeating headers enable compatibility between

implementations and deployments where the response might be contained within a single IP packet or multiple responses are generated with a separate packet for each service. Regardless of the implementation, the amplification is always present as a single properly formatted service definition exceeds the request payload size. Most of the devices provide multiple services thus increasing BAF. SSDP is designed to be employed in a LAN setting to provide local service discovery there are no legitimate use cases for it to be reachable on the Internet. Because of poor SSDP implementation by device manufacturers and insecure default configuration, these requests are processed and responded to when received on public IP addresses facing the Internet.

4.4.2. Scanning and measuring abusable SSDP reflectors

In this research SSDP payload of 84 bytes bundled with `zmap` [121] is being used. The author has confirmed that honeypots commonly receive the same scanning payloads. As SSDP is built on top of the HTTP this payload is human readable: `M-SEARCH * HTTP/1.1\nHOST:239.255.255.250:1900\nST:ssdp:all\nMAN:"ssdp:discover"\n`.

This request payload targets `239.255.255.250` which is a multicast IP address, even though the same address is used as the destination of the IP packet in local networks for Internet scanning unicast IP addresses are used for which reflectors respond in the same way.

For the amplification and rate limit detection, exactly the same payload as in the scanning stage is being used. There is no universal RRL configuration for all SSDP reflectors therefore to detect the rate limit the measurement count is set to 50 (default in this research).

4.4.3. Attack capacity

This section presents a single measurement of the SSDP protocol conducted on May 3, 2020. At the scanning stage, only 456 reflectors replied with a smaller response than the request and were excluded from the measurement stage in which 149,000 reflectors were measured. To achieve theoretical BAF >1 response payloads of 84 bytes (size of the request) and below are filtered out (213 reflectors). The analyzed response payloads start at 110 bytes therefore all reflectors are having BAF of >=1.31. Compared to other measured protocols SSDP reflectors overwhelmingly generate amplification.

Total measured attack capacity of the SSDP range from 1019 Gbps (for at least 2 received responses) to 734 Gbps (RR 100%). SSDP is experiencing a gradual capacity decrease with RR increase – 981 Gbps (RR 20%), 895 Gbps (RR 40%), 855 Gbps (RR 60%), 808 Gbps (RR 80%). This section reviews the properties of the calculated attack capacity at RR 80% – **808 Gbps**.

The count of reflectors per number of received responses presented in figure 32 identifies multiple SSDP specific spikes. With very high confidence the author concludes that there are four common RRLs (not including the initial scanning response) around 17, 35, and 44 responses and the more aggressive limit of 2 responses. SSDP gradual capacity decrease with RR increase but no extra capacity loss at RR 100% is explained by very limited last packet loss. This differs from some of the other measured protocols but the data set is insufficient to determine the root cause of this discrepancy. It could be a simple variance in the Internet behavior (e.g., load, other attacks, day of the week) or could be protocol-specific causes.

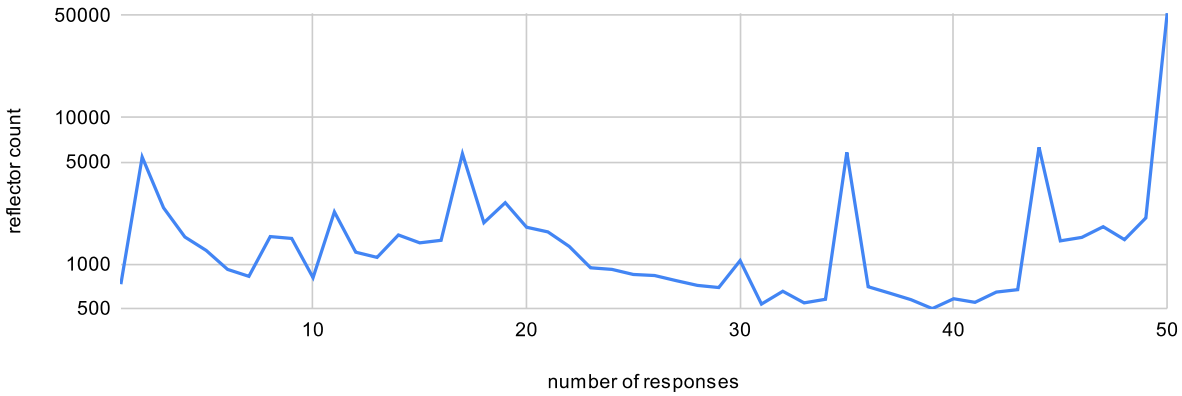

Figure 32: Count of the SSDP reflectors per number of received responses

The total measured SSDP attack capacity (808 Gbps at RR 80%) geographic distribution is presented in figure 33 where all countries that were measured providing less than 1 Gbps of attack capacity are excluded as insignificant contributors. SSDP reflectors individually contribute limited bandwidth (e.g., compared to CLDAP) to the protocol capacity which can be observed by the limited number of countries exceeding the threshold. The top 5 countries contribute 63% to the attack capacity. Taiwan albeit being a relatively small country is the largest contributor with 219 Gbps (27%) of the calculated attack capacity. Followed by Russia, Switzerland, France, and the Netherlands with the respective 119 Gbps, 78 Gbps, 56 Gbps, and 39 Gbps contributions to the attack capacity. This geographic distribution indicates that there might be end-user devices improperly installed and managed by a few national level ISPs.

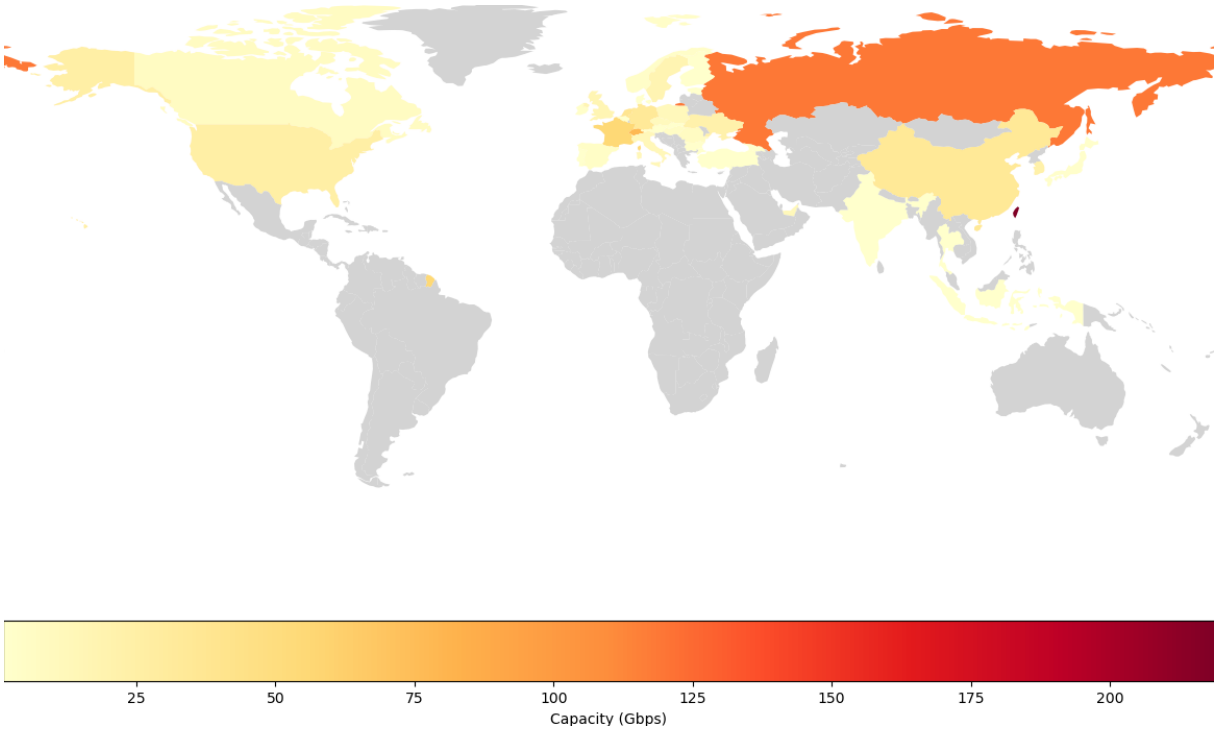

Figure 33: Measured capacity geographic distribution of the SSDP protocol

Networks contributing the most to the SSDP (RR 80%) attack capacity are presented in table 4. The largest Taiwan ISP that operates under the brand name HiNet contributes 209.5 Gbps which is 26% of the global attack capacity of the SSDP and 96% of Taiwan's

contribution. HiNet provides residential and business end users with Internet connectivity therefore it is likely that SSDP is present on the HiNet deployed devices (Internet gateway routers) with insecure default configuration exposing SSDP on the public interface. Few more ISPs (e.g., PVimpelCom) that provide Internet connectivity to end users are likely having the same end-user device deployment problem. World's largest European data center (OVH, Hetzner, LeaseWeb) presence is unusual, and more likely that these reflectors are honeypots or software (e.g., multimedia) implementations of SSDP rather than end-user devices deployed in data center setting. The most interesting is the second highest capacity contributor Infomaniak Network from Switzerland which has not been a noteworthy contributor for any other measured protocol. This hosting provider offers both general and specialized services from the public offering most suspect are Multimedia and NAS Synology. A random sampling of responses confirms through the Server header that these reflectors are `Synology/DSM` which is a network area storage device.

ASN	AS name	Country	Gbps
3462	Data Communication Business Group	TW	209.5
29222	Infomaniak Network SA	CH	64.7
8402	PVimpelCom	RU	22.7
16276	OVH SAS	FR	22.6
4837	CHINA UNICOM China169 Backbone	CN	20.5
24940	Hetzner Online GmbH	DE	18.5
60781	LeaseWeb Netherlands B.V.	NL	17.6
5384	Emirates Telecommunications Corporation	AE	12.3
21409	Ikoula Net SAS	FR	9.7
13188	Content Delivery Network Ltd	UA	9.2
4808	China Unicom Beijing Province Network	CN	7.9
17858	LG POWERCOMM	KR	7.6
3786	LG DACOM Corporation	KR	7.4
20860	Iomart Cloud Services Limited	GB	7.3
3216	PVimpelCom	RU	6.8

Table 4: Top AS measured attack capacity contribution for the SSDP

SSDP response size distribution is presented in figure 34. It excludes a negligible number of outliers (286 reflectors) that are having response sizes above 7116 bytes and skewing visualization. The largest received response payload was 154,517 bytes thus causing a BAF of 1839.4. There are multiple noteworthy response size spikes which in some cases with the surrounding response sizes resemble normal distribution that could be produced by the same or similar (model, maker, or firmware) devices. The most common response size of 3028 bytes was received from 16,448 reflectors producing a BAF of 36. The second most common response size of 2298 bytes was received from 14,586 reflectors producing a BAF of 27.4. The third most common response size of 2004 bytes was received from 5870 reflectors producing a BAF of 23.9. The fourth most common response size of 7116 bytes was received from 4311 reflectors producing a BAF of 84.7.

SSDP responses containing indirectly identifying device properties not only permit grouping and further classification but by being also human-readable, enables the author to review individual responses. The most common response payload size (3028 bytes from 16,448 reflectors) single real response is presented in appendix B. Random sampling of this

response size indicates predominantly identical responses with only the varying fixed length unique identifier (uuid). In this case, every response consisted of 10 packets each containing repeating headers causing amplification. This response exposes services (InternetGatewayDevice, WANDevice, WFAWLANConfig, etc.) that unmistakenly identify that these are Internet gateways with Wi-Fi functionality most likely residential routers. The Server header value `Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0` indicates the Linux kernel version released in 2003 although it is common to use older Linux versions for low-power embedded devices it is likely that these are outdated devices persisting online until the end of their lifetime. A 2018 report identified 181,986 reflectors with exactly this header value [136].

This response on its own is insufficient to conclude if it is a single model, single maker, or a common firmware for this device classification is required. Networks containing these responses overwhelmingly provide Internet connectivity to residential clients. Capacity contribution wise Taiwan is in the first place followed by Russia, few individual networks make up the most capacity for Taiwan while for Russia there are many ISPs. This pattern might indicate that these devices are ISP owned (managed) preconfigured routers installed in new residential client installations and persisting until physical failure that could take even a decade or more.

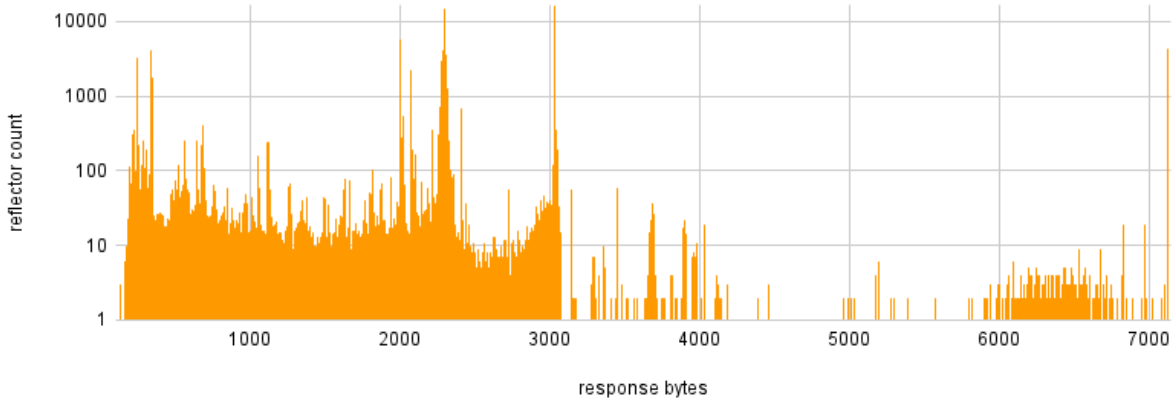

Figure 34: SSDP average response payload size distribution

The SSDP reflector (RR 80%) measured speed distribution is presented in figure 35. The majority of the reflectors (59.6%) were measured to have a speed below 10 Mbps, generally, these reflectors are considered low contributors across the measured protocols. 27.3%, 6.3%, 2.4% of the reflectors were measured to have respective speeds of 10, 20, 30 Mbps (10 Mbps ranges). Meaning 95.7% of the reflectors had measured speed below 40 Mbps. The low speed of reflectors suggests that these might be low-power devices with limited speed Internet connectivity corresponding with the ISP offers for residential users. There were 51 reflectors having speeds between 250 Mbps and the maximum measured 710 Mbps.

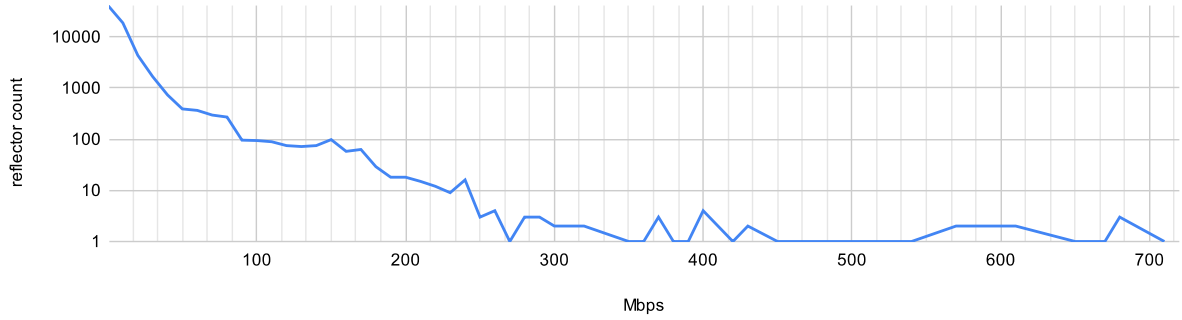

Figure 35: SSDP reflector speed distribution

Overall SSDP was measured to have a significant DDoS attack capacity. It has experienced extraordinary remediation quantity-wise from 81 million in 2012 [134] to 149 thousand reflectors in the presented 2020 measurement data. If this reflector count was reported today it would have caused record-breaking DDoS attacks and while the available network capacity was much more limited in 2012 it still could have caused much larger attacks if attackers had started abusing it to the full extent. Extensive remediation could have been motivated more by the vast number of vulnerabilities rather than reflection potential. RRL has been measured to be present in some implementations and the limited number of high-contributing networks can remediate the SSDP issue even further. Leaked information about reflectors themselves (e.g., Server header) can be used for device classification.

4.5. SNMP

Simple Network Management Protocol (SNMP) is a network management protocol enabling both remote monitoring and configuration functionality. This functionality can be accessed individually or managed centrally for a whole organization. SNMP is well defined and standardized via dozens of RFC, it has three major versions (SNMPv1, SNMPv2, SNMPv3) which have evolved over time to provide more functionality and security features (e.g., only SNMPv3 supporting encryption and advanced administration) [137]. SNMP is an industry standard and is widely implemented in a wide variety of networked devices from low-end residential Wi-Fi routers to high-end industrial PLCs. SNMP can be installed on user workstations but more commonly on servers to integrate the organization’s device management in a central server. SNMP listens for UDP packets containing all standard requests on port 161 and for asynchronous trap requests on port 162 [138]. DDoS attacks and consequently this measurement utilize the former.

4.5.1. How protocol is abused

As SNMP is an old industry standard but not a legacy protocol and is still deployed in a large number of new devices it has been abused for a long time. In academic literature potential for the SNMP reflection abuse was discussed as early as 2001, although the presented evaluation has remained relevant to this day the reached conclusion “likely not a threat” [139] has proven to be false. Historical underestimation of reflected DDoS potential has to an extent contributed to the current situation. New attack vector disclosure and public technical reports by mitigators are more recent trends therefore it is not established what were

the first major SNMP attacks and their properties. The Spamhaus project reported SNMP as a newer type of DDoS attack they received in December 2011 that was comparable capacity-wise to the largest DDoS attacks reported at that time [140]. Cloudflare reported that it mitigated the SNMP DDoS attack with 21 Gbps capacity in late 2011 [67]. Industry raised serious concerns about SNMP DDoS in 2012 [141]. Therefore it can be considered that significant abuse of the SNMP began in 2011 and has been continuing to this day. From the IXP viewpoint, SNMP was observed to be 7th most abused protocol [39]. Cloudflare reported that in the second quarter of 2022, SNMP was still one of the top network-layer attack vectors but significantly less prevalent than the other measured DNS, SSDP, and CLDAP protocols [28].

SNMP protocol design follows the manager-agent pattern where agents present on the end devices expose multiple commands to managers for getting and setting data. One of these commands `GetBulkRequest` requests a response containing a potentially large amount of data including large data tables [142] thus causing amplification. This is the command that has been abused for the DDoS attacks.

SNMP follows the classic DDoS abuse pattern. A large number of devices are deployed or sold with enabled SNMP protocol by default. Although genuinely useful on its own (in internal networks) the default configurations expose this protocol to the Internet which has almost no legitimate use cases. Meanwhile, these devices are abused for reflection and are not properly managed by the owners, meaning they could be contributing to the attack capacity potentially until the end of their life.

4.5.2. Scanning and measuring abusable SNMP reflectors

In this research SNMPv2 payload of 33 bytes containing the `GetBulkRequest` command is being used in both the scanning and measurement stage. This specific payload has been observed by the author to be the most predominant one (c. 2019) being received by the author's honeypots (outside the scope of this thesis) on the SNMP port from both research and malicious scanners. SNMP major versions are not fully backward compatible but all SNMPv2 and SNMPv3 compatible implementations must support `GetBulkRequest` command [143]. Meaning this research excludes SNMPv1 from the scanning and measuring although depending on implementation SNMPv1 devices might respond with an error packet, those generally provide no or insignificant amplification. Although higher amplification is possible by targeting SNMPv1 specific commands it is far below what can be amplified by `GetBulkRequest`. Measuring SNMPv1 would require scanning and measurement implementation as a separate protocol which is not justified as there are no recent reports indicating that specifically SNMPv1 is being abused.

For the amplification and rate limit detection, exactly the same payload as in the scanning stage is being used. There is no universal RRL configuration for all SNMP reflectors therefore to detect the rate limit the measurement count is set to 50 (default in this research).

4.5.3. Attack capacity

This section presents a single measurement of the SNMP protocol conducted on the May 29, 2020. At the scanning stage, 484,242 reflectors replied with a smaller response than the request and therefore were excluded from the measurement stage in which 1,351,441

reflectors were measured. As per methodology the minimal response value selection for the capacity analysis is subjectively determined by an expert (the author for this thesis) on a per-protocol basis. As this filtering is post measurement stage it can be adjusted for the completed measurements to extract different views. One of the determining factors can be the review of the unfiltered response size distribution itself. Responses few bytes above 33 bytes theoretically create BAF >1 in practice because of packet loss it would produce real-world BAF <=1. SNMP protocol measurement contains a significant portion of the reflectors (31.6% when filtered to minimum 33 byte responses) having this low theoretical BAF of 1 (33 byte response) to 1.18 (39 byte response). The author claims that these are non-amplifying/non-contributing reflectors to the global SNMP DDoS attack capacity.

SNMP is a perfect case demonstrating the benefits of the proposed measurement methodology vs. normal scanning and theoretical estimates. All of the latter approaches assume that 484,242 (scanned but not measured) and 386,675 (33-39 byte responses, even more with the ones having responses <33 bytes) reflectors are either high risk or high contributing depending on the metric used. Which is 47.4% of all the 1,835,683 reflectors identified in the scanning stage which are disregarded even before the capacity analysis begins. Global SNMP attack capacity (RR 80%) including all reflectors with BAF>=1 (>33 byte responses) calculated to be 4.06 Tbps. But the question stands: what is the calculated capacity difference?

This section discusses attack capacity calculations for the reflectors that have replied with 40 bytes or more (BAF >=1.21). Global SNMP attack capacity (RR 80%) was calculated to be **2.47 Tbps**. Meaning this 4.06-2.47=1.59 Tbps difference is achievable only if the attacker has the 1.59 Tbps of spoofing capacity available in which case it would be more efficient to flood the victim directly with spoofed packets because of less packet loss (no loss to reflectors and on reflectors). This spoofing capacity is not available to individual attackers, only large botnets, e.g., Mirai, can generate (without spoofing) this traffic volume.

Total measured attack capacity of the SNMP range from 2.67 Tbps (for at least 2 received responses) to 2.29 Tbps (RR 100%). SNMP is experiencing a more gradual capacity decrease with RR increase than some of the other measured protocols – 2.62 Tbps (RR 20%), 2.59 Tbps (RR 40%), 2.54 Tbps (RR 60%), 2.47 Tbps (RR 80%). The initial aggressive filtering decision likely facilitated that.

Although gradual capacity decline doesn’t indicate any RRL being present the count of reflectors per number of received responses presented in figure 36 points to a sharp decline around 27-29 responses. This most likely identifies the RRL of 30 present on a large subset (tens of thousands) of reflectors. Interestingly there is no additional capacity drop at these 30 responses (RR 60%) meaning that these reflectors could be producing small responses. Response analysis or possibly device classification would be required to determine if these devices were produced by the same manufacturer or use the same software.

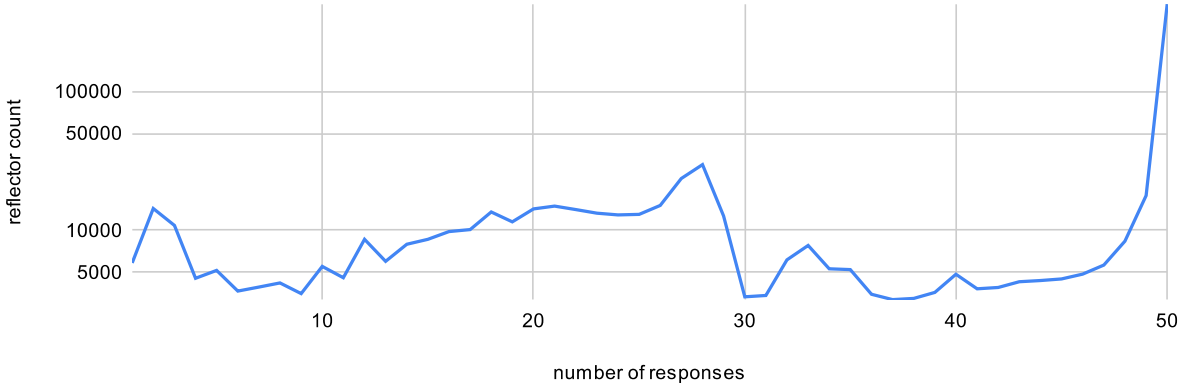

Figure 36: Count of the SNMP reflectors per number of received responses

The total measured SNMP attack capacity (2.47 Tbps at RR 80%) geographic distribution is presented in figure 37 where all countries that were measured providing less than 1 Gbps of attack capacity are excluded as insignificant contributors. Japan is the highest contributor with 660.8 Gbps of attack capacity, followed by the USA, China, Russia, and India with the respective 355.3 Gbps, 183.4 Gbps, 112.9 Gbps, and 96.2 Gbps capacity. The top five countries produced 55.7% of the total attack capacity. Any scanning or theoretical estimate research would rank South Korea as the highest risk or contribution purely based on the reflector count of 193,973 (in the analyzed filtered subset) which was the highest while Japan was having 76,079 reflectors (4th place by count). South Korea was calculated to have only 24.5 Gbps attack capacity meaning these reflectors on average are insignificant contributors likely having RRL and low BAF reinforcing that reflector count is an unreliable metric and justifying the methodology presented in this thesis.

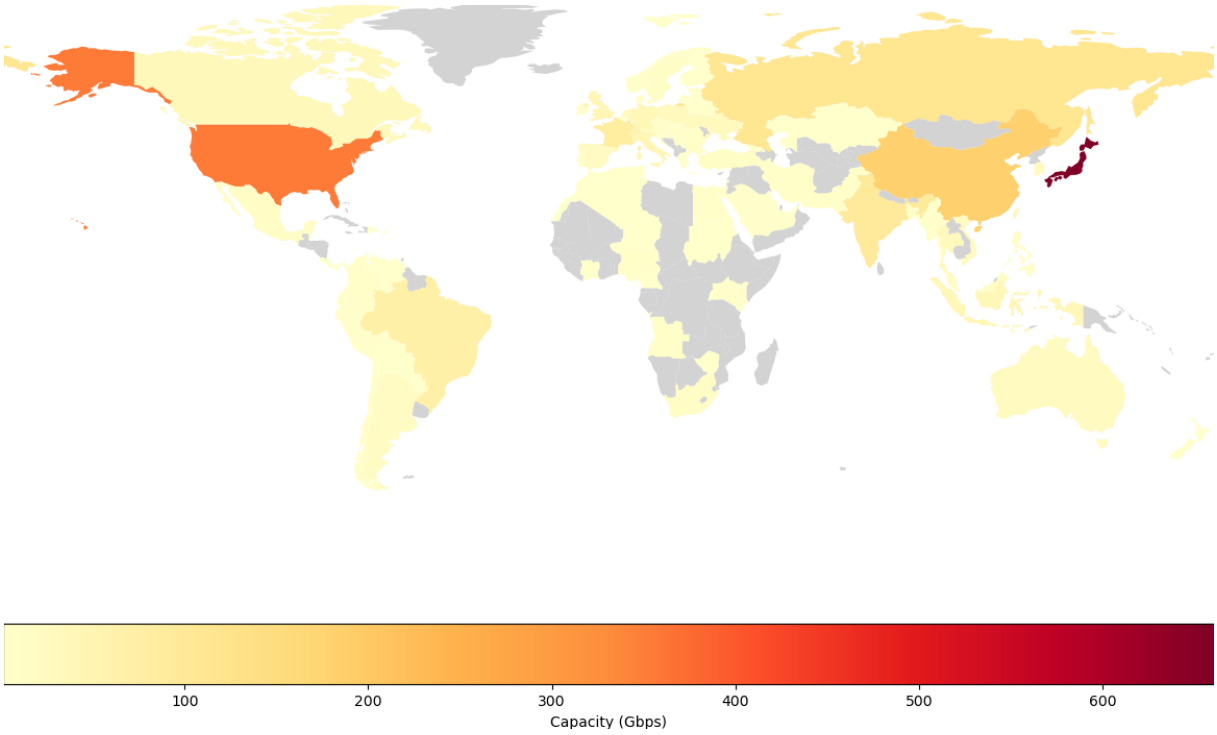

Figure 37: Geographic measured capacity distribution of the SNMP protocol

Networks contributing the most to the SNMP (RR 80%) attack capacity are presented in table 5. These top 15 networks contributed 37.1% to the calculated capacity. Japan is

overrepresented compared to the other protocol measurements. The most contributing network FreeBit (15.1% of the capacity) provides an infrastructure to ISPs and various service providers. Other top contributors from Japan all provide Internet and telecommunication services to both businesses and individuals. The author speculates that in Japan's case, there are either end-user or infrastructure device vendors that have default configuration permitting SNMP and these devices are being deployed by various ISPs. As OVH is providing data center services these SNMP reflectors might be installed on the servers by administrators and be misconfigured or rely on default installation. The remaining networks provide comparable services and might contain both residential and infrastructure devices or servers acting as the reflectors significantly contributing to the measured capacity.

ASN	AS name	Country	Gbps
10013	FreeBit Co.,Ltd.	JP	381.7
4134	Chinanet	CN	81.5
18126	Chubu Telecommunications Company, Inc.	JP	76.2
7679	QTnet,Inc.	JP	53
34310	Penta SA	CH	53
16276	OVH SAS	FR	47.8
17488	Hathway IP Over Cable Internet	IN	46.9
4837	CHINA UNICOM China169 Backbone	CN	41.5
18144	Energia Communications,Inc.	JP	31.3
17511	OPTAGE Inc.	JP	30.1
13768	COGECO-PEER1	US	22.9
7522	STNet, Incorporated	JP	20.3
7922	COMCAST-7922	US	19.7
3462	Data Communication Business Group	TW	18.4
8100	ASN-QUADRANET-GLOBAL	US	14.5

Table 5: Top AS measured attack capacity contribution for the SNMP

SNMP response size distribution is presented in figure 38. It excludes a negligible number of outliers (464 reflectors) that are having response sizes above 5143 bytes and skewing visualization. The largest of the excluded responses was 115,950 bytes having BAF 115950/33=3513.6. There are another 13 one-of reflectors with responses ranging between 13,347 and 34,834 bytes.

Because of the purpose of SNMP to produce information about the device individual response sizes or response size clusters might group together devices from the same manufacturer possibly even the same model. The most common response of 1928 bytes producing BAF of 1928/33=58.4 was received from 133,560 reflectors and is almost an order of magnitude more prevalent than any other. This and most of the other sizeable responses are not equal in the contents as these have variance in the response tree contents as opposed to small-size responses containing static error messages or empty response trees.

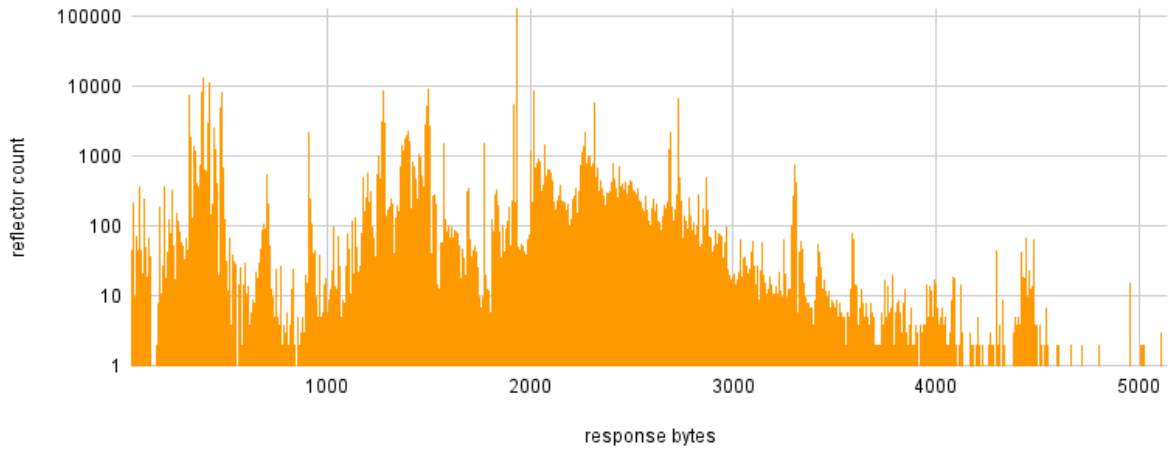

Figure 38: SNMP average response payload size distribution

The SNMP reflector (RR 80%) measured speed distribution is presented in figure 39. Overwhelming 82.3% of the reflectors were measured to have a speed below 10 Mbps, generally, these reflectors are considered low contributors across the measured protocols. 11.2%, 3.6%, 1.1% of the reflectors were measured to have respective speeds of 10, 20, 30 Mbps (10 Mbps ranges). Meaning 98.3% of the reflectors had measured speed below 40 Mbps. These low speeds for a large number of reflectors suggest that these are low-power devices with limited-speed Internet connectivity. Based on the research in [3.4. Classifying devices on the Internet] and data in table 5 the author speculates that these might be residential routers deployed by a limited number of ISPs having unnecessarily enabled SNMP by default. There are only 7 individual reflectors having speeds between 300 and 880 Mbps.

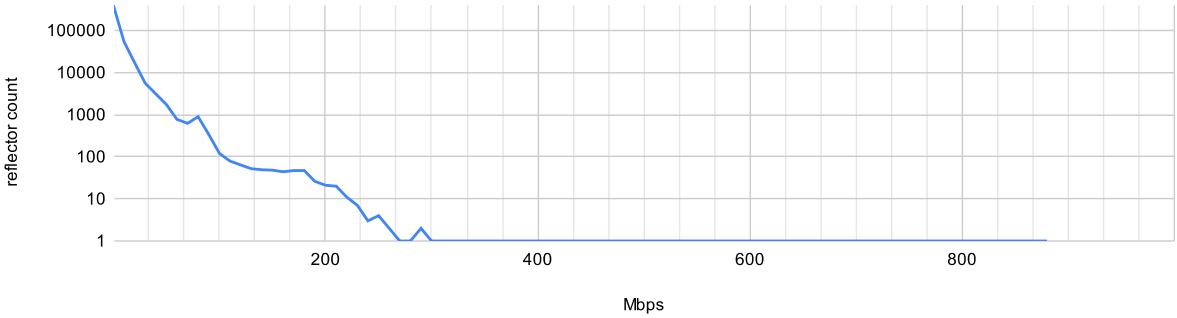

Figure 39: SNMP reflector speed distribution

Overall SNMP was measured to be widely spread on the Internet and totaling in a large contribution to the global attack capacity. There is a RRL of 30 likely identified but it doesn't affect the capacity. Device classification is required to understand if the author's speculation about a small number of vendors providing residential end devices for a limited number of ISPs is a cause for a large share of the measured SNMP attack capacity.

4.6. CLDAP

Connection-less Lightweight Directory Access Protocol (CLDAP) is based on LDAP (with a restricted set of operations) while utilizing UDP (port 389) for transport protocol, originally not having any modern security considerations [144]. The primary justification for the CLDAP was the removal of TCP overhead in a local network where UDP functions

reliably. CLDAP has never been fully standardized and widely implemented and has been retired [145]. Most of the CLDAP reflectors have been identified as Microsoft's Active Directory (AD) service implementations featuring a partial CLDAP implementation that exposes only a single command – LDAP ping [146], [147].

4.6.1. How protocol is abused

CLDAP has been reported in media to be abused for reflection as early as 2007 [148] while industry and academic sources began consistently reporting on CLDAP DDoS only in late 2016. This discrepancy has no definitive explanation, either there has been a misreport or earlier abuse was insignificant enough not to report on it. Akamai reported that it first started mitigating CLDAP on October 14, 2016 [64]. On January 7, 2017, an attack peaked at 24 Gbps with BAF reportedly reaching 70 and averaging at 56.89 [64].

Considering that in 2016 it has been reported that of 78,908 publicly reachable CLDAP reflectors 98.3% were Microsoft AD variants and 1.6% were compatible Samba AD implementations [147] only Microsoft AD compatible services are being abused therefore only this functionality is to be scanned and measured. Per Microsoft AD specification LDAP over the UDP transport clients can only perform two specific anonymous operations (rootDSE search and LDAP abandon) which are intended for use by LDAP ping requests used for the AD Domain Services domain controller [149]. Therefore a small "ping" request generates an amplified response containing the server properties producing BAF ranging from 56 to 70 [146].

4.6.2. Scanning and measuring abusable CLDAP reflectors

AD expects LDAP ping request from a client to contain a rootDSE search query for example: `(&(DnsDomain=abcde.corp.microsoft.com)(Host=abcdefgh-dev)(User=abcdefgh-dev$)AAC=\80\00\00\00)(DomainGuid=\3b\b0\21\ca\d3\6d\d1\11\8a\7d\b8\df\b1\56\87\1f)NtVer=\06\00\00\00))` [149]. Generally, malicious actors prefer request payloads to be as small as possible. In this research, a payload bundled with `zmap` [121] 53 bytes in length is being used. This payload contains the filter `(objectClass=*)` which per RFC 4512 requires a server to "provide information about itself and other information that is specific to each server, this is represented as a group of attributes located in the rootDSE" [150]. Although it can be refined even more down to 39 bytes which increases BAF to around 67 [151] there is insufficient evidence of backwards compatibility therefore the standard payload is being utilized.

For the amplification and rate limit detection, exactly the same payload as in the scanning stage is being used. There is no universal RRL configuration for all AD servers therefore to detect the rate limit the measurement count is set to 50 (default in this research).

Considering that most if not all reported reflectors are servers there is diminished concern for overloading those compared to embedded devices, practically none with the selected measurement request count. Considering that there are only AD compatible CLDAP reflectors supporting a single command there is no concern regarding payload selection or tuning. Overall there are no special considerations required for the CLDAP reflector scanning and measuring for this research.

4.6.3. Attack capacity

This section presents a single measurement of the CLDAP protocol conducted on May 25, 2020. 2,293 reflectors that replied with a smaller response size than the request in the scanning stage were not measured. This is a low proportion compared to some of the other measured protocols. In total 51,886 reflector measurements were conducted. All reflectors that responded in the measurement stage with payloads that are 53 bytes or less (size of the request payload) are filtered out as no amplification occurs therefore all analyzed reflectors provide BAF >1. The filtered set consists of 7,884 reflectors that replied at least once in the measurement stage.

Total measured attack capacity of the CLDAP range from 961.5 Gbps (for at least 2 received responses) to 551.6 Gbps (RR 100%). Gradual capacity decrease with RR increase is completely expected – 913.1 Gbps (RR 20%), 898.7 Gbps (RR 40%), 885.7 Gbps (RR 60%), 869.7 Gbps (RR 80%). Significant measured capacity drop-off at RR 100% is common among most protocol measurements and is to be expected. In this case, the drop-off is borderline suspicious, other CLDAP measurements have less disproportional drop-off between RR 80% and 100%. Although it could be indicative of a sizeable subset of reflectors having RRL at 50 (RR 100%), it could also be a variance in a global network traffic flow (e.g., congestion) causing UDP packet drop at different rates on different measurement dates.

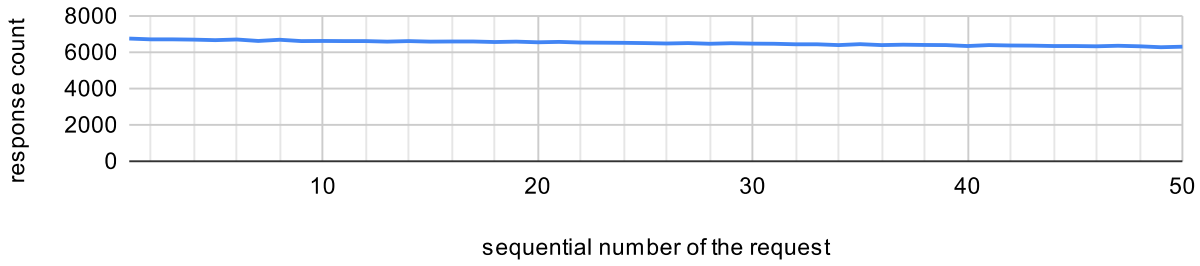

Figure 40: Number of responses for the every CLDAP request sent

Measurement data enables the author to evaluate the uncertain RRL presence further. The count of the CLDAP reflectors per number of received responses is presented in figure 41 it clearly pinpoints the cause of sudden capacity drop-off (RR 100%) to be a response loss of 2 packets (4%). But this still isn't enough to draw a conclusion. Response counts for the sequential requests are presented in figure 40 although technically there is no guarantee for UDP packets to be received in the order sent aggressive RRL and anomalies can be detected which is not the case here instead expected steady decline is observed. Combining these two data presentations together enables the author to conclude that the significant capacity drop-off (RR 100%) likely occurs because of the normal function of the Internet and not present RRL. Overall data doesn't indicate any widespread RRL to be present. Simultaneously the measurement data can't determine if any RRL is present above the selected measurement count.

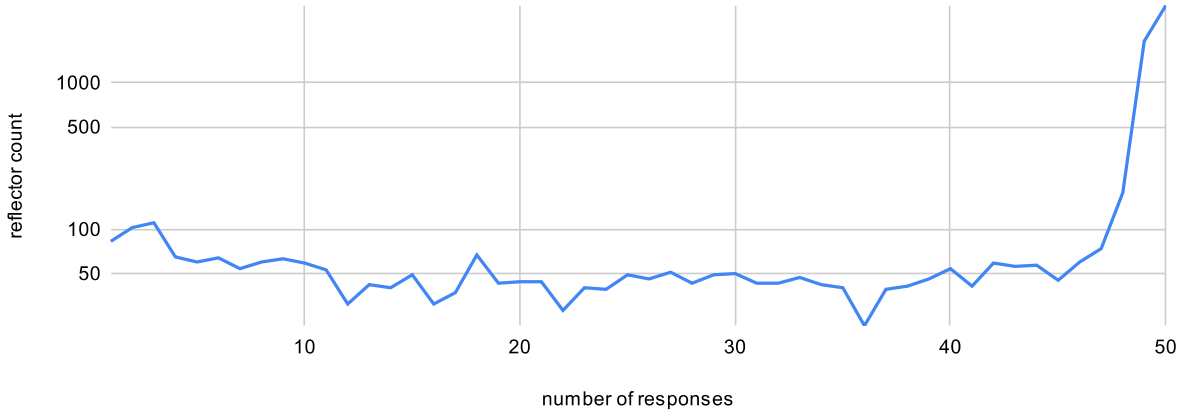

Figure 41: Count of the CLDAP reflectors per number of received responses

The total measured attack capacity of **869.7 Gbps** (RR 80%) for the CLDAP is presented in figure 42 where all countries that were measured providing less than 1 Gbps of attack capacity are excluded as insignificant contributors. The top 5 countries for the CLDAP contribute 57.6% of the overall capacity. All of these countries are developed – USA 252.5 Gbps, Germany 104.5 Gbps, Netherlands 55.4 Gbps, Great Britain 47 Gbps, and France 41.6 Gbps, substantiating that CLDAP reflectors are misconfigured enterprise systems.

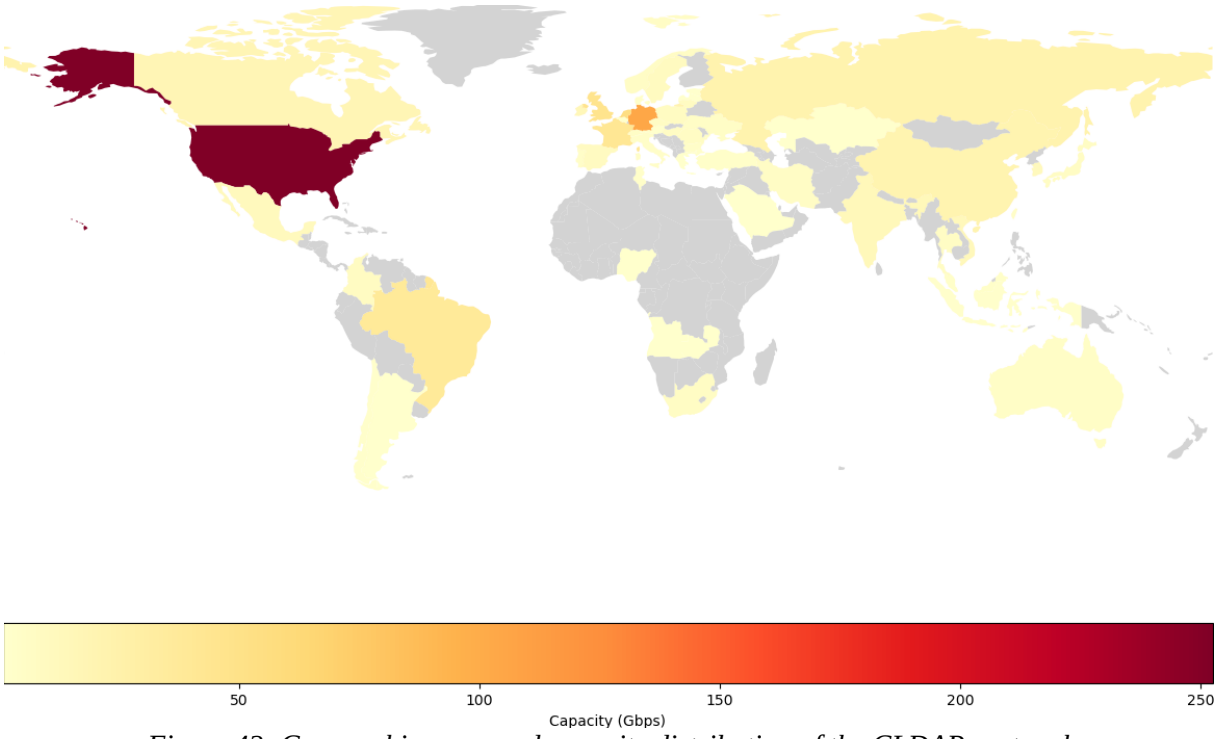

Figure 42: Geographic measured capacity distribution of the CLDAP protocol

Networks contributing the most to the CLDAP (RR 80%) attack capacity are presented in table 6. Although separating globally dispersed AS by country generally provides more insight, in this case, the opposite is true while Microsoft's ASN 8075 is the highest contributor the global capacity for this AS is more remarkable. Microsoft's ASN 8075 located in all countries altogether contributes 137.4 Gbps which is 15.8% of the total measured attack capacity. Therefore affirming that a product (AD) by Microsoft is most commonly located in its network as part of service offerings. The remainder of the top contributing networks are as

anticipated – the world's largest data centers providing dedicated servers and cloud solutions (Hetzner, OVH, Online) and the world's largest cloud providers (Google, Amazon). It is likely that these providers are used by both individual companies and service providers that provide AD hosting. Iomart provides managed hosted Microsoft solutions thus there could be an issue with default deployment configuration or end-user misconfiguration. Interserver (ASN 19318) provides affordable unmanaged virtualized Microsoft Windows servers which might attract less experienced administrators causing misconfiguration. This AS capacity distribution further confirms that these CLDAP reflectors are AD services. Protocols common on low-power residential devices would not have this type of overwhelming data center and cloud provider distribution.

ASN	AS name	Country	Gbps
8075	MICROSOFT-CORP-MSN-AS-BLOCK	US	66.5
24940	Hetzner Online GmbH	DE	55.6
8075	MICROSOFT-CORP-MSN-AS-BLOCK	NL	34.8
53755	IOFLOOD	US	15.3
16276	OVH SAS	FR	14.3
8075	MICROSOFT-CORP-MSN-AS-BLOCK	IE	11.4
15169	GOOGLE	US	10.7
20860	Iomart Cloud Services Limited	GB	10.5
19318	IS-AS-1	US	10.4
12876	Online S.a.s.	FR	9.3
8560	1&1 Ionos Se	DE	9
8972	Host Europe GmbH	DE	8.8
16509	AMAZON-02	US	8.3
8151	Uninet S.A. de C.V.	MX	8.3
14618	AMAZON-AES	US	8.1

Table 6: Top AS measured attack capacity contribution for the CLDAP

Response content analysis is outside of the scope of this research but per CLDAP ping specification it can leak some information about AD, e.g., identify precise ownership. Response size distribution is presented in figure 43 it is fairly even in comparison to other measured protocols. There are no responses smaller than 81 bytes resulting in minimal observed BAF 81/53=1.52 but only extremely small reflector subset responds with low BAF. The maximum measured BAF is 3162/53=59.66. More than 97% of measured reflectors are having BAF above 35 which is significant in comparison to other measured and overviewed protocols.

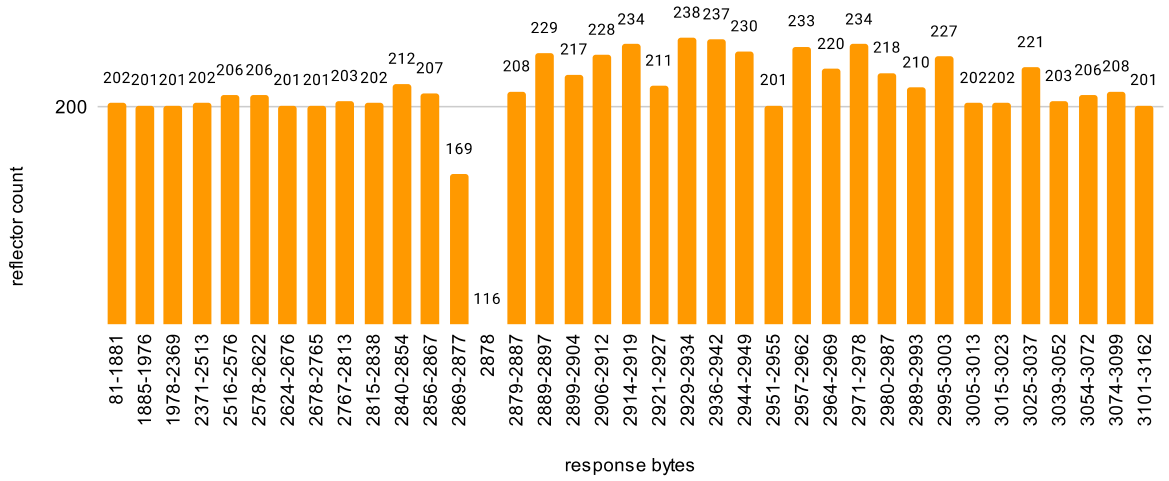

Figure 43: CLDAP average response payload size distribution

The CLDAP reflector measured speed distribution is presented in figure 44. 12% of the reflectors had speeds below 10 Mbps and can be considered low contributors to the overall capacity. 57% of the reflectors had speeds up to 100 Mbps. About 43% of the reflectors had speeds from 100 Mbps up to maximum measured 952 Mbps, and only 6 reflectors had speeds above 900 Mbps. Malicious actors would prefer to abuse the largest contributors (100 Mbps or more) if no RRLs exist, if RRLs do actually exist then anything besides low contributors would suffice.

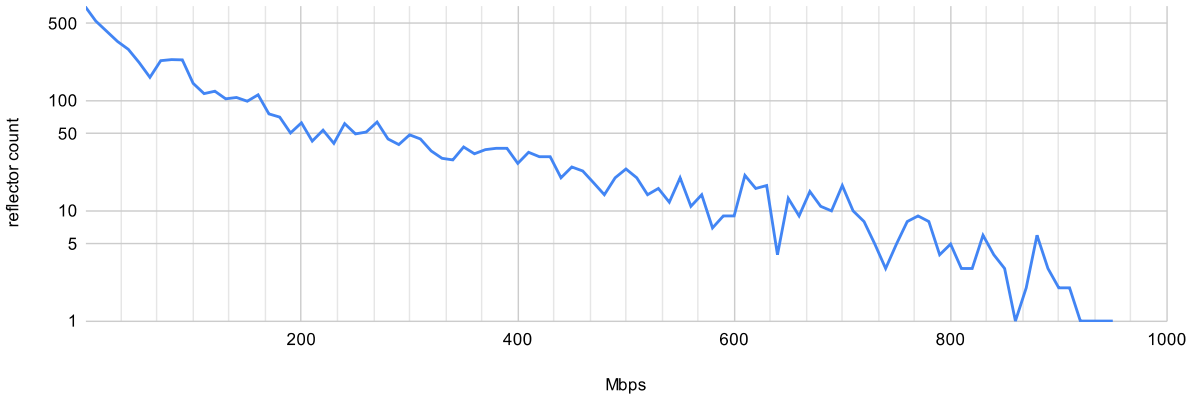

Figure 44: CLDAP reflector speed distribution

Overall CLDAP has been measured to be a protocol with a comparatively small reflector set having large BAF and no (or insufficient) RRL totaling in noteworthy global attack capacity of 869.7 Gbps (RR 80%). This combination makes protocol prime for abuse but reviewed literature reports attacks in only tens of Gbps. More detailed investigation is warranted to determine if a higher RRL is present, and are the measured reflectors honeypots.

4.7. Memcached

This protocol is unique compared to all the other measured ones in multiple ways. Most of the long-term abused protocols are present on CPE or other low-power devices reachable on the Internet. Memcached is generally deployed in an enterprise environment where each high-performance server can have unrestricted gigabit or 10-gigabit connectivity, as this

protocol is providing high-performance service then the software is not a bottleneck either. Meaning each reflector could fill all the available bandwidth capacity thus negatively affecting the performance of its primary functionality that could be detected by the administrators. Notification efforts also could reach the responsible administrators that have the incentive and capability to act upon them. As this protocol sufficiently affected mitigation service providers, they actively participated in the remediation efforts [62].

The Memcached protocol has been the record holder for the reflected DDoS attack size against an individual target since 2018 with the reported observed attack capacity of 1.7 Tbps [60]. The author has measured its attack capacity to be only 319 Mbps in May 2020, contributed by only 12 reflectors which could have been aggressive honeypots, measuring methodology allows to exclude insignificantly contributing hosts from the calculation. Therefore the author considered this protocol to be fully remediated which will likely not have a resurgence. Because of the current way how the attack capacity is being understood decision makers and the public could be wrongly assuming that the most referenced number in the capacity context is undisputably relevant to the current state of affairs. Which is not the case. How quickly was this protocol remediated? For how long this protocol and attack size has been wrongly considered a major concern?

In the same way as the peak attack is observed in a specific network and point in time a singular measurement is no different. The solution for the “next Memcached” is to have a system continuously measuring attack capacity for each actively abused protocol where newly abused protocol monitoring could be added quickly. No further periodical measurements are needed for the seemingly fully remediated protocols like Memcached as the likelihood for these protocols to be abused again is low and can be caused only by the default configuration bundled with the application package or the source code.

The author confidently reported this full protocol remediation in a paper [10] and presented it at the 16th International Conference on Cyber Warfare and Security in 2021, without receiving any pushback from the reviewers, audience members, and later on readers. In a 2021 paper, Kopp et al. [39] reported their observations from a major European IXP privileged vantage point. The paper included observed Memcached attacks reaching 37 Gbps of attack size from 1,556 reflectors. And while the analyzed data were collected between September 23, 2019 and April 20, 2020 and is just a month before the author’s measurements (the latest version presented in this thesis) commenced in May 2020, the author presumes that this discrepancy rather illustrates limitations of the proposed measurement methodology instead of the possibility that full remediation occurred in a span of 1 month. Implemented [4.9.1. Data validation] explicitly can’t guarantee the data quality for the first measurement of a protocol having a low reflector count because of the lack of a stable baseline (minimum few scanned reflectors per minute, preferably few measured per minute) and lack of the pre-existing baseline. Whilst it accentuates another author’s claim – there already are underutilized approaches suitable for the remediation which are discussed in [6. Remediating DDoS attacks].

4.8. Other protocols

The attack capacity monitoring system has been designed with the ability to add new protocols easily, meaning, any potential, newly or previously abused protocol can be added conveniently to the monitoring system. This section discusses some of the other potential candidates for measurement implementation that are present in different types of academic and industry sources [28], [35], [39], [68], [73]. Any UDP-based protocol can be reflected but for large-scale abuse causing sizeable DDoS attacks three preconditions must be met – a large number of reachable reflectors, significant BAF, and lack of RRL, the protocols discussed in this section lack one or more of these properties preventing them from being record breakers. The protocol set discussed in this section is not meant to be exhaustive but rather overviews different protocol categories and their distinct properties.

4.8.1. CharGEN

Character Generator Protocol (CharGEN) is a historic protocol defined in RFC 864 as a useful debugging service that simply sends data without regard to the input [152]. This data is character sequence preferably a recognizable pattern such as printable ASCII characters. Although CharGEN supports both TCP and UDP, in the scope of this thesis only UDP is reviewed. Per RFC 864 CharGEN UDP on port 19 responds with a random number (between 0 and 512) of characters (bytes) in one datagram as a response to each received datagram [152]. While this was not a concern in 1993 when the protocol was defined, presently it is unimaginable that any new typical protocol would be defined with such an inherent amplification and reflection by design. CharGEN is the easiest protocol to abuse for DDoS attacks and to implement for the measurement from all discussed in this thesis.

From the IXP viewpoint, CharGEN was reported to be the eighth most abused protocol for DDoS attacks peaking at 7.6 Gbps observed attack capacity [39]. Real BAF has been reported to be 358.8 [153] which is significantly higher than expected from the protocol definition. Implementations might ignore the protocol definition and skew BAF upwards by not having low character count responses or implementing a higher upper limit for the response size.

There are no indications that CharGEN attack capacity could be increasing making it a low priority for the attack capacity measurement. As there are no common use cases for this protocol nowadays it is rarely deployed by default and a natural decline in the quantity of reflectors is expected. The primary interest would be a singular measurement for the detailed analysis. Legacy deployment reduction can already be analyzed through simple protocol scans conducted by many research parties.

4.8.2. RPC

Open Network Computing Remote Procedure Call (RPC) specification contains a port mapper service that enables programs to bind to nondefault ports where the RPC port mapper is queried first by the connecting clients to determine the actual port to use. RPC port mapper abuse for the DDoS attacks was first reported in 2015 with BAF reaching 27 [154]. From the IXP vantage point, RPC has been reported to be the sixth most contributing protocol for the DDoS peaking at 33 Gbps attack capacity [39]. In 2022 RPC was reported to be more

contributing to the attack capacity than the measured SNMP and NTP protocols [28]. Because of the consistent presence in the DDoS contribution rankings the RPC port mapper is prioritized for the measurement implementation.

4.8.3. Industrial protocols

Industrial protocols can be present on physical ICS devices and workstations managing these devices. Industrial devices vary greatly in their purpose and the negative effect that an attacker can cause. There has been a monumental effort of reducing publicly reachable industrial devices (which partially include IoT devices) via academic and industry scanning and notifying efforts [94], [97]. The primary concern is the compromisation of devices to change their state or disable functionality, these devices are generally susceptible to DDoS and even DoS because of limited processing power. Rarely industrial protocols are considered for the role of reflectors.

For an industrial protocol to be abused for the DDoS attacks there are specific conditions to be met besides supporting UDP communications. There should be a significant number of reachable devices on the Internet. These devices should simultaneously require remote access (functioning) capability and not cause catastrophic consequences if compromised, thus justifying reachability and ignoring received warning notifications from scanners and tasked governmental entities, e.g., CERTs. These devices should have sufficient processing power and networking connectivity which allow the processing of incoming packets (many per second) without adversely affecting device functionality. It is rare for these conditions to be met together, BACnet is one case.

BACnet is a communication protocol for building automation and control enabling devices produced by different vendors to interconnect over a network [155]. BACnet is a complex protocol designed to accommodate any type of automation that could be required in a building. For research purposes, BACnet generally is considered an ICS protocol [94]. In 2017 BACnet scanning identified 16,400 publicly reachable devices for which amplification scans were conducted in which 30 % of all BACnet devices allowed for a BAF of 20 or larger peaking at 120 [156]. In 2017 after distributed security advisory European IXP vantage point identified total bandwidth flow below 100 Mbps on the BACnet port range, a large portion most likely being scanning traffic [157]. Maximizing BACnet BAF requires reconnaissance scans and payload tuning which makes this protocol unattractive for attackers to abuse. For the same reason, it makes measuring this protocol not worth the effort.

IoT protocols sometimes are grouped with ICS in scanning research. The difference is the maximum catastrophic consequences that the compromised IoT and ICS devices can cause. IoT devices have experienced explosive growth (including publicly reachable deployments) and various protocols have been developed for those. CoAP is a simple UDP protocol that is intended for low-power devices (IoT) on unreliable networks which has been abused for DDoS attacks and has a BAF of 34 [158]. Although IoT specific protocol development and deployment are to be monitored (and potentially measured), the record-breaking 1.2 Tbps DDoS attack was a direct flood not involving amplification and reflection from 500,000 IoT devices compromised by Mirai malware [159], and additionally, many of the IoT devices already implement widely abused protocol such as SSDP and SNMP [160].

4.8.4. VPN protocols

Virtual private network (VPN) protocols have exploded in usage caused by privacy and security concerns, and the growth of remote working. VPNs generally offer secured remote access to a private network. By design most VPN servers have to be reachable on the Internet thus turning the ones utilizing UDP protocol into reflectors, meaning, only RRL is a valid remediation path.

OpenVPN is one of the most well-known VPN protocols widely deployed in the industry. OpenVPN literature almost exclusively focuses on preventing DDoS attacks against the OpenVPN infrastructure, e.g., using hash-based message authentication codes [161]. OpenVPN implementing servers are listening on UDP port 1194 for new clients and established connections, when a client wants to initiate a new session it sends a special initiation packet to which the server responds multiple times until receives acknowledgment or exhausts the resending limit [162]. As the spoofed victim will not respond with the expected acknowledgment this retry feature addressing UDP unreliability creates both packet and bandwidth amplification proportional to the resending limit.

OpenVPN has been suspected of being already abused for DDoS reflection years before industry reports and other research publications [48]. IXP viewpoint demonstrated OpenVPN being the twelfth protocol by maximum observed attack capacity peaking at 4.7 Gbps [39]. NSFOCUS measured the average BAF to be 5.9 [163] which is insufficient to cause a record-breaking or even a major DDoS attack without a large reflector set and a colossal spoofable bandwidth investment from a malicious attacker. A later industry report by Corero Network Security observed attacks reaching 30 Gbps capacity and indicating a growing reachable reflector count [162].

This development prioritizes OpenVPN measurement implementation over all other potential protocols discussed. Not only OpenVPN deployment is growing but many of those deployments are industry grade (high-powered servers with high capacity network connections) combined with uncertain real-life RRL and no calculated global average BAF which might be a warning sign that protocol could be abused to a greater extent than previously observed. Any other VPN protocols utilizing UDP are potential candidates for abuse and might justify exploratory measurements. As VPN implementations are generally considered security software, security features such as RRL are expected to be present which in combination with limited reflector count might explain the lack of publicly known other VPN protocol abuse.

4.8.5. Gaming protocols

Gaming is a niche that has been oftentimes ignored or excluded when discussing computing in general. Although it has experienced explosive growth and has become acknowledged use of technology, the DDoS cases publicly discussed are only major attacks against gaming infrastructure causing worldwide disruption in entertainment. There are also attacks against individual gamers when a specific gaming protocol (or other means) leaks the IP address. Finally, services implementing gaming protocols can be abused for the reflection, while centrally managed servers and game clients can be remediated by the developers if a major abuse is detected, the game servers that are individually managed and game clients that

implement multiplayer server functionality might remain susceptible to reflection for a long time. Most of these multiplayer use cases require public reachability.

Legacy games with limited or no centralized multiplayer options are especially susceptible as exposing network ports to the Internet might be the only way how to play it with other people and the implementations are less likely to have a RRL present. Quake3 network code part has been ported to many games therefore all of them inherit amplification possibility with a BAF of around 15 [151]. Source game engine is used in many once popular shooter games and has a BAF of around 5 having 25,000 reflectors [151]. A more modern case is Steam in-home streaming which enabled remote gaming and had a BAF of 17 and totaled around 347,000 reflectors until it was fully remediated after a software update [151].

Although there are numerous end-user gaming configurations susceptible to amplification abuse neither academic [39] nor industry [28] sources report any noteworthy contribution to the DDoS attack capacity. Therefore measurement implementations are not warranted at the moment.

4.8.6. OS specific protocols

Network services running on end-user computers which are built-in by default in an OS can become major contributors to DDoS attacks. A specific OS version can be deployed on millions of computers where an abusable service may be enabled by default for all or some of them, or might be enabled manually for a large subset of those. End-user computers have sufficient processing power and may have high-speed Internet connectivity which provides public reachability deliberately or because of a misconfiguration. While servers (having the same or different default services) can be fitting into this category, most mobile devices don’t have direct reachability and thus are unlikely to be abused in this fashion. These properties make default OS services prime for large-scale abuse while simultaneously providing an efficient remediation path through automated OS updates.

Apple Remote Management service (ARMS) is a remote management service often used for organizational management of macOS systems. In 2019 it was reported that ARMS has been abused for the first time in DDoS attacks and reached a BAF of 35.5 having around 54,000 reflectors [164]. Further confirming that a major university released a security advisory that a significant number of their macOS systems participated in DDoS attacks advising disabling the service or firewalling it [165]. From the IXP vantage point, ARMS was reported to be the ninth most abused protocol for the DDoS peaking at 6.2 Gbps observed attack capacity [39].

Network Basic Input/Output System (NetBIOS) is a legacy network protocol suite that includes a name service (WINS on Microsoft Windows OS) which is only needed within local networks and only for legacy use. There is no legitimate use for it on the public Internet. Although this service is a legacy one with dwindling numbers only in 2015 first DDoS attacks have been reported then reaching 15.7 Gbps capacity [166]. NetBIOS has a comparatively low BAF reaching 3.85 [167] which combined with a limited reflector count explains the low prevalence of abuse for DDoS attacks.

4.8.7. Discovery protocols

Discovery protocols are designed to enable services and devices of a specific class or made by a specific vendor to discover and establish communications between themselves in an automatic way. These protocols are almost exclusively designed to function on internal networks and not be publicly reachable. Although a multitude of discovery protocols have been defined only a few have sufficient numbers of network misconfiguration enabling public reachability. IXP vantage point identified WS-Discovery and Ubiquiti Device Discovery protocols being significantly abused for the DDoS reflection in the real world [39].

Ubiquiti produces a wide variety of networked devices for both residential and commercial uses, most of the produced device models share part of the firmware which includes device discovery functionality. A simple 4-byte request payload generates a large response containing properties of the device (e.g., name, model, firmware version) causing BAF up to 35 [168]. It was reported to be the eleventh most abused protocol from an IXP vantage point in 2019 peaking at 5.2 Gbps attack capacity [39]. In 2019 abuse generated a DDoS attack peaking at a 348.91 Gbps capacity utilizing 24.57% of the reachable Ubiquiti devices [169]. Cloudflare reported that in the second quarter of 2022 Ubiquiti device discovery abuse was experiencing growth and was considered an emerging threat to their infrastructure while not being a significant one in actuality [28].

Web Services Dynamic Discovery (WS-Discovery) is a standardized protocol utilizing multicast to locate services (e.g., printers) on a local network [170]. Although the standard is not new, only in 2019 the widespread abuse for DDoS attacks was reported, in addition to standard UDP reflection cause the secondary cause is that the WS-Discovery probes are intended to use the LAN scoped multicast IP address, however, implementations are responding to these probes when using a unicast IP address [171]. A later report indicated that 87.7% of the WS-Discovery implementing devices, or approximately 700,000, were video surveillance devices [172]. This camera prevalence is explained by the Open Network Video Interface Forum (standardizing organization) recommendations to implement WS-Discovery since 2010 [173]. WS-Discovery was reported to reach a BAF of 15.3 [171], exceeding the real-world DDoS attack capacity of 350 Gbps [173].

Discovery protocol implementations on networking devices (including the ones requiring remote access, e.g., IP cameras) make those susceptible to default misconfiguration unnecessarily exposing abusable service on the public interface and consequently to the whole Internet. Combining the reachability of these devices with limited remediation possibilities for the end users prioritizes measurement implementation.

4.9. Data quality

For any Internet measurement research arguably the most challenging aspect is data quality. The implemented research methodology always relies on the scanning stage which is susceptible to acknowledged data quality issues that are either addressed or accepted in this research field but the measurement stage raises some additional unique issues. In the context of current research, the main considerations are individual measurement and long-term data quality. These are affected by primarily external factors. This section discusses identified data quality issues and how these were addressed.

4.9.1. Data validation

Internal factors – measurement system setup, developed code, OS, and hardware performance in the context of data quality and ability to process all the incoming data are not discussed in detail here. The implementation decision to conduct slow scans alleviates most of the data quality issues that could be caused by exceeding the system's resources, e.g., fully loaded CPU or network memory buffers would call into question if the conducted individual measurement timings were affected by any delays. The overall system has been developed in an iterative process and relies on high-performance tools for which the author hasn't observed hitting any limits (CPU, RAM, network buffers, etc.) or being overloaded.

Human error does play a role in the data quality. Simply not offloading completed measurement data from the measurement server to the analysis server can fill the storage and invalidate the last conducted measurement entirely. Monitoring exit codes and error messages of the used software tools for every conducted measurement permits to identify major issues and if needed disregard that specific measurement.

After the individual protocol measurement is completed a timeline is generated from the tool outputs, measurement debug logs, system bandwidth utilization, and the packet capture. Selected minute precision permits the identification of any significant anomalies while leveling off individual device measurements (e.g., a singular amplifier providing BAF in thousands). The generated result is compared to the previously established baseline. This is the most powerful approach to detect any issues with data and attempt to trace those back to the source and ultimately make a decision to disregard measurement fully.

This data validation method relies on three major assumptions. First, the randomization of destination IP addresses for the scanning and the consequent measurements produces a stable baseline of decisions and network traffic across the whole timeline. Second, DDoS defense activation should take some time (minutes or hours, not seconds) because of the destination randomization as well. Third, if any network issues arise they take significantly less time than the scanning time frame making it detectable. As the scanning rate in PPS is fixed and known then a variety of ratios can be calculated using the data from the debug sources – how many responses are received per the sent packets, how many are directed to measurement or rejected, how many are decided to be measured or rejected, how many are pre-checked successfully or fail, how many packets and bytes are received per measurement, etc. If the selected ratios are not stable across the whole timeline then manual investigation is required. It does not invalidate the measurement data automatically, there can be scenarios where a single misbehaving device can skew ratios for a part of the timeline.

The time series of a single complete SNMP protocol measurement properties conducted on May 29, 2020 is presented in figure 45. Every single reviewed measurement parameter is expected to be within a baseline range throughout the whole scan. These ranges are parameter and protocol specific, e.g., outgoing network bandwidth "ifstat_out" is almost constant primarily consisting of a constant scanning traffic flow which is slightly inflated by the individual device measurements. This protocol measurement is within the baseline range and accepted for analysis in [4.5. SNMP]. If the outgoing network flow would be varying or the incoming network would be within range but having unstable input for measurement or measurement results these would be indicating that there is an internal issue that needs to be

investigated. Many more combinations can indicate that external factors have affected the measurement and some of those are reviewed in the following section.

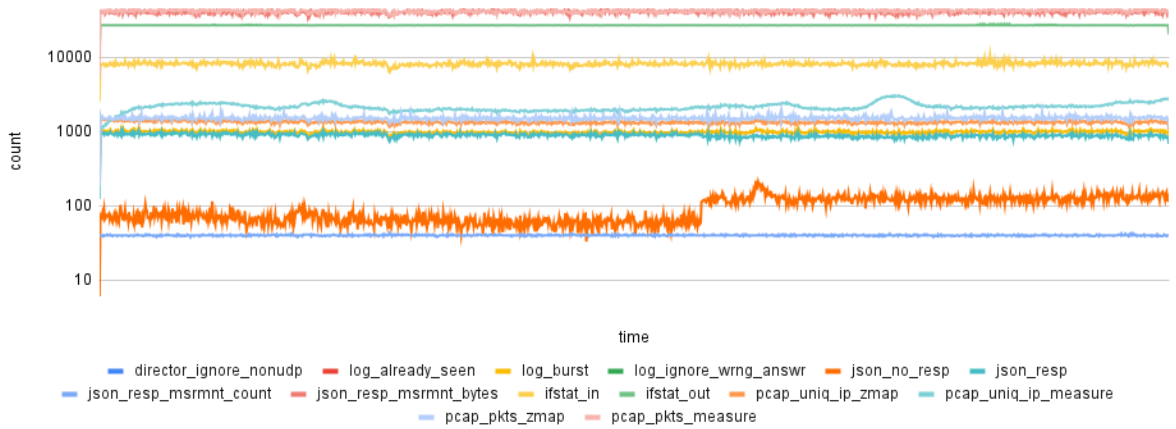

Figure 45: Time series of a full SNMP protocol measurement properties

Variance in the timeline depends on the protocol being measured. The timeline verification has proven to function well for common protocols. If a protocol is measured for the first time and has very few reflectors that respond (e.g., Memcached) then only a few ratios from the timeline can be extracted. In these cases, it is impossible to determine if the measurement produced high-quality data. Persistently monitored protocols are measured at least twice per month to mitigate both detectable anomalies in generated timelines and undetectable ones which could be caused by some of the core assumptions being violated. Comparing calculated values and raw debug data across different measurements of the same protocol can provide secondary verification. The variance between these measurements generally is significant but it is still suitable to verify that changes introduced to the code base do not affect the results in a way that those become incomparable with the previous ones.

4.9.2. Blacklisting

Temporary and permanent (across multiple scans) blocking (blacklisting, greylisting) from the scanned networks is a known issue for all Internet measurement research and scanning in general. Multiple public grey lists that include the scanning IP addresses and networks that utilize those were identified. Historically these lists have been created from honeypot data or user reports and were fairly static – adding new IP addresses would take time while removal might not have been even implemented. These lists can be interchangeably used by organizations or end-users in routers, firewalls, and network services. CZ.NIC project Turris ships routers to end-users with enabled greylist containing IP addresses used in this research [174] which is also reused by third parties. It is not feasible to detect small individual globally dispersed networks using greylists this way. Maintainers of these types of greylists do not care about intent and all scanning activity from any party is treated as undesired. A more modern approach is network intelligence which enables to make automated decisions on a case by case basis. One of the largest network intelligence services GreyNoise labels this research IP addresses as benign through communication with the author [175]. Blacklisting concerns prevent measuring rate limiting for high RRL protocol implementations as the consequences might affect subsequent scans and measurements.

The number of unique /24 subnets per protocol that have sent at least one UDP packet within a scan or measurement stage is presented in figure 46. The decrease in three out of four protocols includes both decrease in device count and an increase in network blocking. Cybergeen reflector count in figure 52 presents a similar picture of 3 protocols having a decrease in the reflector count. It might be a mix between blocking and reflector remediation. Without an independent scan from a "clean" (not included in any grey- or blacklists) IP address range not actively used in scanning or malicious activities before it is not possible to determine how many networks have blocked the measurements.

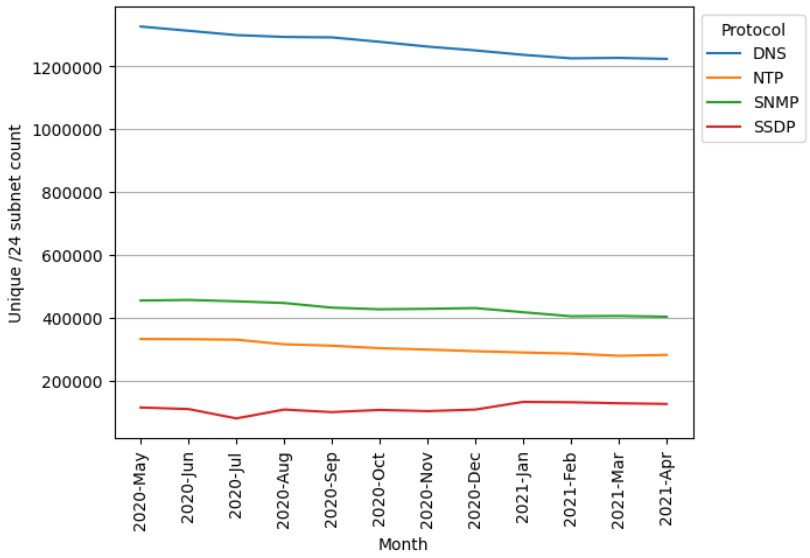

Figure 46: Unique /24 subnets per different protocols

A unique challenge that is uncommon for scanning research is DDoS defenses kicking in and affecting data quality for specific target networks. It is a proper functioning of defense systems as packet payloads being sent are the ones triggering amplification and the packets received as responses look like real DDoS attacks on a miniature scale. It is a hard problem to address properly because of slow scan rate, churn, different scenarios of defenses activating, and dynamically changing network paths. The only feasible way how to address this was for the measurement server to select a data center and transit provider that do not have automatic DDoS defenses for low traffic amounts.

Figure 45 visualizes a way how the major data quality issues are detected, including major blockings that occurred within a single protocol measurement. In this case “json_no_resp” (a reflector that reached the measurement stage but didn’t respond in it) displays a persistent increment in the second half of the measurement. This type of increase would correspond with what could be a temporary DDoS defense kicking in and staying on until the end of the measurement (plus some additional timeout). These parameters are always reviewed in combination with each other and in this case for the same time period “pcap_uniq_ip_measure” (IP addresses where the measurement packets are sent to) increased significantly which corresponds to an increase in failed individual measurements. Meaning in this case alternative explanations are more likely – the end (beginning) of a workday in a region where these devices are prevalent.

Long-term data quality decline is unavoidable in any Internet scanning research relying on a static IP address set (including this one). Even if only slow safe scans are being executed

over the years blacklists and their propagation into firewalls accumulate. Rarely active scanning research IP addresses are removed from the blacklists.

A combination of blacklisting, long-term decline, and exclusion requests can be addressed in a uniform way by establishing a simple estimate for each specific network that can't be measured anymore. This estimate can rely either on historical measurement data adjusted for global trends or external data sources. The current capacity calculation relies only on the direct measurements but could be adapted possibly with the theoretical estimates that are discussed in [5.4. Measured vs. theoretical capacity].

4.9.3. IP fragmentation

IP packet fragmentation for this type of network research is expected, the utilized `zmap` scanning tool operates at a low networking layer (before OS fragment reassembly) and does not reassemble fragmented packets. It is observed in the scanning data as the maximum received responses do not exceed 1472 bytes (maximum UDP payload for the standard MTU of 1500 bytes). For the proposed methodology this scanning behavior does not matter as the decision to measure the scanned reflector is based on the received response not being smaller than the scanning payload. All analyzed (and discussed) protocol payloads are always significantly below this 1472 byte threshold. But for the measurement stage, all the timings and calculations are for the whole UDP responses which does involve packet reassembly by the OS. Per RFC 791 fragment reassembly is a task for the "destination host" [89] – the measurement server of this research.

Therefore there are fringe cases where a reflector has responded with a small response count (e.g., 2) with these responses exceeding MTU because of the UDP and Internet routing nature fragments could be delivered to the measurement server in a different order and with different delays. The measurement server OS network stack buffers these fragments until reassembly can be completed and then delivers whole responses to the measurement software. If all these reassembled responses of a single measured reflector are delivered concurrently internally via the network stack then measurement calculations can produce speeds (e.g., 11 Gbps) exceeding the physical network link 1 Gbps.

The number of this type of occurrence is small and varies based on the measured protocol, Internet "behavior", load on the OS network stack, etc. (e.g., 1 case per 100,000 reflectors measured). Nevertheless, it could add up to 100 or more Gbps of bogus capacity for a single protocol measurement. The selected solution is simple – disregarding all measured reflectors with calculated capacity above physical link speed. The author already claims that 2 measured responses provide unreliable data and it is only ever used to highlight this point. Fragmentation is not an issue for the analyzed capacity at RR 80% (minimum 40 responses for the implemented protocols), not a single measurement has been affected at this RR in a way that resulted in an anomalous reflector measurement. Although theoretically possible at any RR, these bogus measurements almost exclusively occur at RR minimum 2 responses data set, the next common RR 20% is rarely affected.

Reassembly in the measurement implementation is conducted by the OS network stack transparently and without any knowledge from the measurement software. Some precision in speed calculations and overall capacity might be increased if individual fragmented packets

would be timed instead of whole UDP payloads. The author estimates that the achieved improvement would be negligible compared to other concerns discussed in this section.

4.9.4. Anomalies

The Internet is an anomalous place but daily Internet users do not normally experience any Internet-level anomalies instead what they perceive as abnormal usually is misbehaving local network or malfunctioning remote service. The full Internet scans do reveal a small number of anomalies that can affect the measurement stage, the two types of these anomalies did affect the testing setup and were addressed. Most common causes for these anomalies are human errors (network device misconfiguration) and protocol implementation errors or incompatibility.

A common anomaly that is often treated as normal behavior in Internet-wide scans is the IP address and port number mismatch. It occurs when a device receiving a packet responds from a different IP address or port than it was addressed to. This is undesired behavior and is unexpected in many protocol implementations, firewalls, and NAT devices thus preventing responses from being received or properly processed by the initial requester. One of the scenarios causing mismatch is the deployment of servers providing services publicly on the Internet which include multiple IP addresses therefore specific services might bind sending to a specific IP address or interface but receive packets on all IP addresses, normally it wouldn't affect normal clients as they would communicate with the proper IP addresses (e.g., pointed domain names). The port mismatch is irrelevant to the capacity measurement. IP address mismatch is detected and preserved in the data structure, as implementation is utilizing a wide port range for receiving measurement responses it is possible to distinguish noise (e.g., backscatter) from the mismatched replies.

The most significant issue that is specific to the measurement and less relevant to the common scanning is routing loops. A single routing loop can produce millions of packets as the response to a single request packet, it happens either if the singular response gets looped or the singular request gets looped and every response packet actually was produced by the reflector. A small number of looped responses (even one) can produce continuous response flow in hundreds of Mbps and sometimes even exceed 1 Gbps which causes delays and packet drops for the measurement of individual reflectors which later can produce unreliable capacity calculation. It has been observed and reported by researchers for years, primarily in the NTP context [36]. The author has observed this issue since 2014 (first conducted NTP measurements) varying in intensity (from daily to once every few months) only in Japan's AS and only in NTP measurement data sets. Starting late 2021 and up to the middle of 2022 author has observed this looping extending to other protocols and for some periods to every consecutive conducted protocol measurement. The DNS research by Nosyk et al. identified 115 routing loops in 2021 scan data acting as "mega" amplifiers abusable for the DDoS attacks which they reported to the network operators [176]. Although the IP addresses are not publicized the described behavior fully matches what is observed in the measurement research (also for non-DNS protocols) therefore these are likely the same routing loops. The selected solution is to fully pause the measurement when the physical link gets saturated until the looped traffic flow stops, usually for 10-30 minutes but it can repeat multiple times from different reflectors for a single protocol measurement.

5. GLOBAL ATTACK CAPACITY

This chapter explores factors limiting the total attack capacity, compares theoretical estimates with the measurement results, and reviews trends of the protocols measured in [4. Protocol measurements]. Global attack capacity is estimated and the largest contributors country and network-wise are identified, supplementary data views are presented. This chapter relies only on the data sets (May 2020 – April 2021) which have been published in the peer-reviewed papers, earlier data have been discarded as incomparable, and newer data are collected and processed for future articles.

5.1. Factors limiting total attack capacity

It might be tempting to sum up all the measured protocol capacity values together to produce a single value of the total worldwide DDoS attack capacity. In reality, there are two major and a wide range of minor factors that limit the attack capacity.

Every single network has a limited upload bandwidth capacity that is available for the outgoing DDoS attack traffic to be utilized. A particular network's connection capacity is directly determined by the utilized physical technology, router capability, free unused capacity of the uplink, and contractual agreement with the ISP or transit provider. The issue is that it is not clear where to draw the "border" for every network and what the capacity of every network actually is. The easiest solution would be splitting the Internet by the AS and using public information from IXP monitoring projects and estimating private peering capacity. But no accurate result is possible, the issue is that a single AS can contain a large number of separate networks with their own limits which decreases estimate quality. Even if reasonable estimates per network basis are established, then the layer of limitation could move up to the transit provider level, as their routers often are not designed to handle maximum load through all the connections same time.

There have been proposed ways how to limit unrealistically large estimates in the DDoS capacity context. A simplistic idea would be a physical throughput constraint of the largest subsea cable as the upper limit of the largest single DDoS attack possible [6] which has many technical considerations but the measured capacity in this thesis is far below this limit anyway. Another approach is to limit individual reflector contribution by the upload capacity extracted from a third-party data set [4] which still produces an unrealistic total capacity estimate of more than 100 Tbps.

Another major factor is that a single device could be providing multiple abusable services at the same time. In these cases, only the protocol providing higher bandwidth should be counted towards total attack capacity. It might be easy if the protocol measurements for each IP address happen within a short time frame (seconds or minutes) but in the designed solution it is not the case. Larger the time difference between measurements per IP the less precise it becomes. IP address reachability is affected by dynamic addressing, operating hours, network anomalies, and other factors.

5.2. Estimating total attack capacity

The bottom-up approach implemented in the individual protocol measurements which excludes individual non-contributing reflectors (based on the RR) from the protocol capacity not only eliminates false attack capacity but consequently also decreases the likelihood that the particular network's calculated capacity exceeds the free available capacity. This thesis doesn't address any individual network bottlenecks.

Overlapping protocols on a single device is the only factor limiting total attack capacity that is considered at the current stage of the research. In theory, multiple large data sets consisting of services running network protocols should have a large number of overlapping devices. The produced total attack capacity relies on 5 protocol (NTP, DNS, SSDP, SNMP, CLDAP) measurements conducted throughout May 2020. These 5 data sets are spread across the whole month therefore only devices with static IP addresses and rarely changing dynamic IP addresses (e.g., always-on CPE) are properly handled. These are the same data sets analyzed in [4. Protocol measurements] with RR 80%.

12,899 IP addresses had overlapping 2 protocols and only 10 had overlapping 3 protocols, there were no 4 or 5 protocol overlaps. For the overlapping protocols, only the maximum bandwidth contributions were selected. The overlapped excluded capacity was calculated to be 278 Gbps or only 0.86% of the all protocol capacity total. The non-overlapping total attack capacity for these 5 protocols for this thesis was calculated to be **31.33 Tbps**.

5.3. Attack capacity over time

[4. Protocol measurements] present individual protocol measurements at a single point in time in May 2020. This section reviews measured reflector count and protocol capacity over the one-year period (May 2020 – April 2021). Reflector count excludes the ones not measured for any reason (generally low BAF) and because of the required comparison between quantity and capacity, only the reflectors having 2 or more measurement responses are included in the presented data.

The reflector count per protocol in figure 47 portrays an expected picture, there is a persistent-protocol dependent proportion maintained between the minimum 2 responses and RR 80%. It is possible that the raw scanning quantitative results would reveal remediation when compared to RR 80% but only in a longer time period. Although even consecutive measurements of a single protocol can vary fairly significantly for a variety of reasons discussed in this thesis, the trend over the year(-s) can be relied on. While 3 out of 4 protocols have an expected steady decline in the reflector count caused by remediation and physical end-of-life of less secure devices, the SSDP demonstrates a slight increase. The most likely cause is the deployment of new abusable devices having no RRL as both quantitative values increase proportionally. This trend is worrying and contradicts many years of preached best practices of not deploying devices with exposed services on public network interfaces.

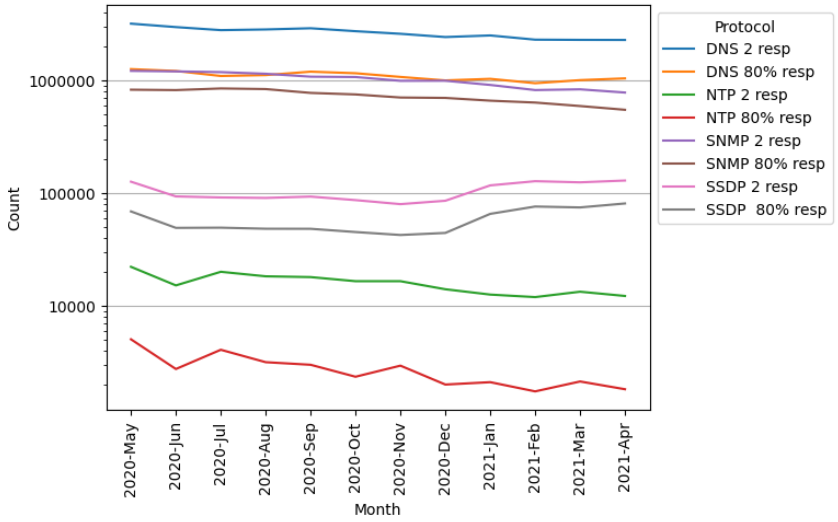

Figure 47: Reflector count for received 2 responses vs. 80% responses

Figure 48 presents the calculated attack capacity for the same quantitative data sets presented in figure 47. Comparing these two visualizations indicate a few interesting possibilities, one of that being that all the initial pre-measuring filterings in most cases already exclude a lot of non-contributors therefore relative capacity proportions are much closer. Although figure 47 presents a clear DNS reflector count decrease trend, the produced capacity is much more stable (stagnating) indicating that the reflector count decrease is fully negated by the individual reflector network connection speed increase. Which fully explains the long-term abuse of the DNS. SNMP reflector count decrease is matched with the appropriate capacity reduction. SSDP reflector count increase is matched with the capacity gain indicating that these newly deployed devices are not remediated for amplification either. The most sensible explanation would be unsecured CPE devices being deployed for new clients by mismanaged ISPs.

NTP and other highly remediated protocols can visualize the capacity difference based on RR, lack of the high RR calculated capacity would always significantly overstate how much abusable potential is available. The proposed methodology provides an excellent way how to track remediation progress but none of the measured protocols experienced a full remediation cycle within the time frame of the measurement data set. Best results would be produced if the protocols were measured for many years and newly abused would be added instantaneously.

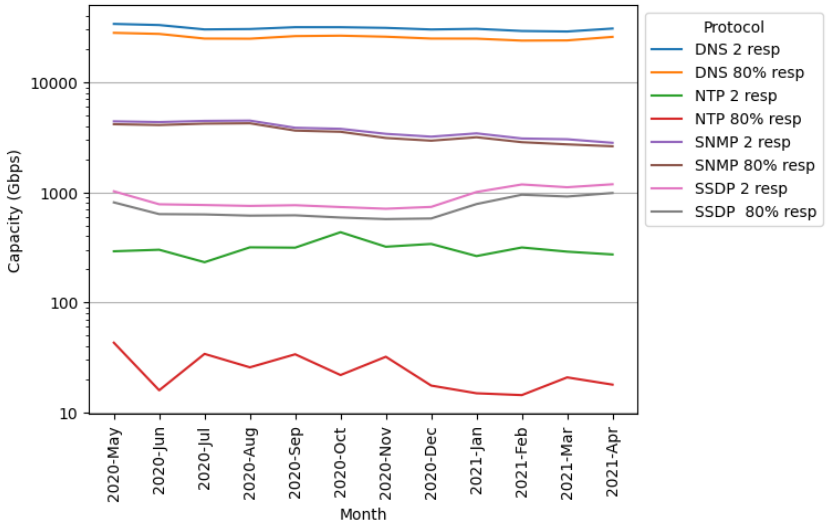

Figure 48: Measured capacity for received 2 responses vs. 80% responses

5.4. Measured vs. theoretical capacity

This section provides validation and comparison of the proposed measurement methodology of this thesis to the only existing alternative methodology [4], more details discussed together with co-authors are published in [6], in the same paper author of this thesis has replicated and updated theoretical capacity methodology to match relevant timeframe of this research and available data sets. Measurement methodology provides measured or empirical capacity while the alternative methodology [4] provides theoretical or potential capacity.

5.4.1. ASN Estimates

The top 5 ASN (with country code) per each protocol are presented with the theoretical capacity in figure 49 and the measured capacity in figure 50. These Sankey diagrams also clarify a contribution across those ASNs and by protocol, which enriches the understanding of the top ASN contributors. Specifically, it can be observed that the top ASNs often contribute across more than one protocol. It is not possible to overstate the value of this insight, in that it also permits remediators to focus any interventions with ASNs across more than one protocol. Furthermore, it demonstrates that they are not uniform contributors and that the top contributors often dwarf the contribution of those further down the list. These two facts alone imply a national or international policy intervention efficiency that could be exploited.

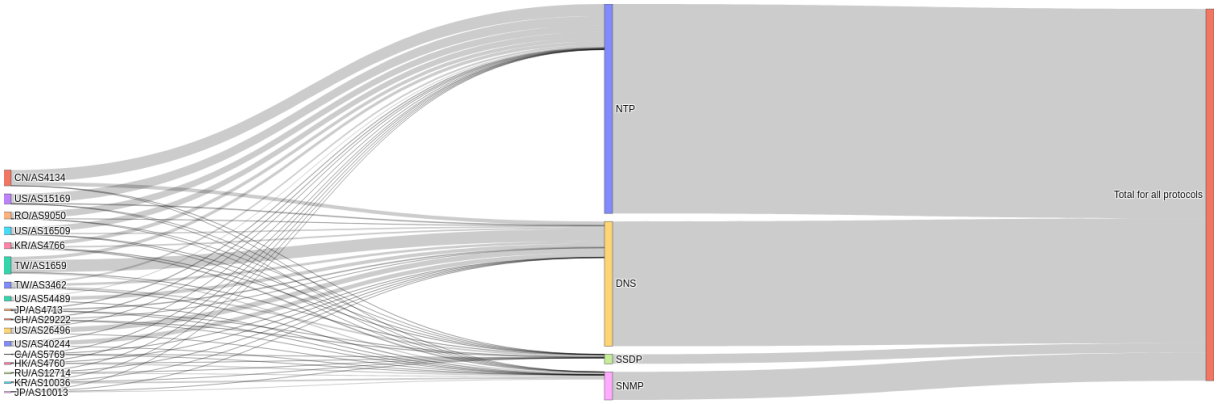

Figure 49: Top 5 ASNs by theoretical potential in May 2020

Many of the theoretical top 5 ASNs do also show up in the measured top 5. Producing a graph of more than the top 5 quickly becomes unintelligible. However, similar results are found near the top of the rankings for any top N, thus justifying theoretical estimation as a cost-effective method that produces similar results.

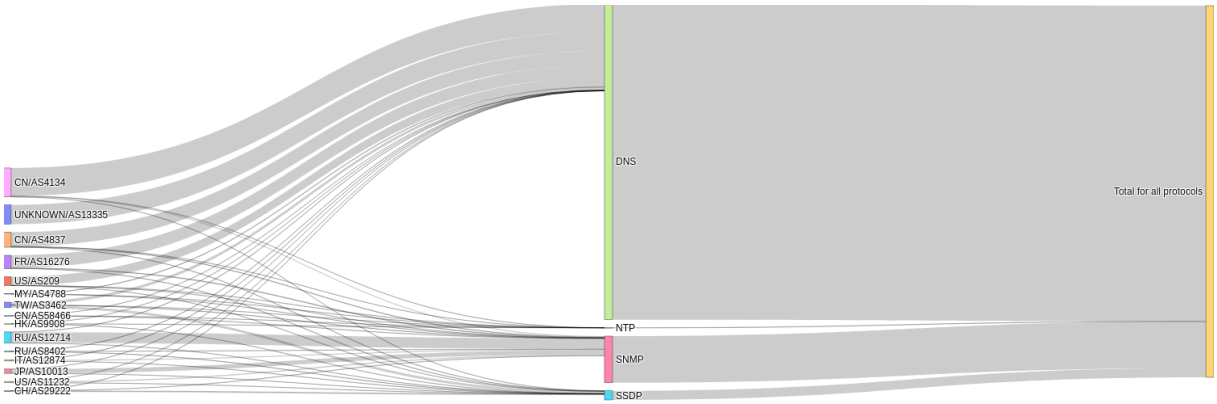

Figure 50: Top 5 ASNs by measured capacity in May 2020

That same narrative bears out when the measured capacity is reviewed instead of the theoretical, though sometimes the ASN in the top N changes. This makes sense since the theoretical estimation method relied on both CyberGreen and MLab data, and mostly used averages or percentiles to come to their conclusion. By empirically scanning and measuring some of the quantitative biases inherent in the estimative approaches can be avoided.

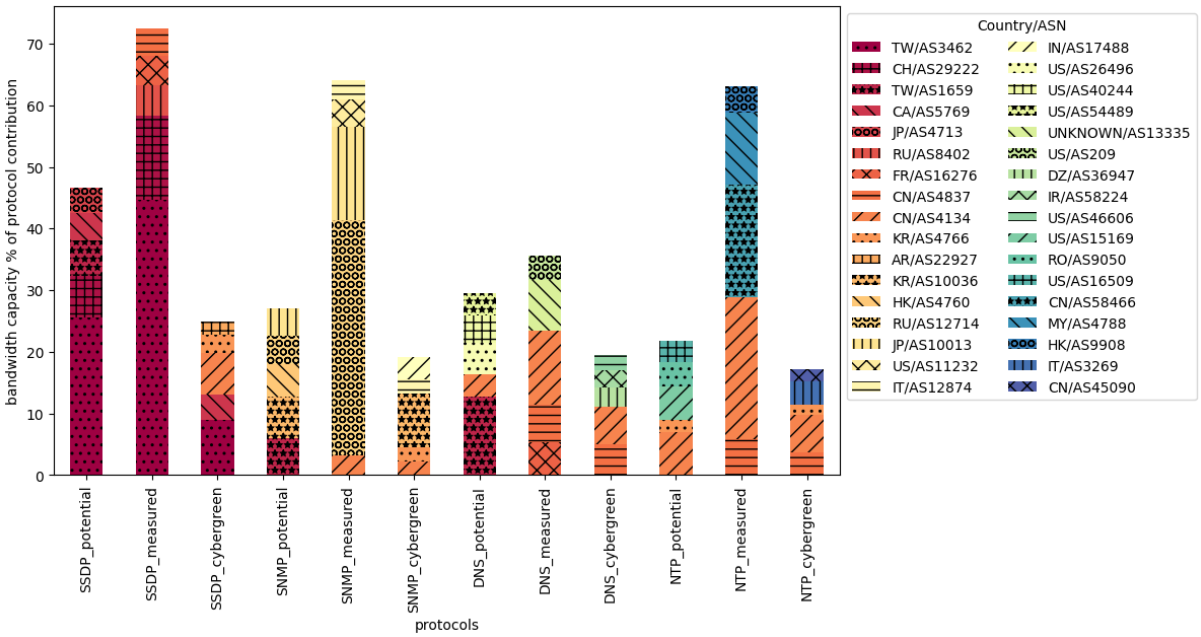

Figure 51: Top DDoS capacity contributing ASNs in May 2020 according to 3 measures

Figure 51 visualizes the difference between DRDoS severity measurement strategies. If the focus is on the reflector count reductions (as represented by CyberGreen data), or only theoretical estimation methods are used (as represented [4]), many opportunities for cost-efficient harm mitigation could be missed. The estimation and measurement results suggest it is possible to achieve a 30-70 percent reduction in reflective DDoS harm for any protocol by working with only 5 ASNs on remediation strategies. As if that isn't enough of a targeted intervention opportunity, the same ASNs are often heavy contributors to other protocols' pollution as well, so it is not needed to target 20 ASNs to achieve great impact across the four major reflective protocols. Those policy interventions could occur at the international level (diplomatic discussions), national level (regulatory requirements), ASN level (rate limiting, reflector exclusions, BCP38), or device manufacturer level (secure by design, default configuration) to affect change in the handful of IoT device manufacturers that harm everyone [177] and perhaps IoT firmware liability could be considered as an effective mechanism beyond the ASN [178]. The really key thing is that it prioritizes which ASNs should really be targeted in one’s sphere of influence: The ones with the most bandwidth.

CyberGreen reflector node count data over time is presented in figure 52. Certainly, great reductions have been achieved over the twelve months in reflector counts. The variance in NTP though seems interesting. Either this suggests complications in scanning methodologies or network instability or perhaps massive variations in NTP deployment.

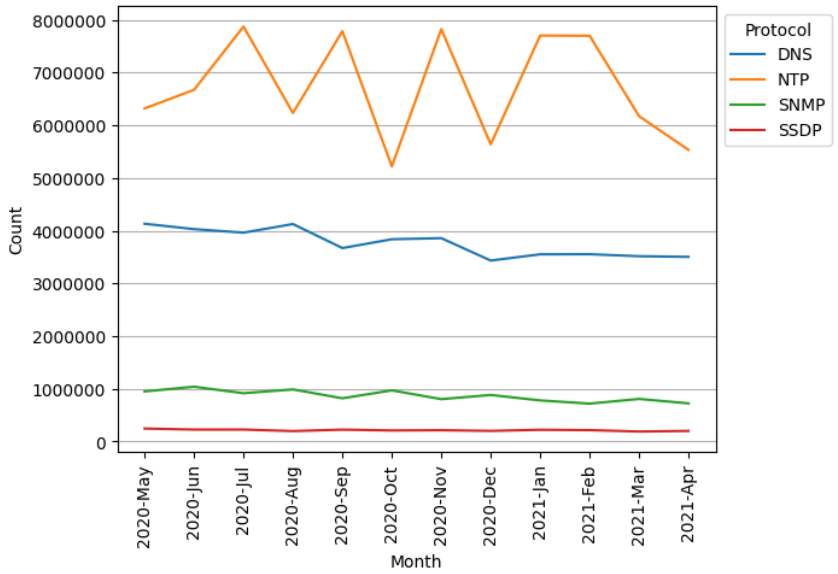

Figure 52: CyberGreen node count

If the CyberGreen harm mitigation thesis is correct, that reducing reflector count reduces potential, then such reductions should be seen in the measured capacity in figure 53. Since no similar reductions can be observed, it can be concluded that the wrong reflectors are being targeted for interventions, and that bandwidth of the ASN or reflector really is the greatest contributing factor to large DDoS attacks. Theoretical NTP potential greatly overestimates the impact (it uses CyberGreen count data), compared to the NTP measured contribution that is 3 orders of magnitude lower! Additionally, SSDP reflector counts are static in figure 52 and yet the measured potential is rising slightly in figure 53.

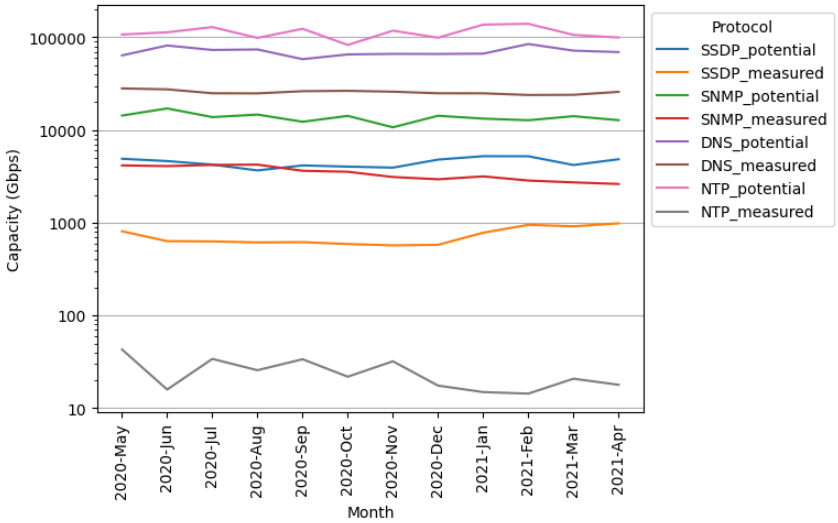

Figure 53: Theoretical potential vs. measured bandwidth capacity

Focusing on reflector count suggests interventions at many locations all around the globe are equal in priority, with relatively linear results. The author with co-authors articulates that one would get very non-linear effects (order of magnitude improvements) by focusing on bandwidth instead of just large reflector counts [6]. This is a fantastic opportunity in a policy sense given the development of carpet bombing and multi-protocol attacks [179].

5.4.2. Comparing measured and theoretical capacity

Theoretical capacity is faster and easier while still being useful if one can not scan for any reason. Empirical measurement methods are more accurate but also more time and resource consuming. The two are complementary though, because the differential can expose if mitigations or remediations are in place, or if network errors are at play, particularly when used over time. Through the exploration [6] it was determined that both have their place in organizations seeking to explore DDoS issues. Albeit the measuring has a huge disadvantage of not being able to be applied before particular protocol implementation and the first conducted measurement (post-processing and calculation can still be changed for the old measurement data), while the theoretical estimates can “look back in time” as far as any adapted data sources permit and adjust any constraints for the capacity calculations.

Conducting these estimates and measurements for ASNs rather than countries is the way forward. This is because the variance in bandwidth at the AS level is often lower, but also because it attributes the organization where policy intervention might be most impactful. That in turn also removes some of the diplomatic argument that this intervention is just a tool of foreign policy, and thus are all nations considered equally responsible to focus on top high bandwidth ASNs.

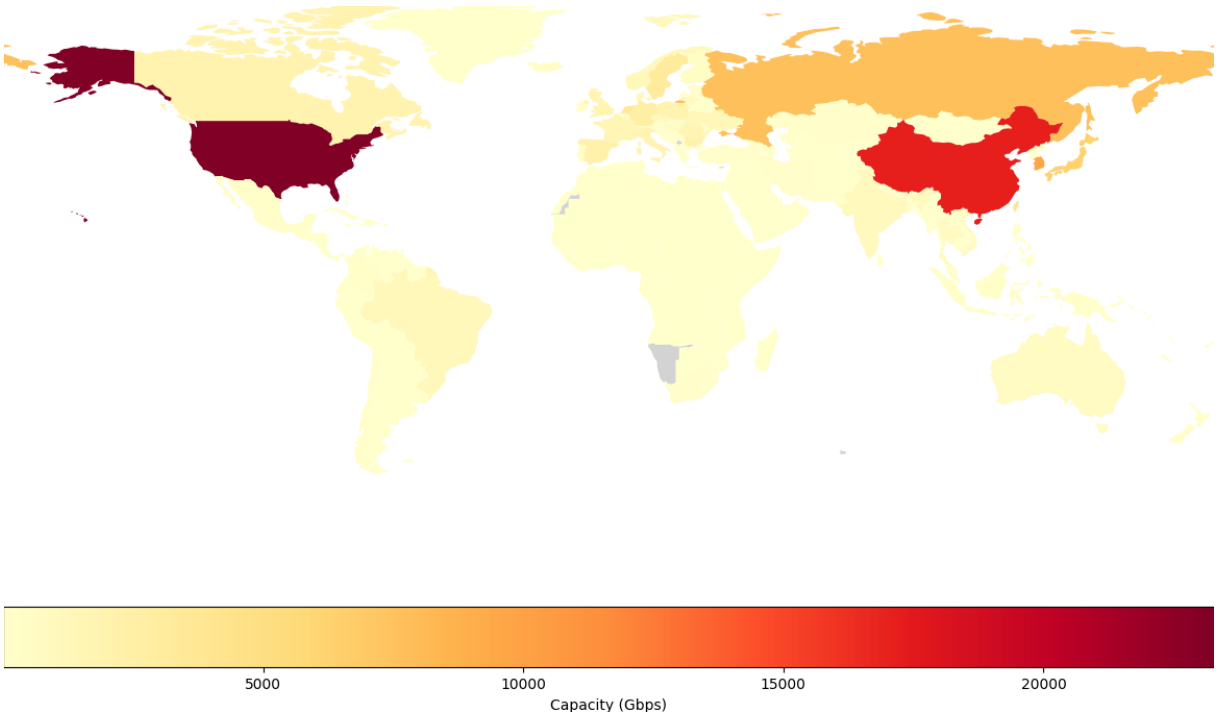

Figure 54: Theoretical capacity for DNS, NTP, SSDP, SNMP protocols in May 2020

Geographic capacity distribution is provided for the theoretical capacity in figure 54, and the measured capacity in figure 55. It demonstrates these two comparative understandings – measured vs. theoretical, ASN vs. Country. In all visualizations (including non-geographic ones) measured values are more pronounced and extreme. The top 3 countries are the same but the first 2 places are switched, why? The first possibility could be a scanning (measurement) point location in the USA for the theoretical estimate, and Europe for the measured capacity. This reinforces the inconvenient truth about the Internet measurement –

results differ based on where and when the scan is conducted. Other discrepancies could be attributed to differences in the bandwidth calculation and rate limit detection.

These summed capacity presentations are always skewed by some protocols having disproportionately large contributions, especially the measured capacity as some remediation and mitigation have been incorporated in the results, e.g., DNS overshadowing all others.

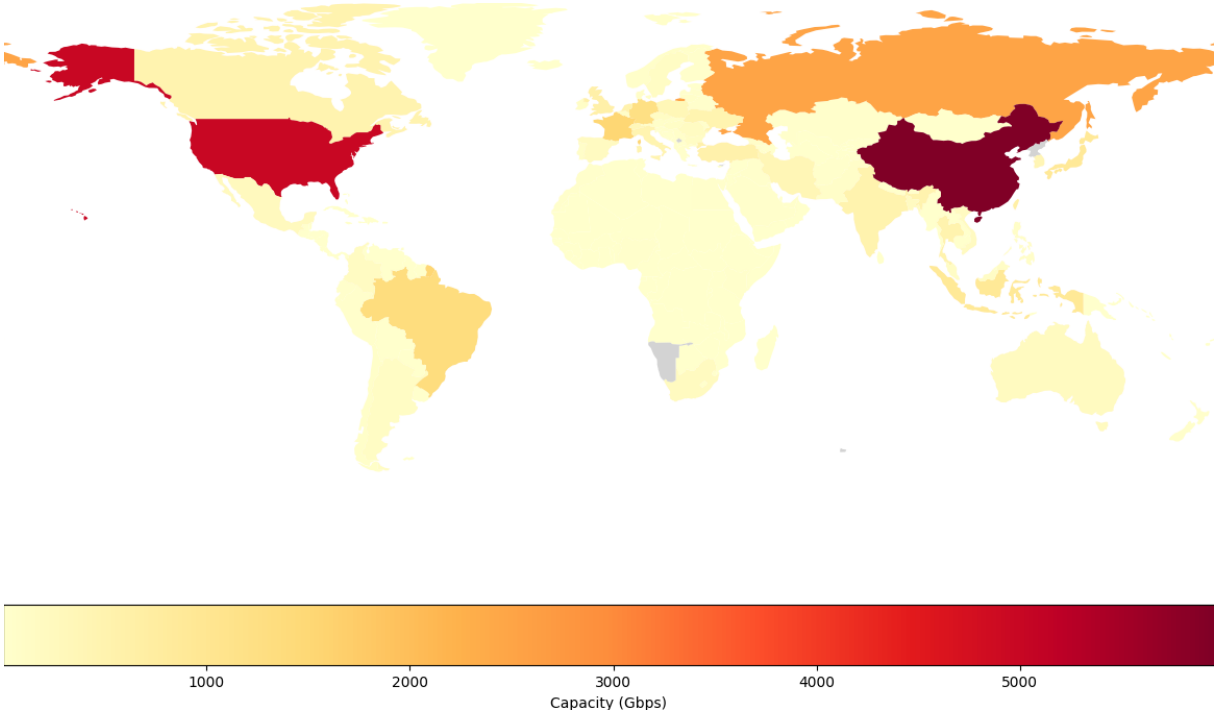

Figure 55: Measured capacity for DNS, NTP, SSDP, SNMP protocols in May 2020

Although these are great representations of summed bandwidth capacity across all protocols, there is limited utility for any intervention and individual protocol considerations raise additional research questions. Taiwan has the highest both measured and theoretical capacity globally for the SSDP protocol. South Korea has the highest theoretical but a low measured capacity for the SNMP protocol. It doesn't mean these countries have the worst reflector problem but quite the opposite can be true if rate limiting, past remediation, and present mitigation are taken into account.

All protocol visualizations demonstrate that these are average countries in absolute and relative terms. It might indicate that some ISPs deploy CPE with a default configuration running these abusable protocols that is also unique for these countries. In the SSDP case, these protocols might not have a rate limiting or have a higher one than the measurement uses. SNMP is a more interesting case as the theoretical potential indicates that there is a significant number of abusable devices but measurement indicates that these devices are either rate limited or bandwidth limited and therefore less contributing to the real global attack capacity.

This is merely one example, to illustrate that both country, ASN, and even protocol-specific considerations all contribute in unique ways to these two metrics. One must be careful to communicate the implications of those factors on the research. Plenty of opportunity for future research examining the interplay of those factors on metrics and these issues.

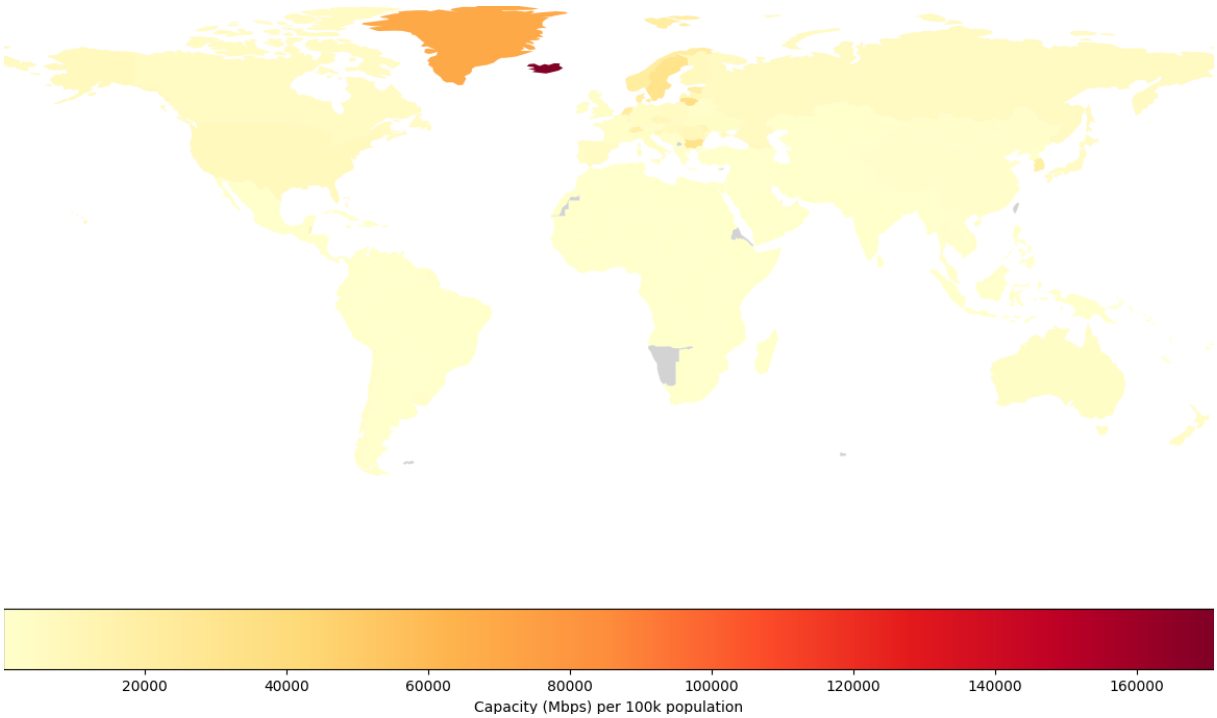

Figure 56: Theoretical capacity relative to population in May 2020

To illustrate this interplay further four geographic representations were produced which normalize relative to the human population in figures 56 and 57 and IP assignment in figures 58 and 59.

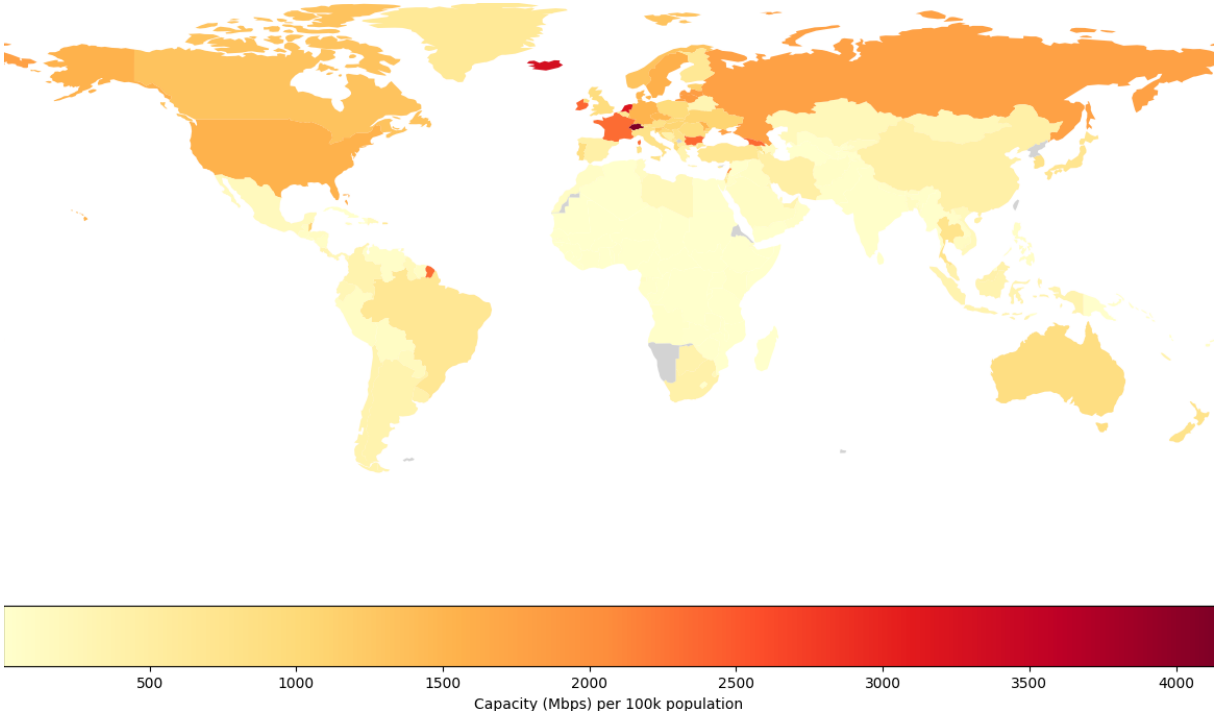

Figure 57: Measured capacity relative to population in May 2020

Internet propagation differs vastly across the globe, the IP address allocation is disproportional. The focus here is on IPv4 ranges which are mostly allocated to countries with early Internet adoption. The per capita representation is not ideal as a visualization, though it is still important. Even with IP address allocation the Internet availability varies vastly within countries. As the concern is only with publicly reachable devices, developing countries with

limited IPv4 address allocations might employ NAT or similar solutions. In the case of internally spoofed traffic or IPv6 direct assignments, this capacity might increase but that is currently indeterminable for technical reasons.

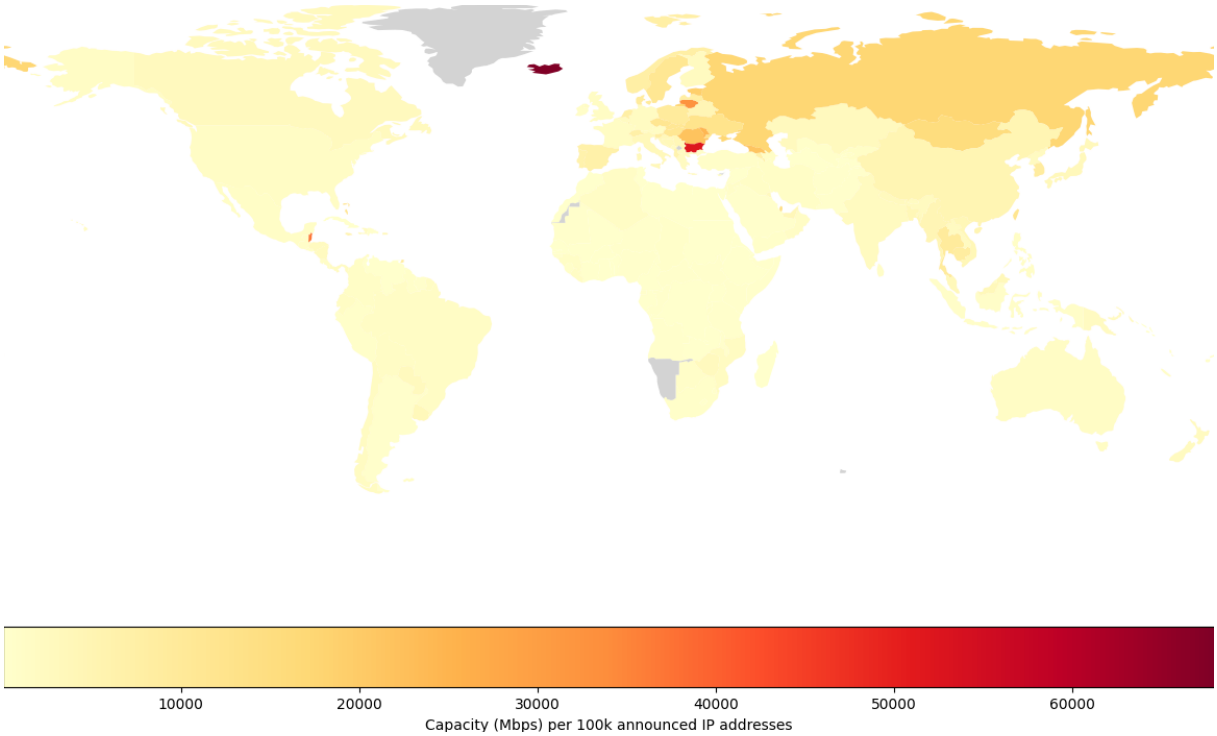

Figure 58: Theoretical capacity relative to announced IP addresses in May 2020

While measured capacity relative to population presents more outliers than the theoretical estimates, the opposite is true for the capacity presentation relative to announced IP addresses. Iceland is noteworthy in all four relative presentations. What can be extracted from these four relative rankings by being diplomatic and unbiased, without knowing the country or even hemisphere it is in? Per capita theoretical contribution is almost proportional to reflector count simply indicating globally disproportional reflector count – the largest in the world. Per capita measured contribution still being high (but not the highest) on its own can indicate various (even contradicting) possibilities – largest (monopolistic) ISP in the country deploying unsecured CPE, popular devices without RRL being common, data centers with large reflector count, high residential Internet connection speeds.

Per announced IP addresses theoretical capacity is still the globally highest ranking – it indicates either Internet connection speeds (primarily upload) being high for the reflectors or having anomalously large reflector count which can't persist in today's Internet without being abused for the DDoS attacks and not causing issues for the reflector hosting networks themselves requiring remediation or mitigation. A middle ground can be true – a large number of reflectors having high-speed Internet connections. Per announced IP addresses measured capacity ranking average across the globe means remediation and possibly mitigation are in place, there is RRL likely commonly present and major networks might have mitigation activated by attack packets. Globally this country has better network management practices and potentially legal policies that address the DDoS attack capacity than most of the other countries, having one of the most effective remediation and mitigation relative to its population. Is Iceland any of that? Yes, most if not all! The small population might skew

rankings more easily even for relative views, but Iceland has both high-speed residential Internet and large data centers, it is technologically, societally, and legally developed. It might indicate that these reflectors are services running in data centers on high-power servers but actually do not contribute as much capacity because of the remediation and mitigation. But neither this nor most other cases can be explained with any more depth at the country level of data resolution.

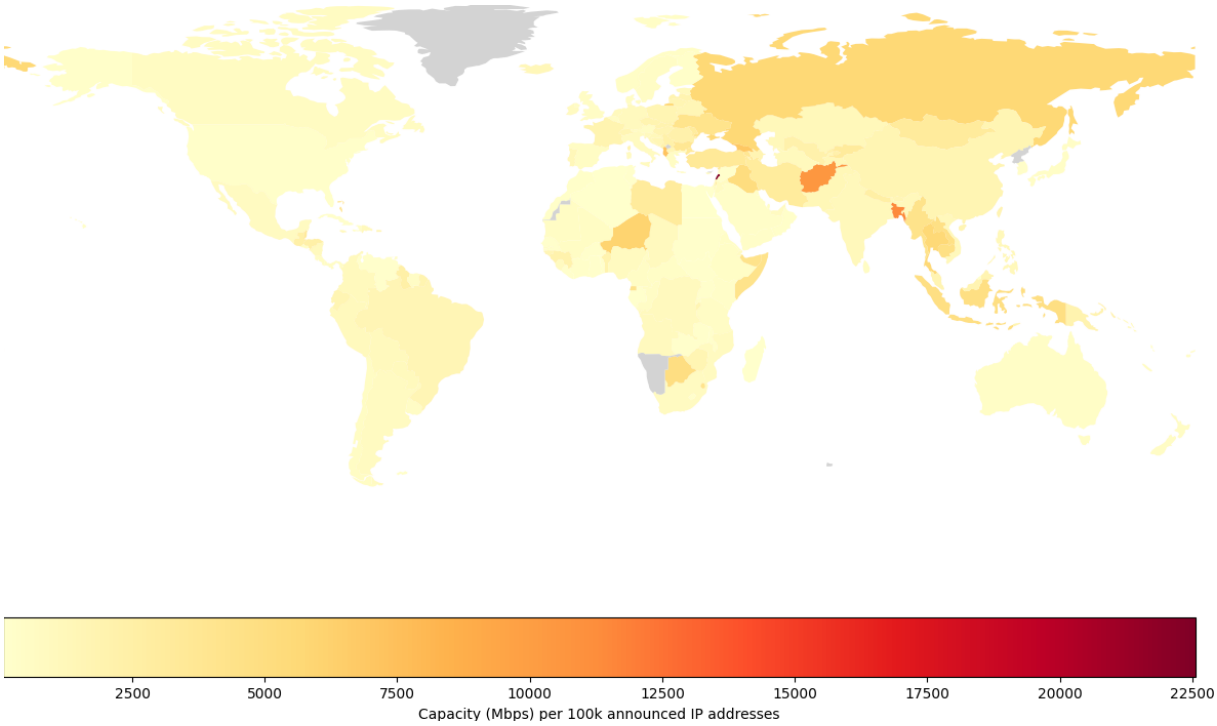

Figure 59: Measured capacity relative to announced IP addresses in May 2020

The intention of all these visualizations is to demonstrate that ASN level discussions are far more accurate quantitatively, and also neutralize some of the diplomatic discussions. Without pointing out any papers, for example, it is common to see quantitative research of reflector, server, botnet, compromised, etc. node counts with (specific countries in) Asia being justifiable in the first place disregarding relative metrics (provided visualizations) and actual contribution (measurement methodology). Thus regardless of large reflector counts the global south does not contribute nearly as much (disproportionally) to DDoS capacity as some reports and research lead to believe.

6. REMEDIATING DDOS ATTACKS

Full technical remediation of either IP spoofing or the presence of reflectors solves the DRDoS issue completely. Significant remediation of both of these root causes would substantially decrease the attack and required mitigation capacity but the growing network bandwidth for both end-user and global transit negates the remediation efforts. But if no remediation efforts had occurred there would be individual and overall DRDoS attack capacity growth proportional to the network capacity growth. This chapter discusses some additional remediation possibilities of the DRDoS attacks and related concerns in the context of this research which have been published in [10].

6.1. Actors and motivation

Understanding some of the actors on the Internet landscape can provide clues as to why the world is struggling with remediation efforts. Ordinary users merely want to access remote services offered by organizations. Malicious actors range widely in their capabilities and motivations but ultimately seek to prevent users from accessing targeted services. Most of the published research focuses on these three types of actors, but there are others who either contribute to the problem or can contribute to the remediation.

6.1.1. ISPs and transit service providers

Many of the transit service providers and some of the ISPs and data centers do not have the capability to filter out large application-layer DDoS attacks. The motivation for these types of actors is to provide network services to all of their clients while maintaining customer satisfaction and fulfilling service-level agreements. If the attack size is not affecting other clients, it might get passed on to the victim. The victim may or may not have the means to deal with the specific attack. If the attack is large enough to affect other clients, then the transit providers have to mitigate it and often the only means to do that is to employ blackholing as remote from the victim in network topology as possible [48]. As the attacked service loses Internet connectivity, the attack can be considered successful.

ISPs are externalizing the cost of having open reflectors on their networks. From their perspective, there might be no drawbacks even on the network bandwidth consumption. Networks that focus on specific customer segments, e.g., residential vs. data centers usually have unbalanced network bandwidth consumption – unused capacity of upload or download respectively. When reflectors in a residential ISP network are generating amplified responses, they consume this unused upload bandwidth capacity. As long as this consumption is relatively small and does not affect other clients or the network routers, there are no ill effects for the ISP and therefore no motivation to address the issue.

Often the target of the DDoS attacks are commercial services hosted in data centers then this unbalance enables to receive attack download bandwidth without any additional expenses to the extent of reserve capacity. If the free capacity is sufficient and the network has an appropriate filtering solution, then the attack can be filtered. The larger the data center is the more free capacity its network might have meaning larger attacks could be filtered out, some

of the largest data centers in the world do offer DDoS filtering at little or no cost and are able to handle most of the attacks. Smaller data centers and ISPs can be overwhelmed by a single attack.

If an ISP has balanced bandwidth by either offering services to both bandwidth-generating and consuming clients or by selling unused capacity as a transit, it can have financial incentives and thus motivation to keep wasted bandwidth to a minimum. Technical solutions or a network monitoring and management practice can significantly reduce wasted bandwidth for the ISP.

No legislation specifically targets the negligence of having public reflectors present on the networks. And even if a DDoS attack from a specific set of reflectors caused provable damages, the liability could be shifted to the end client hosting these reflectors. Overall, a significant number of ISPs that host public reflectors have little motivation to address this issue.

6.1.2. Mitigation service providers

DDoS mitigation services often receive the largest attacks on the Internet. These service providers might exclusively offer DDoS filtering or accompany it with CDN or other network services. The business model is straightforward, acquire ingress bandwidth capacity that exceeds the largest expected attacks and deploy a filtering solution that can drop the attack traffic while forwarding the legitimate packets to the client.

Whenever a new protocol is abused, or a new attack size record is broken, these providers publish a technical report. These reports are the most referenced sources for the DDoS capacity and largest attacks in academia, industry, and media, making them excellent sources of free worldwide marketing.

As long as expected attack capacity and future growth are manageable, and no record-breaking attacks happen, these providers are in a safe market position. These providers have the most technical insights of what is the current state of affairs and what needs to be addressed first. But remediating the current state of affairs are negative incentives to these providers as significantly remediating the issue can lead to lost competitive advantage or even loss of a business model.

6.1.3. Device manufacturers

Often overlooked is the fact that a large number of (most) reflectors are not needed public services but are simply CPE, residential, and business devices with default configurations connected to the Internet. The problem is significantly exacerbated by the class of devices that have router functionality with separate internal and external network interfaces. Services on the internal interface might be required, but they are not contributing to the public reflector problem while services on the external interface could contain reflectors and generally are not needed for functionality required by the actual internal users.

While residential device manufacturers often engage in a race to the bottom price-wise, it is achieved by cutting corners, software quality and security have been first to suffer. If a device is compromised because the default configuration exposed the control panel to the Internet and default credentials or a vulnerability in the software were exploitable, then it may at the very least become part of the botnet. Users of these devices might not even notice

negative effects, or they might wonder why the CAPTCHAs have become more prevalent or why the Internet sometimes slows down. In more extreme cases user's information could be stolen, or even further exploitation conducted to take over other devices in the network. If this is the case with the mass exploitation of specific manufacturer's devices with tangible consequences for the users, it might turn into a negative press. The manufacturers that are valuing their brand are incentivized to minimize such occurrences and fix these issues, so they do not reoccur in the future. Same time a large number of devices operating as open reflectors do not harm direct customers, lacking the bad press or other reasons which would motivate the manufacturer to address the issue.

6.1.4. Policymakers and legislature

In all developed countries, causing a DDoS attack already falls under some criminal act. Malicious actors responsible for reflected DDoS attacks are the hardest to identify. The global nature of the Internet and DDoS could mean a single attack against a company registered in one jurisdiction could be affecting services physically hosted in one or more other jurisdictions which is caused by an attacker located in another jurisdiction who uses spoofing capacity and reflectors located in any number of other jurisdictions. While charging criminals and affecting international law is problematic, the legislatures and regulators have the devotion to improving overall human life and should be employed on the mission to remediate the DDoS attacks.

6.2. Remediating DDoS attacks

What kinds of remediation efforts are being conducted already? The most visible one targeting the source networks is scanning for open reflectors and notifying network administrators. What could be other simple solutions applicable to the previously discussed actors?

6.2.1. Notifying network administrators

Various academic and industry research focuses on conducting scans for knowingly abused services reachable on the Internet and notifying network or abuse contacts. If the network is properly managed these notices are forwarded to the end client and maybe the client is even assisted with solving the issue. Perhaps the network has specific terms of service that mandate clients to limit or block its reflectors that may or may not be enforced. Poorly managed networks do not even forward these notices. While running various reflector honeypots, the author has encountered a large number of forwarded notices on some of the well-managed networks. But the question stands, what the effectiveness of sending such notices is?

The quantity of reflectors for long-term abused protocols seems to be decreasing over time in most cases. The question is, to what extent the notification efforts play a role in this decrease? No research investigates this, and thus no reliable assertion can be made. An alternative argument can be made that devices that are abusable are present on the Internet until the end of their lifetime or until network configuration changes. This could be

completely detached from any remediation efforts, but no investigation into this argument has been conducted either.

The repetitive nature of the notification e-mails combined with the lack of any perceived value for the networks makes its effectiveness questionable. Providing an estimate of potentially wasted bandwidth capacity calculated for each network can offer at least some perceived value. Measuring the impact of this remediation approach is tricky but a detailed report hidden by tracked link and capacity changes in the specific network over time against the baseline could provide insights into its effectiveness. Same time network administrators are less likely to click links than perceive self-contained detailed information in the e-mail.

6.2.2. ISPs and net neutrality

Although net neutrality has been a hot topic in the last few years, there are widespread precedents of it being violated towards some protocols for the benefit of ISPs. While DPI and traffic shaping in residential and mobile networks have been widely investigated, discussed, and criticized, the lesser-known and measured practice of ISPs and data centers blocking or limiting specific ports are not disparaged. Most commonly it targets e-mail sending ports and either block those by default with an opt-out ability but a rate limiting or filtering system could be employed instead. Why client perception differs between these two cases is open for debate, it could be that an opt-out feature is viewed as a sufficient option or motivation is taken into account.

Why are the mail sending ports so special? Spam has been a problem even before reflected DDoS attacks became a norm. As it directly affects the productivity and safety of the users and businesses, various mitigation approaches have been developed, primarily spam filtering and blacklisting compromised spam sending hosts. While spam filtering is not 100% precise so is the blocking of individual IP addresses of compromised hosts as well. If the network administrator is not taking expedited action against a spamming host, it can be assumed that the same will happen with other spam-sending hosts on the same network. Thus blacklisting the whole network or decreasing its reputation seems reasonable to protect the users. As individual blacklist management is time-consuming global blacklists utilized by many mail hosters have become commonplace.

If ISPs wish to offer clients the ability to directly send e-mails that are not rejected or classified as spam by most receivers, they have to actively keep themselves outside the blacklist. Whenever an abusive host appears on the network, swift action has to be taken to stop spamming by limiting network connectivity or requesting the client to solve the issue. Otherwise, clients can not send e-mails, and thus ISP can only have clients that do not need the functionality or accept this drawback. Most of the ISPs elect to deal with the spam issue to avoid being added to the blacklists.

As discussed before, there are ISPs that externalize the costs of poor network management practices and have no incentives to act otherwise. The question is, can they be forced to improve the management of their networks? Can the blacklist approach that is being used for fighting spam be replicated for the DDoS?

What would be the cost for ISPs of being added to the blacklist? There are already other blacklists besides the spam ones. Usually, they consist of individual hosts (or small subnets) whose behavior is deemed malicious, e.g., spreading malware, aggressive scanning, probing

services, and brute-forcing credentials. These blacklists are often deployed by governmental or other organizations striving for more security. Surprisingly this might not penalize the ISP at all as non-abusive clients might not experience any limitations. Combined with the fact that reflectors are not malicious and blacklisting those is pointless, this approach is unsuitable as a mitigation strategy.

Previously discussed offenses could decrease the total reputation of the network. Greylisting the whole ISPs based on this reputation can be effective as it affects many clients. A few hosts brute-forcing credentials might force websites that rely on the network reputation to require all the ISP clients to always solve a CAPTCHA. Credit card transactions or other actions might be delayed for manual processing or fail by default as labeled potentially fraudulent. Client satisfaction can drop, and they could be looking for other ISP which does provide a financial incentive to meet some reputational baseline. Relying on the same greylists could be done only for networks that allow spoofed IP packets and only if it could be proven that the spoofing is actively happening.

The industry might need a new type of greylist for DDoS reputation and a way how to penalize ISPs. The penalty should be relevant to the DDoS issue, e.g., the same CAPTCHA that many DDoS mitigation services employ could penalize the whole networks known to allow spoofed packets or contains a disproportionately large number of open reflectors.

Whenever there is a discussion requiring third parties to implement a new regulation, there is always a question of bearing the costs. It happens with ISPs whenever national governments require expensive DPI or data retention systems. Basic solutions for both blocking packets being addressed to reflectors and IP address spoofing are simple and cheap. Blocking all packets incoming from the Internet addressed to a few known abused ports for an ISP is trivial and cost-free on existing equipment. The only incurred cost is managing or providing self-service functionality for clients to opt out.

This is already happening to an extent, for example, one of the largest ISPs in the USA Charter Communications already enforces default port blocking for their residential and business Internet clients [180]. This ISP is not even concealing the reasons for blocking, from the 16 blocked ports 4 are stated to be “routinely exploited” for the DDoS. These blocked protocols are QOTD, CharGEN, SSDP, and Routing information protocol plus NetBIOS which could be also abused for DDoS attacks. SSDP is the only one of these protocols that have been measured in this thesis, indicating that this might be a rather simple system where exclusions are rare or unsupported and no advanced packet inspection is possible, leaving other more abused but essential protocols like DNS unaffected.

This approach remediates targeted protocol reflectors which does decrease the global attack capacity. Every client opt-out has a computational cost for IP packet routing which can be significant on the network edge, every common protocol being blocked without an opt-out feature has a financial cost for providing support to clients, satisfaction loss from clients, and maybe even the ultimate cost of losing clients. This is one the simplest way how to remediate the global DDoS attack capacity but how to balance everyone’s interests? One of the ways would be for the ISPs to deploy CPEs (primarily network routers) with commonly abused ports blocked by default with the option of manual not automated port opening on the client’s device, not the ISP network which alleviates some of the discussed drawbacks. But if there is

no ISP issued default CPE modem or router then the responsibility transfers to [6.2.3. Devices and regulation].

Some mitigations outside of the victim's network do exist. They are rarely publicized or discussed, in [4.3. DNS] simplistic mitigation has been detected in a transit network that activates when an undisclosed threshold has been reached. An unclear mitigation or remediation has made [4.7. Memcached] unmeasurable. Data in this thesis suggest that there are mitigation and remediation solutions that are applied to the highest abused protocols even in transit. But those are likely computationally costly and the proliferation of those across transit providers is unknown. Can those be applied to all abused protocols across the globe? Can this be enforced through legislation? Are there any hidden motivation conflicts? This avenue is veiled and the author can't answer any of these questions.

6.2.3. Devices and regulation

Publicly reachable reflectors running on consumer devices is something that can be started being addressed already now. California state legislature has passed a bill [181] requiring connected devices to have basic security measures to protect consumers. While this does not directly affect consumers, there are no reasons why this type of legislation could not improve the overall security of the Internet by requiring external interfaces not to provide any unneeded services by default. If this regulation in at least one large market is introduced rationally, then it would be more cost-effective for manufacturers of these consumer devices to supply the same secured version of the device to all markets.

While a legislature could, in the same way, require ISPs to offer basic firewalls with an opt-out feature, it would not have as large a snowball effect in other jurisdictions. As patching of the consumer devices is done rarely, the old ones might continue contributing reflector capacity until end-of-life without this ISP regulation.

To start addressing the issue from the device perspective, decision-makers need to understand which device classes and manufacturers contribute the most to the capacity. Then the most prominent manufacturers can be addressed directly, and this knowledge can be presented to legislatures to justify actions. Only if national legislation has proven to be effective, it is justifiable to advocate for international laws and regulations.

A report on a research project for the European Commission by Leverett et al. [178] discusses general device issues that translate well to this thesis and DDoS attacks. "To improve a system, we have to be able to measure it" [178] is the main goal of the thesis which combined "governments have a duty to collect decent statistics, so that all can understand what's happening and those stakeholders with the ability to fix things can do so" [178] is the only feasible way how to sufficiently address the global DDoS attack capacity within any reasonable time frame. The introduction of liability for software vendors was proposed as well which can transform into responsibility to provide security patches (preferably automated) for years, e.g., disabling all listening ports on the external interfaces of routers and other CPE. The final proposal of establishing a European Union level technical agency to provide "a shared resource for policymakers and regulators" [178] can bring all these points together to properly inform and guide the European Union level process to remediate the DDoS attack capacity which can develop into global improvements and attack capacity reduction.

RESULTS

Research presented in this thesis has resulted in a methodology for measuring individual protocol DDoS attack capacity, the first of its kind in academic and industry public sources. The closest existing sources provide only estimates not based on measuring individual devices. The proposed methodology was implemented and tested in this thesis for some of the currently and historically most abused protocols. The measurement implementation is reusable by design for a continuous monitoring system. Designed measurement modules are completely independent of specific protocols.

For this thesis it was established that devices significantly contributing to the DDoS attack capacity have a minimum response rate of 80% which is used for all the attack capacity calculations: NTP – **43 Gbps**, DNS – **27.5 Tbps**, SSDP – **808 Gbps**, SNMP – **2.47 Tbps**, CLDAP – **870 Gbps**. For these 5 protocols, the non-overlapping global DDoS attack capacity was calculated to be **31.33 Tbps**.

Different data views and visualizations have been explored to analyze these individual protocols – rate limit detection, amplification distribution, attack capacity geographic distribution, response round trip time, individual device speed distribution, etc.

Although countries with the largest attack capacity contributions are always discussed it was established throughout the whole thesis that individual networks (AS) are the primary culprits and remediation discussion should target them not the individual countries or regions. It was established that even more in-depth analysis is needed for the sets of devices behaving uniformly and are present only in particular networks or across many. The device classification topic has been explored and classifiers were developed but the overall device classification research field has not yet progressed sufficiently to apply it to the reflector classification.

Implementation limitations and drawbacks were explored, and challenges with data quality were discussed. Protocols that have low reflector counts can’t be measured for the first time reliably if significant mitigation or remediation exists, e.g., Memcached protocol was measured to have almost no attack capacity although more likely it was mitigated on a network level.

To compare measurement results the only existing alternative the theoretical estimation methodology was adapted for the relevant measurement execution time frame. Although early on the author claimed that the theoretical estimation is an unreliable methodology, it was determined that both methodologies have advantages and disadvantages but both can supplement the understanding of the DDoS attack capacity. The theoretical methodology is easier, cheaper, and faster, it can also be applied to old data sets to “look back in time”. While the measurement methodology calculations can be applied only after the first measurement of the targeted protocol, then a more precise capacity is produced that also reveals if there are any remediation or mitigation that was not addressed by the theoretical estimate.

Relative capacity contributions have been reviewed which is an uncommon approach but it revealed that many claims highlighting “problematic” regions (e.g., Asia) can be misleading when (measured) remediation and population or network size are taken into account.

CONCLUSIONS

When this research topic was proposed and the scope defined, the outcome seemed clear to the author – individual attack capacities would be precisely measured, all these capacities totaled in accordance with the individual network boundaries and limitations, produced results being relied on by the decision makers on all levels for the remediation efforts to address the long-term reflected DDoS issue. Although the measurement methodology was defined and implemented, individual protocols were measured and the total attack capacity calculated thus achieving the desired results many unanticipated issues were identified and many new research questions were raised.

The produced results can be considered the best existing empirical estimates rather than precise measurements. Although the capacity is the main output of the measurement it is affected by many permanent (e.g., location, ISP, transit provider) and transient (e.g., the Internet load, day of the week, defenses engaging) factors discussed throughout this thesis, the additional extracted information for instance rate limiting and amplification distribution can uncover existing remediation or mitigation, which can be continuously monitored and be a much more valuable improvement metric.

Device classification through this research has emerged as the next level of addressing liability and responsibility for the reflector attack capacity contribution rather than a mere presence on the Internet. The author expected the device classification field to progress faster than it did. A significant portion of the published device classification research was not reproducible. Although this research contributed to the device classification field for the reflector classification a more complex multi-protocol feature extraction is required.

While conducting the experimental measurements various behaviors that can be considered remediation or temporary mitigation were identified for some protocols on transit networks. These defenses are not publicized and are not necessarily observable by malicious actors but rather are directed toward victims. Some of these fully remediate a protocol's attack capacity – Memcached from the measurement system vantage point is fully remediated. How prevalent and efficient is this mitigation or remediation? What prevents all the other abused protocols to be remediated this way? The author has discussed some of the possible technical reasoning but there are a lot of unknowns that also can't be investigated from the established measuring location alone. The author has discussed and speculated about possible motivating factors that might have a major concealed impact resulting in insufficient global remediation. Maybe major ISPs or transit providers have a competitive advantage by having more reserve capacity available to mitigate the DDoS attacks? Maybe DDoS attack mitigators as major actors don't want to fully remediate the DDoS capacity to preserve their business model? More than on one occasion in an academic setting the author was accused of spreading unsubstantiated or conspiratorial opinion just for raising such questions.

The implemented methodology and produced measurement data permit multiple future research directions. The most promising ones are – changes happening to the protocols over the longer term besides the capacity (for the implemented protocols data are already acquired), capacity by devices (when suitable classifiers are developed), network owner notification of "wasted" bandwidth, and measuring resulting remediation.

ACKNOWLEDGMENTS

I thank my scientific advisor professor Dr. Guntis Bārzdiņš for his guidance.

I greatly appreciate all of my co-authors – Éireann Leverett, Roman Graf, Jesús Rubio, Gábor Visky, Kimmo Heinäaro, Kārlis Podiņš, Aaron Kaplan, Lukas Bortnik, Dan Heering, Kimberly Tam, Olaf Maennel, Roland Meier, Luca Gambazzi, Vincent Lenders, Erwin Orye, for our fruitful collaborations. I have learned something from every single one of you.

Research topic “Untapped Malicious Potential: Calculating a maximum rDDoS metric” presented by Éireann Leverett at the *Cyberchess 2017* conference has been the primary inspiration for me to focus on the DDoS attack capacity in this thesis, for which I am particularly grateful.

The research presented in this thesis has been partially supported by the NATO Cooperative Cyber Defence Centre of Excellence research projects conducted in 2017-2021 while the author was employed there as a researcher.

This research and doctoral thesis development has been supported by the European Social Fund project “Strengthening of the Capacity of Doctoral Studies at the University of Latvia within the Framework of the New Doctoral Model”, identification No. 8.2.2.0/20/I/006.

REFERENCES

[1] "History of DDoS Attacks," *Radware*, Mar. 13, 2017. https://security.radware.com/ddos-knowledge-center/ddos-chronicles/ddos-attacks-history/ (accessed May 04, 2017).

[2] G. Aceto, A. Botta, P. Marchetta, V. Persico, and A. Pescapé, "A comprehensive survey on internet outages," *Journal of Network and Computer Applications*, vol. 113, pp. 36–63, Jul. 2018, doi: 10.1016/j.jnca.2018.03.026.

[3] J. Pescatore, "DDoS attacks advancing and enduring: A SANS survey," 2014.

[4] E. Leverett and A. Kaplan, "Towards estimating the untapped potential: a global malicious DDoS mean capacity estimate," *Journal of Cyber Policy*, vol. 2, no. 2, pp. 195–208, May 2017, doi: 10.1080/23738871.2017.1362020.

[5] G. Visky, A. Lavrenovs, E. Orye, D. Heering, and K. Tam, "Multi-Purpose Cyber Environment for Maritime Sector," *iccws*, vol. 17, no. 1, pp. 349–357, Mar. 2022, doi: 10.34190/iccws.17.1.26.

[6] A. Lavrenovs, E. Leverett, and A. Kaplan, "The tragedy of common bandwidth: rDDoS," in *New Security Paradigms Workshop*, Virtual Event USA: ACM, Oct. 2021, pp. 43–58. doi: 10.1145/3498891.3500928.

[7] G. Visky, A. Lavrenovs, and O. Maennel, "Status Detector for Fuzzing-Based Vulnerability Mining of IEC 61850 Protocol," in *Proceedings of the European Conference on Information Warfare and Security*, Academic Conferences International Ltd, 2021. doi: 10.34190/EWS.21.007.

[8] A. Lavrenovs and R. Graf, "Explainable AI for Classifying Devices on the Internet," in *2021 13th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia: IEEE, May 2021, pp. 291–308. doi: 10.23919/CyCon51939.2021.9467804.

[9] R. Meier, A. Lavrenovs, K. Heinaaro, L. Gambazzi, and V. Lenders, "Towards an AI-powered Player in Cyber Defence Exercises," in *2021 13th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia: IEEE, May 2021, pp. 309–326. doi: 10.23919/CyCon51939.2021.9467801.

[10] A. Lavrenovs, "Towards Remediating DDoS Attacks," in *Proceedings of the International Conference on Cyber Warfare and Security*, TN, USA, 2021. doi: 10.34190/IWS.21.046.

[11] L. Bortnik and A. Lavrenovs, "Android Dumpsys Analysis to Indicate Driver Distraction," in *Digital Forensics and Cyber Crime. Proceedings of the 11th EAI International Conference, ICDF2C 2020, Boston, MA, USA, October 15-16, 2020.*, in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 351. Cham: Springer International Publishing, 2021, pp. 139–163. doi: 10.1007/978-3-030-68734-2_8.

[12] A. Lavrenovs, K. Heinäaro, and E. Orye, "Towards Cyber Sensing: Venturing Beyond Traditional Security Events," in *Proceedings of the 19th European Conference on Cyber Warfare*, Chester, UK: ACPI, Jun. 2020. doi: 10.34190/EWS.20.062.

[13] A. Lavrenovs, R. Graf, and K. Heinaaro, "Towards Classifying Devices on the Internet Using Artificial Intelligence," in *2020 12th International Conference on Cyber Conflict*

(CyCon), Estonia: IEEE, May 2020, pp. 309–325. doi: 10.23919/CyCon49761.2020.9131713.

[14] A. Lavrenovs and G. Visky, “Investigating HTTP response headers for the classification of devices on the Internet,” presented at the 2019 IEEE 7th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Liepaja, Latvia, Nov. 2019. doi: 10.1109/AIEEE48629.2019.8977115.

[15] A. Lavrenovs and G. Visky, “Exploring features of HTTP responses for the classification of devices on the Internet,” presented at the 2019 27th Telecommunications Forum (TELFOR), Belgrade, Serbia, Nov. 2019. doi: https://doi.org/10.1109/TELFOR48224.2019.8971100.

[16] A. Lavrenovs, “Towards Measuring Global DDoS Attack Capacity,” in *2019 11th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia: IEEE, May 2019, pp. 1–15. doi: 10.23919/CYCON.2019.8756851.

[17] K. Podins and A. Lavrenovs, “Security Implications of Using Third-Party Resources in the World Wide Web,” in *2018 IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE)*, Vilnius: IEEE, Nov. 2018, pp. 1–6. doi: 10.1109/AIEEE.2018.8592057.

[18] A. Lavrenovs and F. J. R. Melon, “HTTP security headers analysis of top one million websites,” in *10th International Conference on Cyber Conflict, CyCon 2018, Tallinn, Estonia, May 29 - June 1, 2018*, 2018, pp. 345–370. doi: 10.23919/CYCON.2018.8405025.

[19] A. Lavrenovs and K. Podins, “Privacy violations in Riga open data public transport system,” in *2016 IEEE 4th Workshop on Advances in Information, Electronic and Electrical Engineering, AIEEE 2016 - Proceedings*, Vilnius, Lithuania: IEEE, Nov. 2016, pp. 1–6. doi: 10.1109/AIEEE.2016.7821808.

[20] J. Mirkovic and P. Reiher, “A taxonomy of DDoS attack and DDoS defense mechanisms,” *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, p. 39, Apr. 2004, doi: 10.1145/997150.997156.

[21] 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*, vol. 15, no. 4, pp. 2046–2069, 2013, doi: 10.1109/SURV.2013.031413.00127.

[22] Imperva, “DDoS Attack Types & Mitigation Methods.” https://www.incapsula.com/ddos/ddos-attacks.html (accessed May 21, 2018).

[23] Y. Xie and S.-Z. Yu, “Monitoring the application-layer DDoS attacks for popular websites,” *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 1, pp. 15–25, 2009.

[24] M. Shtern, R. Sandel, M. Litoiu, C. Bachalo, and V. Theodorou, “Towards Mitigation of Low and Slow Application DDoS Attacks,” in *2014 IEEE International Conference on Cloud Engineering*, Mar. 2014, pp. 604–609. doi: 10.1109/IC2E.2014.38.

[25] E. Cambiaso, G. Papaleo, and M. Aiello, “Taxonomy of slow DoS attacks to web applications,” in *International Conference on Security in Computer Networks and Distributed Systems*, Springer, 2012, pp. 195–204.

[26] H. Wang, Z. Xi, F. Li, and S. Chen, “Abusing Public Third-Party Services for EDoS Attacks,” in *Proceedings of the 10th USENIX Workshop on Offensive Technologies (WOOT 16)*, Austin, TX: USENIX Association, 2016. [Online]. Available: https://www.usenix.org/conference/woot16/workshop-program/presentation/wang

[27] M. Kührer, T. Hupperich, C. Rossow, and T. Holz, “Hell of a Handshake: Abusing TCP for Reflective Amplification DDoS Attacks,” in *Proceedings of the 8th USENIX Workshop on Offensive Technologies (WOOT 14)*, San Diego, CA: USENIX Association, 2014. [Online]. Available: https://www.usenix.org/conference/woot14/workshop-program/presentation/kuhrer

[28] O. Yoachimik, “DDoS attack trends for 2022 Q2,” Cloudflare, Jul. 2022. Accessed: Jan. 17, 2023. [Online]. Available: https://blog.cloudflare.com/ddos-attack-trends-for-2022-q2/

[29] R. K. C. Chang, “Defending against flooding-based distributed denial-of-service attacks: a tutorial,” *IEEE Commun. Mag.*, vol. 40, no. 10, pp. 42–51, Oct. 2002, doi: 10.1109/MCOM.2002.1039856.

[30] M. Antonakakis *et al.*, “Understanding the mirai botnet,” in *Proceedings of the 26th USENIX Security Symposium*, Vancouver, BC, Canada, Aug. 2017, pp. 1092–1110.

[31] Hurricane Electric LLC, “Internet Backbone and Colocation Provider.” http://he.net/ (accessed Jul. 04, 2018).

[32] P. Ferguson and D. Senie, “Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing.” The Internet Society, May 2000. [Online]. Available: https://tools.ietf.org/html/bcp38

[33] F. Baker and P. Savola, “Ingress Filtering for Multihomed Networks.” The Internet Society, Mar. 2004. [Online]. Available: https://tools.ietf.org/html/bcp84

[34] NTP Public Services Project, “History,” *NTP FAQ*. http://www.ntp.org/ntpfaq/NTP-s-def-hist.htm (accessed Nov. 05, 2017).

[35] C. Rossow, “Amplification Hell: Revisiting Network Protocols for DDoS Abuse,” in *Proceedings of the 2014 Network and Distributed System Security Symposium*, San Diego, CA, USA: Internet Society, Feb. 2014. doi: 10.14722/ndss.2014.23233.

[36] J. Czyz, M. Kallitsis, M. Gharaibeh, C. Papadopoulos, M. Bailey, and M. Karir, “Taming the 800 pound gorilla: The rise and decline of NTP DDoS attacks,” in *Proceedings of the 2014 Conference on Internet Measurement Conference*, ACM, 2014, pp. 435–448.

[37] M. Anagnostopoulos, G. Kambourakis, S. Gritzalis, and D. K. Yau, “Never say never: Authoritative TLD nameserver-powered DNS amplification,” in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*, IEEE, 2018.

[38] M. Anagnostopoulos, G. Kambourakis, P. Kopanos, G. Louloudakis, and S. Gritzalis, “DNS amplification attack revisited,” *Computers & Security*, vol. 39, pp. 475–485, 2013.

[39] D. Kopp, C. Dietzel, and O. Hohlfeld, “DDoS Never Dies? An IXP Perspective on DDoS Amplification Attacks,” in *Passive and Active Measurement*, O. Hohlfeld, A. Lutu, and D. Levin, Eds., in Lecture Notes in Computer Science, vol. 12671. Cham: Springer International Publishing, 2021, pp. 284–301. doi: 10.1007/978-3-030-72582-2_17.

[40] S. M. Mousavi and M. St-Hilaire, “Early detection of DDoS attacks against SDN controllers,” in *Computing, Networking and Communications (ICNC), 2015 International Conference on*, IEEE, 2015, pp. 77–81.

[41] X. Ma and Y. Chen, “DDoS detection method based on chaos analysis of network traffic entropy,” *IEEE Communications Letters*, vol. 18, no. 1, pp. 114–117, 2014.

[42] R. Karimazad and A. Faraahi, “An anomaly-based method for DDoS attacks detection using RBF neural networks,” in *Proceedings of the International Conference on Network and Electronics Engineering*, 2011, pp. 44–48.

[43] J. J. Santanna *et al.*, “Booters—An analysis of DDoS-as-a-service attacks,” in *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, IEEE, 2015, pp. 243–251.

[44] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, “Inferring internet denial-of-service activity,” *ACM Transactions on Computer Systems (TOCS)*, vol. 24, no. 2, pp. 115–139, 2006.

[45] C. Fachkha, E. Bou-Harb, and M. Debbabi, “Fingerprinting Internet DNS Amplification DDoS Activities,” presented at the The Sixth IFIP International Conference on New Technologies, Mobility and Security (NTMS), Dubai, UAE: IEEE, Mar. 2014, pp. 1–5. doi: 10.1109/NTMS.2014.6814019.

[46] Q. Yan, F. R. Yu, Q. Gong, and J. Li, “Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 602–622, 2016.

[47] M. Walfish, M. Vutukuru, H. Balakrishnan, D. Karger, and S. Shenker, “DDoS defense by offense,” *ACM Transactions on Computer Systems (TOCS)*, vol. 28, no. 1, p. 3, 2010.

[48] C. Dietzel, A. Feldmann, and T. King, “Blackholing at ixps: On the effectiveness of ddos mitigation in the wild,” in *International Conference on Passive and Active Network Measurement*, Springer, 2016, pp. 319–332.

[49] C. Jin, H. Wang, and K. G. Shin, “Hop-count filtering: an effective defense against spoofed DDoS traffic,” in *Proceedings of the 10th ACM conference on Computer and communications security*, ACM, 2003, pp. 30–41.

[50] J. Mirkovic, E. Kline, and P. Reiher, “RESECT: Self-Learning Traffic Filters for IP Spoofing Defense,” in *Proceedings of the 33rd Annual Computer Security Applications Conference*, ACM, 2017, pp. 474–485.

[51] Center for Applied Internet Data Analysis based at the University of California’s San Diego Supercomputer Center, “State of IP Spoofing.” https://spoofer.caida.org/summary.php (accessed Jul. 23, 2018).

[52] Q. Lone, M. Luckie, M. Korczyński, and M. van Eeten, “Using Loops Observed in Traceroute to Infer the Ability to Spoof,” in *Passive and Active Measurement Conference (PAM)*, Mar. 2017.

[53] R. Banerjee *et al.*, “Internet outages, the eyewitness accounts: Analysis of the outages mailing list,” in *International Conference on Passive and Active Network Measurement*, Springer, 2015, pp. 206–219.

[54] H. S. Gunawi *et al.*, “Why does the cloud stop computing?: Lessons from hundreds of service outages,” in *Proceedings of the Seventh ACM Symposium on Cloud Computing*, ACM, 2016, pp. 1–16.

[55] M. Karami and D. McCoy, “Understanding the Emerging Threat of DDoS-as-a-Service,” in *Proceedings of the 6th USENIX Workshop on Large-Scale Exploits and Emergent Threats*, Washington, D.C.: USENIX, 2013. [Online]. Available: https://www.usenix.org/conference/leet13/workshop-program/presentation/Karami

[56] R. A. Paulson and J. E. Weber, “Cyberextortion: an overview of distributed denial of service attacks against online gaming companies,” *Issues in Information Systems*, vol. 7, no. 2, pp. 52–56, 2006.

[57] M. Karami, Y. Park, and D. McCoy, “Stress testing the booters: Understanding and undermining the business of DDoS services,” in *Proceedings of the 25th International Conference on World Wide Web*, International World Wide Web Conferences Steering Committee, 2016, pp. 1033–1043.

[58] “Low Orbit Ion Cannon.” Accessed: Jul. 09, 2022. [Online]. Available: https://github.com/NewEraCracker/LOIC

[59] R. Fordyce, “DDoS Attacks as Political Assemblages,” *Platform Journal of Media and Communication*, vol. 5, pp. 6–20, 2013.

[60] C. Morales, “NETSCOUT Arbor Confirms 1.7 Tbps DDoS Attack; The Terabit Attack Era Is Upon Us,” Mar. 05, 2018. https://asert.arbornetworks.com/netscout-arbor-confirms-1-7-tbps-ddos-attack-terabit-attack-era-upon-us/ (accessed Mar. 09, 2018).

[61] Akamai SIRT Alerts, “Memcached-fueled 1.3 Tbps attacks,” Mar. 01, 2018. https://blogs.akamai.com/2018/03/memcached-fueled-13-tbps-attacks.html (accessed Mar. 09, 2018).

[62] M. Majkowski, “Memcrashed - Major amplification attacks from UDP port 11211,” *Cloudflare*, Feb. 27, 2018. https://blog.cloudflare.com/memcrashed-major-amplification-attacks-from-port-11211/ (accessed Mar. 03, 2018).

[63] M. Majkowski, “Stupidly Simple DDoS Protocol (SSDP) generates 100 Gbps DDoS,” *Cloudflare*, Jun. 28, 2017. https://blog.cloudflare.com/ssdp-100gbps/ (accessed Aug. 20, 2017).

[64] J. Arteaga and W. Mejia, “CLDAP Reflection DDoS,” Akamai Technologies, Threat advisory, Mar. 2017. Accessed: Jul. 05, 2017. [Online]. Available: https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/cldap-threat-advisory.pdf

[65] M. Prince, “Technical Details Behind a 400Gbps NTP Amplification DDoS Attack,” Feb. 13, 2014. https://blog.cloudflare.com/technical-details-behind-a-400gbps-ntp-amplification-ddos-attack/ (accessed Jan. 25, 2018).

[66] M. Prince, “The DDoS That Knocked Spamhaus Offline (And How We Mitigated It),” Mar. 20, 2013. https://blog.cloudflare.com/the-ddos-that-knocked-spamhaus-offline-and-ho/ (accessed Jan. 25, 2018).

[67] J. Graham-Cumming, “The Wednesday Witching Hour: CloudFlare DoS Statistics,” Aug. 03, 2012. https://blog.cloudflare.com/the-wednesday-witching-hour-cloudflare-dos-st/ (accessed Jan. 25, 2018).

[68] The Shadowserver Foundation, “The scannings will continue until the Internet improves.” http://blog.shadowserver.org/2014/03/28/the-scannings-will-continue-until-the-internet-improves/ (accessed Jun. 30, 2018).
[69] OpenNTPProject.org, “NTP Scanning Project.” http://openntpproject.org/ (accessed Dec. 08, 2017).
[70] Qihoo 360 Technology Co., Ltd, “Scan volume per 10 minutes,” *NetworkScan Mon*. http://scan.netlab.360.com/ (accessed Jul. 19, 2018).
[71] Open Resolver Project, “Detailed History and Breakdown.” http://openresolverproject.org/breakdown.cgi (accessed Apr. 17, 2017).
[72] The CyberGreen Institute, “Cyber Health Stats.” https://www.cybergreen.net/ (accessed Jan. 23, 2018).
[73] Qihoo 360 Technology Co.,Ltd, “Insight into Global DDoS Threat Landscape,” *DDoS Mon*. https://ddosmon.net/insight/ (accessed Jul. 20, 2018).
[74] B. Carpenter, T. Chown, F. Gont, S. Jiang, A. Petrescu, and A. Yourtchenko, “Analysis of the 64-bit Boundary in IPv6 Addressing,” RFC Editor, RFC7421, Jan. 2015. doi: 10.17487/rfc7421.
[75] F. Gont and T. Chown, “Network Reconnaissance in IPv6 Networks,” RFC Editor, RFC7707, Mar. 2016. doi: 10.17487/RFC7707.
[76] T. Narten, R. Draves, and S. Krishnan, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6,” RFC Editor, RFC4941, Sep. 2007. doi: 10.17487/rfc4941.
[77] O. Gasser, Q. Scheitle, S. Gebhard, and G. Carle, “Scanning the IPv6 Internet: Towards a Comprehensive Hitlist,” in *Traffic Monitoring and Analysis - 8th International Workshop, TMA 2016, Louvain la Neuve, Belgium, April 7-8, 2016.*, 2016. [Online]. Available: http://dl.ifip.org/db/conf/tma/tma2016/tma2016-final51.pdf
[78] L. Hendriks, R. de Oliveira Schmidt, R. van Rijswijk-Deij, and A. Pras, “On the Potential of IPv6 Open Resolvers for DDoS Attacks,” in *Passive and Active Measurement*, M. A. Kaafar, S. Uhlig, and J. Amann, Eds., Cham: Springer International Publishing, 2017, pp. 17–29. doi: 10.1007/978-3-319-54328-4_2.
[79] Z. Durumeric, M. Bailey, and J. A. Halderman, “An Internet-Wide View of Internet-Wide Scanning,” in *23rd USENIX Security Symposium (USENIX Security 14)*, San Diego, CA: USENIX Association, 2014, pp. 65–78. [Online]. Available: https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/durumeric
[80] Shodan, “Shodan is the world’s first search engine for Internet-connected devices.” https://www.shodan.io/ (accessed Jul. 09, 2017).
[81] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, “A Search Engine Backed by Internet-Wide Scanning,” in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, Denver, CO, USA: ACM Press, Oct. 2015, pp. 542–553. doi: 10.1145/2810103.2813703.
[82] Z. Durumeric, E. Wustrow, and J. A. Halderman, “ZMap: Fast Internet-wide Scanning and Its Security Applications,” in *Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13)*, Washington, D.C.: USENIX, 2013, pp. 605–620. [Online]. Available: https://www.usenix.org/conference/usenixsecurity13/technical-sessions/paper/durumeric

[83] D. Adrian, Z. Durumeric, G. Singh, and A. Halderman, “Zippier ZMap: Internet-Wide Scanning at 10 Gbps,” in *WOOT’14 Proceedings of the 8th USENIX conference on Offensive Technologies*, San Diego, CA, Aug. 2014.
[84] The ZMap Team, “The ZMap Project.” https://zmap.io/ (accessed Aug. 30, 2017).
[85] R. Graham, “MASSCAN: Mass IP port scanner,” *GitHub*. https://github.com/robertdavidgraham/masscan (accessed Mar. 06, 2017).
[86] D. Myers, E. Foo, and K. Radke, “Internet-wide scanning taxonomy and framework,” in *Australasian Information Security Conference (ACSW-AISC)*, I. Welch and X. Yi, Eds., Sydney, NSW: Australian Computer Society, Inc, 2015. [Online]. Available: https://eprints.qut.edu.au/81897/
[87] G. Lyon, “Nmap: the Network Mapper - Free Security Scanner.” https://nmap.org/ (accessed Jan. 15, 2017).
[88] J. Postel, “User Datagram Protocol,” RFC Editor, RFC0768, Aug. 1980. doi: 10.17487/rfc0768.
[89] J. Postel, “Internet Protocol,” RFC Editor, RFC0791, Sep. 1981. doi: 10.17487/rfc0791.
[90] US-CERT, “UDP-Based Amplification Attacks,” Mar. 02, 2018. https://www.us-cert.gov/ncas/alerts/TA14-017A (accessed Apr. 09, 2018).
[91] Internet Systems Consortium, Inc., “A Quick Introduction to Response Rate Limiting.” https://kb.isc.org/article/AA-01000/0/A-Quick-Introduction-to-Response-Rate-Limiting.html (accessed Nov. 08, 2017).
[92] American Registry for Internet Numbers, Ltd., “Whois-RWS API Documentation.” https://www.arin.net/resources/whoisrws/whois_api.html (accessed Aug. 14, 2017).
[93] H. Debar, M. Dacier, and A. Wespi, “A revised taxonomy for intrusion-detection systems,” in *Annales des télécommunications*, Springer, 2000, pp. 361–378.
[94] A. Mirian *et al.*, “An Internet-wide view of ICS devices,” in *Proceedings of 2016 14th Annual Conference on Privacy, Security and Trust (PST)*, Dec. 2016, pp. 96–103. doi: 10.1109/PST.2016.7906943.
[95] R. Lippmann, D. Fried, K. Piwowarski, and W. Streilein, “Passive operating system identification from TCP/IP packet headers,” in *Workshop on Data Mining for Computer Security*, Citeseer, 2003, pp. 40–49.
[96] T. Kohno, A. Broido, and K. C. Claffy, “Remote physical device fingerprinting,” *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 2, pp. 93–108, 2005.
[97] M. Dahlmanns, J. Lohmöller, I. B. Fink, J. Pennekamp, K. Wehrle, and M. Henze, “Easing the Conscience with OPC UA: An Internet-Wide Study on Insecure Deployments,” in *Proceedings of the ACM Internet Measurement Conference*, Virtual Event USA: ACM, Oct. 2020, pp. 101–110. doi: 10.1145/3419394.3423666.
[98] X. Feng, Q. Li, H. Wang, and L. Sun, “Acquisitional Rule-based Engine for Discovering Internet-of-Things Devices,” in *27th USENIX Security Symposium (USENIX Security 18)*, Baltimore, MD: USENIX Association, Aug. 2018, pp. 327–341. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/feng
[99] M. Nawrocki, T. C. Schmidt, and M. Wahlisch, “Uncovering Vulnerable Industrial Control Systems from the Internet Core,” in *NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium*, Budapest, Hungary: IEEE, Apr. 2020, pp. 1–9. doi: 10.1109/NOMS47738.2020.9110256.

[100] K. Yang, Q. Li, and L. Sun, “Towards automatic fingerprinting of IoT devices in the cyberspace,” *Computer Networks*, vol. 148, pp. 318–327, Jan. 2019, doi: 10.1016/j.comnet.2018.11.013.

[101] Y. Meidan *et al.*, “ProfilIoT: a machine learning approach for IoT device identification based on network traffic analysis,” in *Proceedings of the Symposium on Applied Computing*, Marrakech Morocco: ACM, Apr. 2017, pp. 506–509. doi: 10.1145/3019612.3019878.

[102] A. Sivanathan *et al.*, “Characterizing and classifying IoT traffic in smart cities and campuses,” in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Atlanta, GA: IEEE, May 2017, pp. 559–564. doi: 10.1109/INFCOMW.2017.8116438.

[103] B. Bezawada, M. Bachani, J. Peterson, H. Shirazi, I. Ray, and I. Ray, “Behavioral Fingerprinting of IoT Devices,” in *Proceedings of the 2018 Workshop on Attacks and Solutions in Hardware Security - ASHES ’18*, Toronto, Canada: ACM Press, 2018, pp. 41–50. doi: 10.1145/3266444.3266452.

[104] P. Yadav, A. Feraudo, B. Arief, S. F. Shahandashti, and V. G. Vassilakis, “Position paper: A systematic framework for categorising IoT device fingerprinting mechanisms,” in *Proceedings of the 2nd International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things*, Virtual Event Japan: ACM, Nov. 2020, pp. 62–68. doi: 10.1145/3417313.3429384.

[105] A. Ignatiev, “Towards Trustable Explainable AI,” in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, Yokohama, Japan: International Joint Conferences on Artificial Intelligence Organization, Jul. 2020, pp. 5154–5158. doi: 10.24963/ijcai.2020/726.

[106] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017, pp. 4765–4774. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[107] N. L. Tsakiridis *et al.*, “Versatile Internet of Things for Agriculture: An eXplainable AI Approach,” in *Artificial Intelligence Applications and Innovations*, I. Maglogiannis, L. Iliadis, and E. Pimenidis, Eds., in IFIP Advances in Information and Communication Technology, vol. 584. Cham: Springer International Publishing, 2020, pp. 180–191. doi: 10.1007/978-3-030-49186-4_16.

[108] I. Garcia-Magarino, R. Muttukrishnan, and J. Lloret, “Human-Centric AI for Trustworthy IoT Systems With Explainable Multilayer Perceptrons,” *IEEE Access*, vol. 7, pp. 125562–125574, 2019, doi: 10.1109/ACCESS.2019.2937521.

[109] R. Fielding and J. Reschke, “Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content,” RFC Editor, RFC7231, Jun. 2014. doi: 10.17487/rfc7231.

[110] R. Fielding and J. Reschke, “Hypertext Transfer Protocol (HTTP/1.1): Authentication,” RFC Editor, RFC7235, Jun. 2014. doi: 10.17487/rfc7235.

[111] MaxMind, Inc, “GeoLite2 Free Downloadable Databases,” *MaxMind Developer Site*. https://dev.maxmind.com/geoip/geoip2/geolite2/ (accessed Apr. 10, 2018).

[112] I. Cvitić, D. Peraković, M. Periša, and M. Botica, “Novel approach for detection of IoT generated DDoS traffic,” *Wireless Networks*, Jun. 2019, doi: 10.1007/s11276-019-02043-1.
[113] A. Lavrenovs, “Web accessible device security in Latvia’s Internet,” Master’s Thesis, University of Latvia, Riga, 2013. [Online]. Available: https://dspace.lu.lv/dspace/handle/7/19030
[114] “ZGrab 2.0 Fast Go Application Scanner.” Accessed: Oct. 12, 2022. [Online]. Available: https://github.com/zmap/zgrab2
[115] A. V. Arzhakov and I. F. Babalova, “Analysis of current internet wide scan effectiveness,” in *Proceedings of 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, St. Petersburg, Russia: IEEE, Feb. 2017, pp. 96–99. doi: 10.1109/EIConRus.2017.7910503.
[116] The ZMap Team, “zmap/src/state.c global configuration and defaults,” *GitHub*. https://github.com/zmap/zmap/blob/master/src/state.c (accessed Jun. 29, 2018).
[117] J. Postel, “Internet Control Message Protocol,” RFC Editor, RFC0792, Sep. 1981. doi: 10.17487/rfc0792.
[118] D. L. Mills, “Network Time Protocol (NTP),” RFC Editor, RFC0958, Sep. 1985. doi: 10.17487/rfc0958.
[119] D. Mills, “Network Time Protocol (Version 3) Specification, Implementation and Analysis,” RFC Editor, RFC1305, Mar. 1992. doi: 10.17487/rfc1305.
[120] D. Mills, J. Martin, J. Burbank, and W. Kasch, “Network Time Protocol Version 4: Protocol and Algorithms Specification,” RFC Editor, RFC5905, Jun. 2010. doi: 10.17487/rfc5905.
[121] The ZMap Team, “UDP Data Probes,” *GitHub*. https://github.com/zmap/zmap/blob/master/examples/udp-probes/README (accessed Mar. 09, 2018).
[122] P. V. Mockapetris, “Domain names - implementation and specification,” RFC Editor, RFC1035, Nov. 1987. doi: 10.17487/rfc1035.
[123] S. Bradshaw and L. DeNardis, “The politicization of the Internet’s Domain Name System: Implications for Internet security, universality, and freedom,” *New Media & Society*, vol. 20, no. 1, pp. 332–350, 2018.
[124] American Registry for Internet Numbers, Ltd., “This file contains a list of autonomous system numbers and names of all registered ASNs.” ftp://ftp.arin.net/info/asn.txt (accessed Jun. 12, 2018).
[125] Root Server Operators, “Events of 2015-11-30,” Dec. 04, 2015. http://root-servers.org/news/events-of-20151130.txt (accessed Sep. 04, 2017).
[126] J. Damas, M. Graff, and P. Vixie, “Extension Mechanisms for DNS (EDNS(0)),” RFC Editor, RFC6891, Apr. 2013. doi: 10.17487/rfc6891.
[127] R. van Rijswijk-Deij, A. Sperotto, and A. Pras, “DNSSEC and its potential for DDoS attacks: a comprehensive measurement study,” in *Proceedings of the 2014 Conference on Internet Measurement Conference*, ACM, 2014, pp. 449–460.
[128] J. Dickinson, S. Dickinson, R. Bellis, A. Mankin, and D. Wessels, “DNS Transport over TCP - Implementation Requirements,” RFC Editor, RFC7766, Mar. 2016. doi: 10.17487/RFC7766.

[129] Microsoft, “Response Rate Limiting in Windows DNS Server.” https://blogs.technet.microsoft.com/teamdhcp/2015/08/28/response-rate-limiting-in-windows-dns-server/ (accessed Jun. 11, 2018).
[130] Internet Systems Consortium, Inc., *BIND 9 Administrator Reference Manual*.
[131] P. Vixie and V. Schryver, “DNS Response Rate Limiting (DNS RRL),” ISC-TN-2012-1-Draft1, Apr. 2012. Accessed: May 22, 2018. [Online]. Available: https://ftp.isc.org/isc/pubs/tn/isc-tn-2012-1.txt
[132] A. Donoho *et al.*, “UPnP Device Architecture 2.0.” Apr. 17, 2020. Accessed: Apr. 27, 2020. [Online]. Available: https://openconnectivity.org/upnp-specs/upnpresources.zip
[133] S. Albright, P. J. Leach, Y. Gu, Y. Y. Goland, and T. Cai, “Simple Service Discovery Protocol/1.0,” Internet Engineering Task Force, Internet-Draft draft-cai-ssdp-v1-03, Nov. 1999. Accessed: Nov. 16, 2021. [Online]. Available: https://datatracker.ietf.org/doc/draft-cai-ssdp-v1/03/
[134] H. D. Moore, “Security Flaws in Universal Plug and Play,” Rapid7, Jan. 2013. Accessed: Nov. 15, 2021. [Online]. Available: https://information.rapid7.com/rs/411-NAK-970/images/SecurityFlawsUPnP%20%281%29.pdf
[135] Emergency Response Team, “SSDP DDoS Attack Mitigation,” Radware, Nov. 2014. Accessed: Nov. 17, 2021. [Online]. Available: https://support.radware.com/ci/okcsFattach/get/16387_3
[136] “A New Twist In SSDP Attacks,” NETSCOUT, Jun. 2018. Accessed: Nov. 15, 2021. [Online]. Available: https://www.netscout.com/sites/default/files/asert-blog/uploads/2018/06/ssdp_diffraction.pdf
[137] “SNMPv3 White Paper,” SNMP Research International, White Paper. Accessed: Dec. 14, 2021. [Online]. Available: https://snmp.com/snmpv3/v3white.shtml
[138] J. D. Case, M. Fedor, M. L. Schoffstall, and J. Davin, “Simple Network Management Protocol (SNMP),” RFC Editor, RFC1157, May 1990. doi: 10.17487/rfc1157.
[139] V. Paxson, “An analysis of using reflectors for distributed denial-of-service attacks,” *SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 3, pp. 38–47, Jul. 2001, doi: 10.1145/505659.505664.
[140] C. Thompson, “SNMP DDoS Vector - Secure Your Network NOW!,” The Spamhaus Project, Dec. 2011. Accessed: Mar. 30, 2023. [Online]. Available: https://www.spamhaus.org/news/article/678/snmp-ddos-vector-secure-your-network-now
[141] “SNMP Reflected Amplification DDoS Attack Mitigation,” Broadband Internet Technical Advisory Group, Technical Working Group Report, Aug. 2012. [Online]. Available: https://www.bitag.org/documents/SNMP-Reflected-Amplification-DDoS-Attack-Mitigation.pdf
[142] R. Presuhn, “Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP),” RFC Editor, RFC3416, Dec. 2002. doi: 10.17487/rfc3416.
[143] D. Mauro and K. J. Schmidt, *Essential SNMP*, 2nd ed. Sebastopol: O’Reilly Media, Inc., 2009.
[144] A. Young, “Connection-less Lightweight X.500 Directory Access Protocol,” RFC Editor, RFC1798, Jun. 1995. doi: 10.17487/rfc1798.

[145] K. Zeilenga, “Connection-less Lightweight Directory Access Protocol (CLDAP) to Historic Status,” RFC Editor, RFC3352, Mar. 2003. doi: 10.17487/rfc3352.
[146] Black Lotus Labs, “CLDAP Reflectors on the Rise Despite Best Practice,” Lumen Technologies, Oct. 2022. Accessed: Apr. 03, 2023. [Online]. Available: https://blog.lumen.com/cldap-reflectors-on-the-rise-despite-best-practice/
[147] T. Sellers, “Project Sonar Study of LDAP on the Internet,” Rapid7, Nov. 2016. Accessed: Oct. 28, 2022. [Online]. Available: https://www.rapid7.com/blog/post/2016/11/08/project-sonar-study-of-ldap-on-the-internet/
[148] D. Allen, “Misconfigured Windows Servers contributed to DDoS attacks,” Oct. 2022. Accessed: Jan. 12, 2023. [Online]. Available: https://www.onmsft.com/news/misconfigured-windows-servers-contributed-to-ddos-attacks/
[149] “[MS-ADTS]: Active Directory Technical Specification,” Microsoft Corporation, Jan. 2023. [Online]. Available: https://winprotocoldoc.blob.core.windows.net/productionwindowsarchives/MS-ADTS/%5bMS-ADTS%5d.pdf
[150] K. Zeilenga, “Lightweight Directory Access Protocol (LDAP): Directory Information Models,” RFC Editor, RFC4512, Jun. 2006. doi: 10.17487/rfc4512.
[151] “AMP-Research,” Repository. Accessed: Nov. 21, 2022. [Online]. Available: https://github.com/Phenomite/AMP-Research
[152] J. Postel, “Character Generator Protocol,” RFC Editor, RFC0864, May 1983. doi: 10.17487/rfc0864.
[153] “Internet Accessible CHARGEN Service,” National Cyber Security Centre, Ireland. Accessed: Mar. 22, 2023. [Online]. Available: https://www.ncsc.gov.ie/emailsfrom/Shadowserver/DoS/Chargen/
[154] S. Mansfield-Devine, “The growth and evolution of DDoS,” *Network Security*, vol. 2015, no. 10, pp. 13–20, Oct. 2015, doi: 10.1016/S1353-4858(15)30092-1.
[155] “Standard 135-2016, BACnet - A Data Communication Protocol for Building Automation and Control Networks,” ASHRAE, 2016.
[156] O. Gasser, Q. Scheitle, C. Denis, N. Schricker, and G. Carle, “Security Implications of Publicly Reachable Building Automation Systems,” in *2017 IEEE Security and Privacy Workshops (SPW)*, San Jose, CA: IEEE, May 2017, pp. 199–204. doi: 10.1109/SPW.2017.13.
[157] O. Gasser, Q. Scheitle, B. Rudolph, C. Denis, N. Schricker, and G. Carle, “The Amplification Threat Posed by Publicly Reachable BACnet Devices,” *JCSANDM*, Nov. 2017, doi: 10.13052/2245-1439.614.
[158] ASERT Team, “CoAP Attacks In The Wild,” NETSCOUT, Jan. 2019. Accessed: Mar. 06, 2023. [Online]. Available: https://www.netscout.com/blog/asert/coap-attacks-wild
[159] M. De Donno, N. Dragoni, A. Giaretta, and A. Spognardi, “DDoS-Capable IoT Malwares: Comparative Analysis and Mirai Investigation,” *Security and Communication Networks*, vol. 2018, pp. 1–30, 2018, doi: 10.1155/2018/7178164.
[160] A. T. Vasques and J. J. C. Gondim, “Amplified Reflection DDoS Attacks over IoT Mirrors: A Saturation Analysis,” in *2019 Workshop on Communication Networks and*

Power Systems (WCNPS), Brasilia, Brazil: IEEE, Oct. 2019, pp. 1–6. doi: 10.1109/WCNPS.2019.8896290.

[161] Jan Just Keijser, *OpenVPN Cookbook - Second Edition*. Packt Publishing, 2017.

[162] H. Nguyen and A. Tsarou, “Remote Workers and the Rise of OpenVPN Amplification DDoS Attacks,” Corero Network Security, White paper, 2020. Accessed: Mar. 27, 2023. [Online]. Available: https://go.corero.com/wp-open-vpn-ddos-download

[163] A. Zhang, “What You Should Know about OpenVPN Reflection Attacks,” NSFOCUS, Sep. 2020. Accessed: Mar. 29, 2023. [Online]. Available: https://nsfocusglobal.com/what-you-should-know-about-openvpn-reflection-attacks/

[164] S. Bjarnason and R. Dobbins, “A Call to ARMS: Apple Remote Management Service UDP Reflection/Amplification DDoS Attacks,” NETSCOUT, Jun. 2019. Accessed: Jan. 18, 2023. [Online]. Available: https://www.netscout.com/blog/asert/call-arms-apple-remote-management-service-udp

[165] Office of the Chief Information Security Officer, “ARD/ARMS abuse and securing MacOSX devices against DDoS Amplification attacks,” UCLA, Security Advisory, Mar. 2020. Accessed: Feb. 21, 2023. [Online]. Available: https://ociso.ucla.edu/news/ardarms-abuse-and-securing-macosx-devices-against-ddos-amplification-attacks

[166] Akamai Security Intelligence Response Team, “NetBIOS name server, RPC portmap and Sentinel reflection DDoS,” Akamai Technologies, Threat Advisory, Oct. 2015.

[167] “Internet Accessible NetBIOS Name Service,” National Cyber Security Centre, Ireland. Accessed: Feb. 23, 2023. [Online]. Available: https://www.ncsc.gov.ie/emailsfrom/Shadowserver/DoS/NetBIOS/

[168] J. Hart, “Understanding Ubiquiti Discovery Service Exposures,” Rapid7, Feb. 2019. Accessed: Feb. 07, 2023. [Online]. Available: https://www.rapid7.com/blog/post/2019/02/01/ubiquiti-discovery-service-exposures/

[169] ASERT Team, “DDoS Attack Vectors Live or Die,” NETSCOUT, Feb. 2020. Accessed: Jan. 19, 2023. [Online]. Available: https://www.netscout.com/blog/asert/ddos-attack-vectors-live-or-die

[170] OASIS, “Web Services Dynamic Discovery (WS-Discovery) Version 1.1,” OASIS Standard, Jul. 2009. [Online]. Available: https://docs.oasis-open.org/ws-dd/discovery/1.1/os/wsdd-discovery-1.1-spec-os.pdf

[171] J. Respeto, “New DDoS Vector Observed in the Wild: WSD attacks hitting 35/Gbps,” Akamai Technologies, Sep. 2019. Accessed: Mar. 10, 2023. [Online]. Available: https://www.akamai.com/blog/security/new-ddos-vector-observed-in-the-wild-wsd-attacks-hitting-35gbps

[172] A. Zhang, “A Look Into WS-Discovery Reflection Attacks for 2020 Q1,” NSFOCUS, May 2020. Accessed: Mar. 14, 2023. [Online]. Available: https://nsfocusglobal.com/a-look-into-ws-discovery-reflection-attacks-for-2020-q1/

[173] “Abuse of WS-Discovery Protocol Can Lead to Large-Scale DDoS Attacks,” Trend Micro, Aug. 2019. Accessed: Mar. 14, 2023. [Online]. Available: https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/abuse-of-ws-discovery-protocol-can-lead-to-large-scale-ddos-attacks

[174] “Project Turris Greylist,” CZ.NIC, Czechia. Accessed: Sep. 23, 2022. [Online]. Available: https://project.turris.cz/en/greylist.html
[175] J. Fisher, “E-mail: GreyNoise Netsecscanner Inquiry,” Mar. 23, 2023.
[176] Y. Nosyk, M. Korczyński, and A. Duda, “Routing Loops as Mega Amplifiers for DNS-Based DDoS Attacks,” in *Passive and Active Measurement*, O. Hohlfeld, G. Moura, and C. Pelsser, Eds., in Lecture Notes in Computer Science, vol. 13210. Cham: Springer International Publishing, 2022, pp. 629–644. doi: 10.1007/978-3-030-98785-5_28.
[177] E. Rodríguez, A. Noroozian, M. van Eeten, and C. Gañán, “Superspreaders: Quantifying the Role of IoT Manufacturers in Device Infections,” in *Workshop on the Economics of Information Security (WEIS)*, 2021.
[178] E. Leverett, R. Clayton, and R. Anderson, “Standardisation and Certification of the ‘Internet of Things,’” presented at the 16th Annual Workshop on the Economics of Information Security, CA, USA, Jun. 2017.
[179] T. Heinrich, R. R. Obelheiro, and C. A. Maziero, “New Kids on the DRDoS Block: Characterizing Multiprotocol and Carpet Bombing Attacks,” in *Passive and Active Measurement*, O. Hohlfeld, A. Lutu, and D. Levin, Eds., in Lecture Notes in Computer Science, vol. 12671. Cham: Springer International Publishing, 2021, pp. 269–283. doi: 10.1007/978-3-030-72582-2_16.
[180] Charter Communications, “Spectrum Support,” *Blocked Ports*. https://www.spectrum.net/support/internet/blocked-ports (accessed Nov. 08, 2022).
[181] State of California, *SB-327 Information privacy: connected devices.* 2018. [Online]. Available: http://leginfo.legislature.ca.gov/faces/billNavClient.xhtml?bill_id=201720180SB327

APPENDIX A: EXAMPLE MEASUREMENT OUTPUT


```
{"ip": "X.X.X.X", "start_time": 1588550688.7644281, "responses": {"X.X.X.X": [[10, 3028, 0.03528285026550293, 0.036428928375244414], [10, 3028, 0.054569959640502 93, 0.055734872817993164], [10, 3028, 0.020968914031982422, 0.022217750549316406], [10, 3028, 0.0642237663269043, 0.06536293029785156], [10, 3028, 0.043653964996337 89, 0.044815778732299805], [10, 3028, 0.022374868392944336, 0.02420496940612793], [10, 3028, 0.042370796203613 28, 0.043540954589843 75], [10, 3028, 0.053272962570190 43, 0.054404973983764 65], [10, 3028, 0.049173831939697266, 0.050256967544555664], [10, 3028, 0.024599790573120117, 0.027787923812866 21], [10, 3028, 0.036708831787109375, 0.037958860397338 87], [10, 3028, 0.033987760543823 24, 0.035088775542114 26], [10, 3028, 0.041010856628417 97, 0.042154788970947266], [10, 3028, 0.038144826889038086, 0.039289951324462 89], [10, 3028, 0.031628847122192 38, 0.033788919448852 54], [10, 3028, 0.045068979263305664, 0.046248912811279 3], [10, 3028, 0.047730922698974 61, 0.048931837081909 18], [10, 3028, 0.050429821014404 3, 0.051719903945922 85], [10, 3028, 0.046413898468017 58, 0.047540903091430664], [10, 3028, 0.028375864028930664, 0.031262874603271484], [10, 3028, 0.07112479209899902, 0.072305917739866 16], [10, 3028, 0.055925846099853516, 0.057057857513427734], [10, 3028, 0.057284832000732 42, 0.058535814285278 32], [10, 3028, 0.039535760879516 6, 0.040838956832885 74], [10, 3028, 0.067026853561401 37, 0.068204879760742 19], [10, 3028, 0.058706760406494 14, 0.059920787811279 3], [10, 3028, 0.061421871185302734, 0.062704801555944824], [10, 3028, 0.09051895141601562, 0.09163975715637207], [10, 3028, 0.068421840667724 61, 0.069563865661621 1], [10, 3028, 0.051927804946899414, 0.053068876266479 49], [10, 3028, 0.072505950927734 38, 0.073729753494262 7], [10, 3028, 0.065692901611328 12, 0.066830873489379 88], [10, 3028, 0.076590776443481 45, 0.077901840209960 94], [10, 3028, 0.086436748504638 67, 0.087597846984863 28], [10, 3028, 0.075277805328369 14, 0.076403856773376 46], [10, 3028, 0.079415798187255 86, 0.080556869506835 94], [10, 3028, 0.093164920806884 77, 0.094294786453247 07], [10, 3028, 0.080713748931884 77, 0.082050800323486 33], [10, 3028, 0.085166931152343 75, 0.086267948150634 77], [10, 3028, 0.073985810048217 77, 0.075096845626831 05], [10, 3028, 0.083597898483276 37, 0.084906816482543 95], [10, 3028, 0.069713830947875 98, 0.070965766906738 28], [10, 3028, 0.082298755645751 95, 0.083402872085571 29], [10, 3028, 0.062901973724365 23, 0.064031839370727 54], [10, 3028, 0.060178756713867 19, 0.061297893524169 92], [10, 3028, 0.078118801116943 36, 0.079282760620117 19], [10, 3028, 0.091767787933349 61, 0.092971801757812 5], [10, 3028, 0.094492912292480 47, 0.095743894577026 37], [10, 3028, 0.089142799377441 4, 0.090256923397583 01], [10, 3028, 0.087800979614257 81, 0.089010953903198 24]],
"X.X.X.X_packets":
["485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a726f6f746465766963650d0a55534e3a20757569643a6264653131653232 2d316464312d313162322d383738332d6665653938373838373961663a3a75726e3a736368656d61732d75706e702d6f72673a6465766963653a5a7414e4465766963653a0d0a55534e3a20757569643a62646533386230382d316464312d313162322d383136362d643938396237633130383061 3a3a75726e3a736368656d61732d75706e702d6f72673a6465766963653a57414e4465766963653a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a726f6f746465766963650d0a55534e3a20757569643a6264653131653232 2d316464312d313162322d383738332d6665653938373838373961663a3a75726e3a736368656d61732d75706e702d6f72673a6465766963653a5a7414e4465766963653a0d0a55534e3a20757569643a62646533386230382d316464312d313162322d383136362d643938396237633130383061 3a3a75726e3a736368656d61732d75706e702d6f72673a6465766963653a57414e4465766963653a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a736572766963653a57414e436f6d6d6f6e496e7465726616365436f6e6e6669673a0d0a55534e3a20757569643a62646533386230382d316464312d313162322d383136362d643938396237633130383061 3a3a75726e3a736368656d61732d75706e702d6f72673a736572766963653a57414e436f6d6d6f6e496e7465726616365436f6e6e6669673a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a726f6f746e6e6e656374696f6e4465766963653a0d0a55534e3a20757569643a62646533761376 2612d316464312d313162322d383136652d6134373039363238663234323a3a75726e3a736368656d61732d75706e702d6f72673a736572766963653a57414e436f6e6e6e656374696f6e3a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a736572766963653a57414e505050436f6e6e6e656374696f6e3a0d0a55534e3a20757569643a62646537613762612d316464312d313162322d383136652d6134373039363238663234323a3a75726e3a736368656d61732d75706e702d6f72673a736572766963653a57414e505050436f6e6e6e656374696f6e3a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a736572766963653a57414e4950436f6e6e6e656374696f6e3a0d0a55534e3a20757569643a62646537613762612d316464312d313162322d383136652d6134373039363238663234323a3a75726e3a736368656d61732d75706e702d6f72673a736572766963653a57414e4950436f6e6e6e656374696f6e3a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a736572766963653a4c617965723466 3f7277617264696e673a0d0a55534e3a20757569643a62646531316532322d316464312d313162322d383738332d6665653938373838373961663a3a75726e3a736368656d61732d75706e702d6f72673a736572766963653a4c617965723466 3f7277617264696e673a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a736572766963653a77696669616c6c69616e63652d6f72673a6465766963653a5746414465766963653a0d0a55534e3a20757569643a62646561313932382d316464312d313162322d613864302d623230343239656164383863a3a75726e3a736368656d61732d77696669616c6c69616e63652d6f72673a6465766963653a5746414465766963653a0d0a0d0a",
"485454502f312e3120323030204f4b0d0a43616368652d436f6e74726f6c3a206d61782d6167653d3132300d0a4558543a0d0a4c6f636174696f6e3a20687474703a2f2f3139322e3136382e302e313a36353533352f726f6f74446573632e786d6c0d0a5365727665723a204c696e75782f322e342e32322d312e323131352e6e70746c2055506e502f312e30206d696e6975706e70642f312e300d0a53543a2075706e703a736572766963653a77696669616c6c69616e63652d6f72673a736572766963653a574641574c414e436f6e6e6669673a0d0a55534e3a20757569643a62646561313932382d316464312d313162322d613864302d623230343239656164383863a3a75726e3a736368656d61732d77696669616c6c69616e63652d6f72673a736572766963653a574641574c414e436f6e6e6669673a0d0a0d0a"]}}
```

APPENDIX B: SSDP MOST COMMON RESPONSE PAYLOAD

```
HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: upnp:rootdevice
USN: uuid:bde11e22-1dd1-11b2-8783-fee9878879af::upnp:rootdevice

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-upnp-org:device:InternetGatewayDevice:
USN: uuid:bde11e22-1dd1-11b2-8783-fee9878879af::urn:schemas-upnp-
org:device:InternetGatewayDevice:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-upnp-org:device:WANDevice:
USN: uuid:bde38b08-1dd1-11b2-8166-d989b7c1080a::urn:schemas-upnp-org:device:WANDevice:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-upnp-org:service:WANCommonInterfaceConfig:
USN: uuid:bde38b08-1dd1-11b2-8166-d989b7c1080a::urn:schemas-upnp-
org:service:WANCommonInterfaceConfig:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-upnp-org:device:WANConnectionDevice:
USN: uuid:bde7a7ba-1dd1-11b2-816e-a4709628f242::urn:schemas-upnp-org:device:WANConnectionDevice:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-upnp-org:service:WANPPPConnection:
USN: uuid:bde7a7ba-1dd1-11b2-816e-a4709628f242::urn:schemas-upnp-org:service:WANPPPConnection:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-upnp-org:service:WANIPConnection:
USN: uuid:bde7a7ba-1dd1-11b2-816e-a4709628f242::urn:schemas-upnp-org:service:WANIPConnection:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-upnp-org:service:Layer3Forwarding:
USN: uuid:bde11e22-1dd1-11b2-8783-fee9878879af::urn:schemas-upnp-org:service:Layer3Forwarding:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-wifialliance-org:device:WFADevice:
USN: uuid:bdea1928-1dd1-11b2-a8d0-b20429ead886::urn:schemas-wifialliance-org:device:WFADevice:

HTTP/1.1 200 OK
Cache-Control: max-age=120
EXT:
Location: http://192.168.0.1:65535/rootDesc.xml
Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
ST: urn:schemas-wifialliance-org:service:WFAWLANConfig:
USN: uuid:bdea1928-1dd1-11b2-a8d0-b20429ead886::urn:schemas-wifialliance-
org:service:WFAWLANConfig:
```