Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**An Experimental Analysis of Current DDoS Attacks Based on a Provider Edge Router Honeynet** **Πειραματική μελέτη των σύγχρονων καταμενημένων επιθέσεων άρνησης υπηρεσίας μέσω της εφαρμογής συστήματος Honeynet σε ακραίο δρομολογητή παρόχου**
Ονοματεπώνυμο Φοιτητή	**Σταματία Τριαντοπούλου**
Πατρώνυμο	**Δημήτριος**
Αριθμός Μητρώου	**ΜΠΣΠ 15088**
Επιβλέπων	**Παναγιώτης Κοτζανικολάου, Επίκουρος Καθηγητής Τμήματος Πληροφορικής**

Ημερομηνία Παράδοσης **Μάρτιος 2018**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)	(υπογραφή)	(υπογραφή)
Κοτζανικολάου Παναγιώτης Επίκουρος Καθηγητής	Δουληγέρης Χρήστος Καθηγητής	Ψαράκης Μιχαήλ Επίκουρος Καθηγητής

Acknowledgement

I would first like to thank my thesis supervisor Kotzanikolaou Panayiotis, Assistant Professor at the University of Piraeus, Department of Informatics, for his continuous support, his inspiring guidance and encouragement throughout my graduate studies and thesis writing.

I would also like to thank my colleagues Mr. Nikolopoulos Yiannis and Mr. Rountos Dimitris who were involved in the preparation of this thesis. They provided me with a stimulating and supportive environment for my research, so as to implement the practical part of this thesis in the telecommunication organization where I work.

Last but certainly not least, I would like to express my gratitude to my family, for their support, encouragement and patience during my studies and for inspiring me to follow my dreams.

Abstract

Network security has become an increasing problem in the world of telecommunication networks. Cyber-attacks against telecommunication networks may result in very high consequences for data and services, while at the same time IT departments phase increasing security challenges, without always having the required resources. Network experts have tried to face up this serious issue by improving the technical awareness of the threats and technical solutions in transmission networks.

The global growing of using the Internet has made securing networks and information one of the most challenge tasks in the field of networks communications. The transmission of data through telecommunication networks introduces various issues, as it is often vulnerable to malicious attacks. Also, the connection of those networks with the Internet enables attacks by external users.

Security attacks may attempt to undermine the confidentiality, integrity and availability. Attacks such as Distributed Denial of Service (DDoS) are achieved when they can affect the availability of information resources. However, the success and the impact of such kind of attacks differ from the victim and the levels of risk, threats and implications for DDoS activity determined in each case separately.

The goal of this thesis is to analyze the current trends of DDoS attacks. To achieve this we have implemented a honeynet system at an edge router of an Internet provider. The practical part analyzes a collection of unclassified data from a honeynet system in an effort to generate useful threat intelligence and prioritization from the data. This model allows us to obtain a better understanding of threat profile and propose solutions to mitigate the attacks and recommend safety measures depending on the attack.

Keywords: Network security, DDoS attacks, IoT attacks, botnets, honeynet

Περίληψη

Η ασφάλεια δικτύων έχει γίνει όλο και μεγαλύτερο πρόβλημα στον κόσμο των τηλεπικοινωνιακών δικτύων. Οι επιθέσεις στον κυβερνοχώρο κατά των τηλεπικοινωνιακών δικτύων μπορεί να έχουν πολύ μεγάλες συνέπειες για τα δεδομένα και τις υπηρεσίες, ενώ ταυτόχρονα τα τμήματα πληροφορικής αυξάνουν τις προκλήσεις ασφάλειας, χωρίς πάντα να έχουν τους απαιτούμενους πόρους. Οι εμπειρογνώμονες δικτύων προσπάθησαν να αντιμετωπίσουν αυτό το σοβαρό ζήτημα βελτιώνοντας το τεχνικό τους υπόβαθρο ώστε να είναι σε θέση να εντοπίζουν τέτοιου είδους απειλές σε δίκτυα μεταφοράς κορμού και να τις επιλύουν.

Η παγκόσμια χρήση του Διαδικτύου έχει καταστήσει την ανάγκη για ασφαλή δίκτυα μια τις πιο ενδιαφέρουσες πτυχές στον τομέα των δικτύων επικοινωνιών. Η μετάδοση των δεδομένων μέσω τηλεπικοινωνιακών δικτύων εισάγει διάφορα ζητήματα ασφάλειας καθώς συχνά προκαλούνται κακόβουλες ενέργειες από εξωτερικούς χρήστες.

Οι επιθέσεις ασφαλείας μπορεί να επιχειρήσουν να υπονομεύσουν την εμπιστευτικότητα, την ακεραιότητα και τη διαθεσιμότητα. Επιθέσεις όπως κατανεμημένες επιθέσεις άρνησης εξυπηρέτησης (DDoS) επιτυγχάνονται όταν επηρεάζουν τη διαθεσιμότητα των πληροφοριών. Ωστόσο, η επιτυχία και ο αντίκτυπος από τέτοιου είδους επιθέσεις διαφέρουν ανάλογα το θύμα και τα επίπεδα κινδύνου, τις απειλές και επιπτώσεις που προσδιορίζονται σε κάθε περίπτωση ξεχωριστά.

Ο σκοπός αυτής της εργασίας είναι να αναλύσουμε τις τρέχουσες τάσεις των κατανεμημένων επιθέσεων άρνησης εξυπηρέτησης. Για να το επιτύχουμε, εφαρμόσαμε ένα σύστημα honeynet σε έναν ακραίο δρομολογητή ενός παρόχου Διαδικτύου. Το πρακτικό μέρος αναλύει μια συλλογή μη ταξινομημένων δεδομένων από ένα σύστημα honeynet σε μια προσπάθεια δημιουργίας χρήσιμων πληροφοριών απειλών και ιεράρχησης από τα δεδομένα. Αυτό το μοντέλο μας επιτρέπει να κατανοήσουμε καλύτερα το προφίλ της απειλής και να προτείνουμε λύσεις για να μετριάσουμε τις επιθέσεις και μέτρα ασφαλείας ανάλογα με την επίθεση.

Λέξεις Κλειδιά: Ασφάλεια δικτύου, Κατανεμημένες επιθέσεις άρνησης εξυπηρέτησης, IoT επιθέσεις, botnets, honeynet

Table of Contents

List of Figures

List of Tables

List of Abbreviation

DoS	Denial of Service
DDoS	Distributed Denial of Service
BGP	Border Gateway Protocol
DNS	Domain Name System
OSI	Open Systems Interconnection
IDS	Intrusion Detection System
NIDS	Network Intrusion Detection System
ISP	Internet Service Provider
BRAS	Broadband Remote Access Server
BNG	Broadband Network Gateway
IGP	Interior Gateway Protocol
OSPF	Open Shortest Path First
VoIP	Voice over Internet Protocol
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
ICMP	Internet Control Message Protocol
ARP	Address Resolution Protocol
HTTP	Hypertext Transfer Protocol
VLAN	Virtual Local Area Network
MAC	Media Access Control
DHCP	Dynamic Host Configuration Protocol
IPFIX	Internet Protocol Flow Information Export
ACK	Acknowledgement (from TCP 3way handshake)
SYN	Synchronize (from TCP 3way handshake)

TCB	Transmission Control Block
NSF	National Science Foundation
IEEE	Institute of Electrical and Electronics Engineers
ACL	Access Control List
QoS	Quality of Service
ToS	Type of Service
WAN	Wide Area Network
TTL	Time to Live
AUP	Acceptable Use Policy
IoT	Internet of Things
IANA	Internet Assigned Numbers Authority
SIP	Session Initiation Protocol
NTP	Network Time Protocol
SSDP	Simple Service Discovery Protocol
UPnP	Universal Plug and Play
OS	Operational System
DVR	Digital Video Recorder

Chapter 1: Introduction

1.1 Problem statement and motivation

Nowadays, almost all the commercial organizations and individuals are dependent on the Internet: On the one hand, organizations use the Internet either to promote or to provide their services, while the individuals use the Internet for searching and comparing products, gathering information, shopping online and communicating with others through e-mails and social networks. Thus, the Internet provides a platform to run the services and store sensitive information.

Network technology is a key for a wide variety of different applications and services. In modern networks there is a communication gap between the developers of networks and developers of security technology. Network planning is a developed process that depends on the OSI model which has several advantages such as modularity, flexibility and standardization of protocols in contrast to secure network planning which is not a well-developed process. There is not a unique methodology and solution to manage the complexity of security requirements. When transferring packets from one node to another node the communication channel should not be vulnerable to attacks so securing the network is just as important as securing the personal computers.

Network security is a weak link in network systems. The malicious usage and attacks have caused tremendous loss by impairing the functionalities of the networks. Among all network attacks, DoS and DDoS attacks are two of the most harmful threats to network functionality.

Generally, DoS attackers exploit TCP/IP and UDP protocols for launching these attacks. Either, they direct a huge amount of abnormal network packets towards the victim(s) – which results in overloading of their network resources such as consuming entire bandwidth and memory – or they exploit vulnerabilities in network protocols to fail the functioning of network devices. In both cases, network resources or services are restricted for or prevented from the intended users. Major difficulty in detecting such attacks is that network traffic consists of a mix of normal or legitimate traffic and abnormal or attack traffic. Moreover, in most of the cases, attack traffic looks like normal traffic [30].

Traditional intrusion detection relies on the inspection of individual packets, which are scanned for suspicious patterns or activities. However, the massive increase of link speeds and throughputs, especially in large networks, makes this approach ineffective. While the range of attacks that can be performed on backbone networks is increasing, this thesis deals with a particular class of attacks known as DDoS attacks.

Distributed Denial of Service attacks are the primary threat to the quality and availability of Internet services. The growth in IP traffic determines the growth in DDoS attack size. DDoS attacks affect numerous organizations connected to the Internet as they disrupt normal business operations. This constitutes an unresolved issue in the IT world and there has never been an ultimate solution for this. One way to start thinking about the effect of DDoS attacks is to consider why DDoS attacks happen. Common reasons for DDoS incidents include the following [28]:

- The attacker might aim to directly profit from his perceived ability to disrupt the victim's services by demanding payment to avoid the disruption.
- Cyber-criminals sometimes offer DDoS services to take out competitor's websites or otherwise disrupt their operations.
- A DDoS attack might aim to punish the victim for refusing an extortion demand or for causing disruption to the attacker's business model (e.g., spam-sending operations).
- Attackers sometimes might target the organization when fine-tuning DDoS tools and capabilities for future attacks, which will be directed at other victims.
- Some downtime and service disruptions are the result of the non-malicious actions that the organization's employees took by mistake (e.g., a server configuration problem).

DDoS attacks also involve a bunch of compromised systems that have been hijacked and added to a virtual swarm of zombie machines called a botnet[1]. It is impossible to understand whether the devices have been compromised as they work exactly as they normally would.

The fact is that legacy DDoS mitigation solutions, such as scrubbing[2], completely overlook the small, low-threshold attacks. Maybe a DDoS attack is small in size but it doesn't mean that it is not a huge problem. It takes hackers only a few minutes to map a network, steal date, install malware or discover network vulnerabilities.

The problem to the research community takes multiple facets and unfortunately there is no a unique solution to solve this. Some of them are the following:

- Attack Detection. How easily can we detect attacks and how much time we need to mitigate them? Simple solutions like bandwidth metering can take time for the attack to get detected. There must be an intelligent method of quickly and efficiently detecting attacks should be in place. Classical techniques have been used to propose new methods of DoS and DDoS attack detection but the primary challenges lie in performance and availability of packet traffic.
- Mitigation. What's next after an attack has been detected? So far, mechanisms such as random dropping was one of the earliest solutions to handling attacks, although it took the risk to drop legitimate packet requests. Moreover intelligent packet dropping mechanisms based on Traffic Rate Analysis and filtering techniques have also been proposed to mitigate attacks. There is no doubt that filtering techniques by using the Time to Live (TTL) of a network packet introduce a filtering method to selectively suspicious packet flows when an attack is on.

All the above directions of research look to be equally promising in their own right, given the fact that there already exists popular DDoS attack mitigation devices and platforms from network security solutions providers like Cisco and Arbor and still attacks have kept happening with open declarations.

1.2 Related work and challenges

During the last decade, a large number of tools and mechanisms have been developed to defend against the attacks that organizations and hosts are facing. For example, firewalls help to protect these organizations and prevent attackers from performing their unauthorized activities. Moreover, Intrusion Detection Systems (IDSs) allowing companies to detect and identify attacks and provide reaction mechanisms against them. But these tools sometimes lack functionality of detecting new threats and collection of more information about the attacker's activities, methods and skills. Signature based IDS's belong to this category of tools, as they are not capable of detecting new unknown attacks, because they do not have the signatures of the new attacks in their database.

A valuable tool for supporting this mission is the honeynet, a network of seemingly vulnerable honeypot hosts that mimic production machines on an organization's network. A properly configured honeynet can observe threats that may be highly specific to a particular environment, generating valuable data that can inform network defenders.

The first goal would be to understand the subject in detail by taking into consideration the previous incidents and attacks that happened in the past. Scientific articles and related white papers were carefully chosen to understand this subject matter in depth. The second goal would be set up a testbed environment and choose the appropriate tools that are required to carry out the implementation. Honeypots and honeynets are popular tools in the area of network security.

[1] It is also referred to as a "zombie army" https://en.wikipedia.org/wiki/Botnet

[2] Is a technique used by ISPs in order to route potential malicious traffic to an out of path data cleansing station rather than keeping it in the network. When an attack is detected, the traffic is redirected (typically using BGP) to a local scrubbing center where the traffic is analyzed and the attack traffic is filtered out while the clean traffic passes back to the network for delivery.

First surveys in the field of honeypot research presented in 2003 include only a small subset of meanwhile available software and are by this time outdated [2].

Authors in [1] implemented a multi-component honeypot with the front-end component that is easily detectable and penetrable by potential attackers and the back-end component that securely and permanently stores collected data. This honeypot detects and reports telnet attacks on IoT devices.

In [3] T. Luo et al implemented a new type of honeypot called as “intelligent-interaction”. They propose an automatic and intelligent way to collect potential responses using scanner and leverage machine learning techniques to learn the correct behaviors during the interaction with attackers.

Mirai-based DDoS attacks have shown the world the danger that IoT botnets pose for global Internet security and a number of Mirai-related publications emerged. IoT botnets were analyzed in [5] including Mirai. Detailed analysis of Mirai code and behavior can be also found in [1].

In this paper we propose an experimental analysis of current DDoS and other distributed attacks, based on a honeynet architecture installed in an edge router of an ISP. This kind of study requires setting up a honeynet system in an effort to generate useful threat intelligence and prioritization from the captured data.

The proposed system can be applied by large organizations to defend against known DDoS attacks and new future types of attacks. Also, the honeynet can trap the attacker and record the compromised components to provide evidence for use in a legal action.

Against the above discussion, the primary challenges faced in the design of a system effective in mitigating real world DDoS attack are:

- The growing DDoS threat and the severe impact successful attacks have on ISPs network (monitoring network traffic)
- Detect attack behavior before or when it occurs
- What security measures must be taken in order to mitigate DDoS attacks

This thesis also demands proper ways to analyze the attacks when they are actually occurring and suggests possible solutions in mitigating these attacks.

1.3 Contribution of this work and research methodology

The review of the current literature will contribute to identifying the current state of the art on DDoS attacks against high-speed networks as well as the performance of the existing and published detection and mitigation techniques. As such, the literature covers the following main areas:

1. Network layer attacks target layer 3 and 4 of the OSI model and as the name suggests they try to exhaust the network capacity of a victim, which can be the uplink capacity, the network interface controller capacity of the server or also the number of packets that the TCP/IP stack of the operating system of the server can handle.
2. Organizations worried about botnets and DDoS attacks often leverage IDS solutions to mitigate that threat. Detection systems as the honeynet, is a form of passive network monitoring, in which traffic is examined at a packet level and results of the analysis are logged.
3. Particular emphasis will be given to identifying how different network structures impact configuration and performance. In particular, configuration and performance distinctions will be analyzed for large-scale transit networks (global ISP’s network). Techniques and filtering mechanisms will be recommended in order to mitigate the DDoS attacks.

The growing dependence on the Internet makes the impact of successful DDoS attacks increasingly painful for service providers and enterprises. As it is mentioned above, DDoS attacks impose a serious threat to high-speed networks and have serious consequences amongst of the Internet based attacks, e.g. TCP, UDP and ICMP. The next section describes a

comprehensive overview of security issues that ISPs have to take into account in order to secure their network infrastructures.

In 1986, the U.S. National Science Foundation (NSF) established the first backbone network for the Internet. During the 1990s, the explosive growth of the Internet was largely funded by private companies who built their own backbones networks. The Internet eventually became a network of smaller backbones operated by ISP that tap into the biggest national and internal backbones owned by large telecommunications companies [21].

It is common knowledge that Internet is a global system of interconnected computer networks, which is used not only for exchanging information between users but as a “tool” for criminal activities. As the framework of the global network, ISPs are often involved in security incidents, either as a target of an attack or as one of the defenders. Greeks spend more than 80 minutes per day networking, as half of the population is registered on at least one social media platform.

Social media is playing a significant role in the most of the people’s daily life where they are connected together via various online social networking services. So that every user is able to access the complex network using laptops, tablets, smart phones or even simple gadgets. Organizations are adopting the latest networking computer systems and technologies to fulfill their business, social and commercial interests. However, they also bring huge security challenges to deal with security breaches, loss of private and confidential data or service disruption, etc. in order to provide a perfect security mechanism for their clients or customers.

With the widespread use of social media services, telecommunications network usage trends shifted resulting surge of data traffic. For telecommunications network operators, network traffic changed drastically is increasingly difficult to control, especially with the increasing traffic flooding [7].

Backbone network is the largest transmission line that carries data gathered from smaller lines that interconnect with it. It has the ability to connect smaller networks or nodes to create larger networks and transmit data at higher speeds than the rest of the network. As a distributed network consists of a number of devices that are connected to a series of central connectivity devices, such as switches or routers in a hierarchy. Personal computers cannot be connected directly to the backbone network but only through these devices. The participating networks can exceed the core network and introduce private connections with each other for cost reasons, efficiency and safety.

In a typical ISP’s network topology included BNG routers (Edge routers) and Core routers to serve residential and corporate broadband customers (providing IP services, VoIP VPNs and Direct Internet Access). Service providers must be able to rapidly implement security measures against a large number of parties that may be involved in the attack, and deploy these tools and techniques on a large number of devices, usually network entry points. Accordingly, service providers must be able to defend multiple targets from multiple parallel attacks.

As a proof of concept, our analysis is based on real data collected by a honeynet system that was installed on an ISP edge router, for a four-month period time. In the examined scenario, we use an existing ISP’s network topology (Figure 1) and deploy a honeynet system in order to identify and analyze malicious activities based on packets captured and analyzed by a network protocol sniffer and signature-based attack analysis tools.

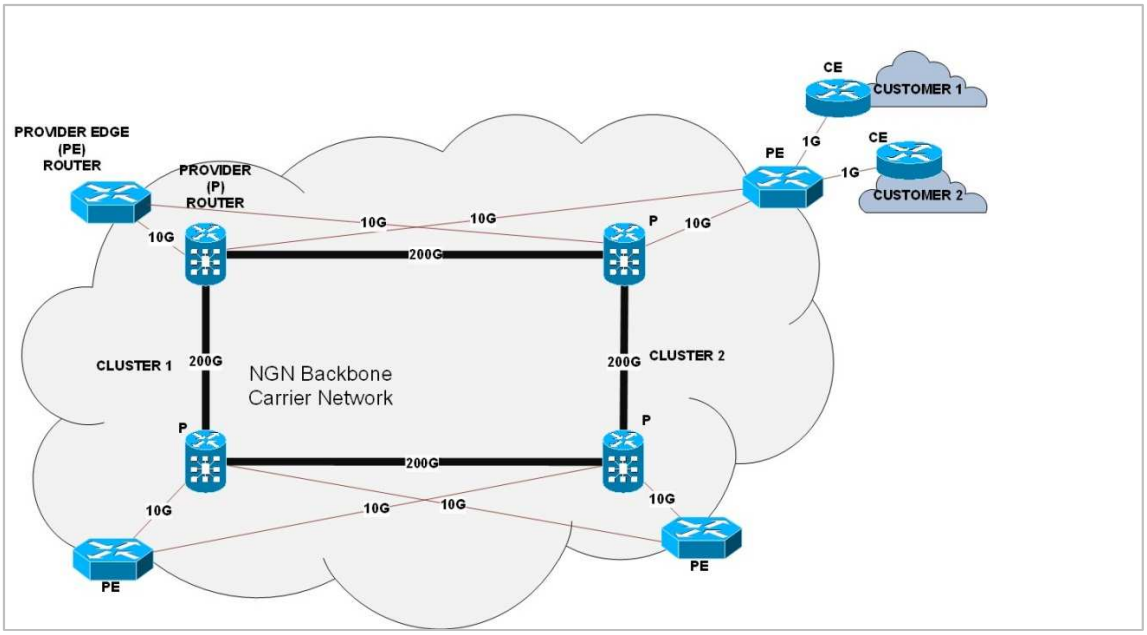


Figure 1: A typical backbone network topology

Given the fact that is difficult to manage the increasing attacks, it is necessary to develop a detection system which allows the monitoring of incoming and outgoing traffic to identify unauthorized usage and mishandling of attackers in computer network systems. This will become feasible in a practical work with a honeynet model, which has the ability to perform real time analysis and record packet traffic on the network layer. It has also the ability to detect possible attacks from unauthorized users. This is achieved by controlling the packet traffic using some rules or signatures that we will be analyzed in the following sections.

The remainder of this thesis is organized as follows:

Chapter 2 covers the IP network traffic and provides information about the 5-layer TCP/IP model and its relevance to the research. The protocols involved at network and transport layers are discussed.

Chapter 3 takes place an example scenario (practical part) that highlights the context of this thesis; Detection tools and packet analyzer sniffers are also referred in this Chapter, followed by the experimental results, along with their detailed analysis.

Chapter 5 presents an in-depth description of how to mitigate attacks and finally, Chapter 6 offers a comprehensive summary of the present work while underlining the main research contributions of the thesis. It further provides an overview of on-going and future work.

Chapter 2: Literature review

2.1 IP network traffic and the functionality of TCP/IP model

This section provides an overview of the Internet model followed by the protocols used at the network and transport layers. The global achievement of the Internet has led to the rapid approbation of the Internet Protocol (IP) technology to build all types of communication networks, including private commercial networks (intranets), military communication networks, private home networks, smart devices and the emerging 4G (and 5G in a few years) cellular networks. The traffic in the modern networks is disparate and most of the applications used in the networking environment prefer IP to transmit the data.

Several models have been developed to categorize the communication protocols into distinct hierarchical structure. The 7-layered OSI model and the 5-layered TCP/IP model are most commonly used in IP network studies. In this thesis, the 5-layered TCP/IP model, illustrated in the table below, is considered. Furthermore, we limit our study to network and transport layer protocols because if we would like to perform packet traffic analysis, we need to be aware of the protocols and their header structure at higher layers.

Layer	7-Layer OSI	5-Layer TCP/IP	Example Protocols and Specifications
L7	Application	Application	Telnet, HTTP, FTP, SMTP, VOIP, SNMP
L6	Presentation		
L5	Session		
L4	Transport	Transport	TCP, UDP
L3	Network	Network	IP, ICMP
L2	Data Link	Data Link	Ethernet (IEEE 802.3), ATM
L1	Physical	Physical	RJ-45, Ethernet (IEEE 802.3)

Table 1: Comparison of 5-layer TCP/IP and 7-Layer OSI models [35]

The network layer is responsible for the transfer of data in the form of packets in an interconnected network. The basic study of network traffic starts from this layer 3, as L1 through L2 do not contribute much from the aspects of end-to-end traffic study. The main function of network protocols is to establish connectivity between all L3 routers and hosts in the network, thereby allowing communication between the direct or indirect connections. The IP protocol, IPv4 and IPv6 (the most recent version of the IP), is the heart of the TCP/IP protocol suite and corresponds to the network layer.

It offers a connectionless and best-effort delivery service to the transport layer. A connectionless service does not require a virtual circuit to be established prior to the process of data transfer thereby offering a packet switched transfer medium where fidelity of the data is not guaranteed. The connectivity between the end-hosts is established by incorporating the routing protocols, such as OSPF and BGP. There are a number of layered management protocols belonging to the network layer. The task of resolving host's physical and IP addresses is managed by ARP and RARP. Maximizing the usage of IP address space allocation to the hosts

can be automated by designing a DHCP environment to the network. Error reporting and delivery of control messages is handled by ICMP.

2.1.1 Transport layer protocols and service ports

Transport protocols utilize the services offered by the underlying network layer protocols and runs on top of the IP. TCP and UDP are two of the most frequently used protocols to support a wide range of applications. Figures 2 and 3 illustrate the header format for TCP and UDP packet segments.

TCP provides reliable, connection-oriented stream service over IP. It sets up a logical full-duplex connection between two end-hosts across a datagram network. It also provides flow and congestion control, thereby allowing receivers to control the rate at which sender transmits information preventing buffers from overflowing. Source IP, destination IP, source port and destination port are the four attributes which help TCP in identifying each connection.

It is commonly known as a three-way handshake procedure for connection establishment and termination and needs proper use of the flags. TCP provides for a harmonious close that involves the independent termination of each direction of the connection, by sending a FIN packet which is reciprocated with an ACK packet. Another alternative for connection termination provided by TCP is through reset (RST) segments. Data reliability is achieved by the use of retransmission timeouts. Hence, TCP uses acknowledge packets and retransmissions to guarantee successful transmission of the data [43].

TCP Segment Header Format

Bit #	0 – 7	8 – 15	16 – 23	24 – 31
0	Source Port		Destination Port	
32	Sequence Number			
64	Acknowledgment Number			
96	Data Offset	Res	Flags	Window Size
128	Header and Data Checksum		Urgent Pointer	
160...	Options			

Figure 2: TCP header format (RFC793) [https://skminhaj.wordpress.com/2016/02/15/tcp-segment-vs-udp-datagram-header-format/]

Unlike TCP, UDP provides a much simpler service to the applications. The UDP is an unreliable, connection-less and not stream-oriented transport layer protocol. It is a very simple protocol which aids in demultiplexing and error checking on the data. It is specified by source and destination ports to identify the connection. Flow and congestion control mechanisms are not utilized by UDP. Since, UDP is connection-less, it does not implement connection establishment and termination procedure. The absence of ACK and retransmission mechanisms makes it an unreliable protocol.

Further, there is no rate control mechanism to adjust the transmission rate and the applications using UDP do not require the heavyweight service of TCP. However, UDP proves to be very helpful in multicasting, network management, routing table updates and real-time multimedia as opposed to TCP [43].

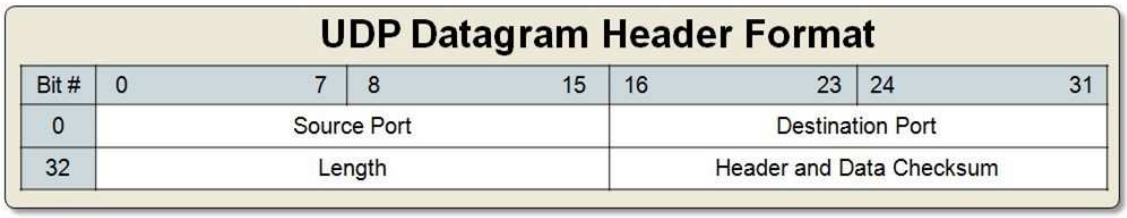


Figure 3: UDP header format (RFC768) [https://skminhaj.wordpress.com/2016/02/15/tcp-segment-vs-udp-datagram-header-format/]

The study of applications and services in a network is of prime importance to ISPs. The TCP and UDP based services keep track of the various applications that are communicating. To differentiate the segments and datagrams for each application, both TCP and UDP have header fields that can uniquely identify these applications. These unique identifiers are the port numbers. In the header of each segment or datagram, there is a source and destination port. The source port number is the number for this communication associated with the originating application on the local host. The destination port number is the number for this communication associated with the destination application on the remote host. Port numbers are assigned in various ways, depending on whether the message is a request or a response. While server processes have static port numbers assigned to them, clients dynamically choose a port number for each conversation. When a client application sends a request to a server application, the destination port contained in the header is the port number that is assigned to the service daemon running on the remote host[3].

Many common applications have default port assignments. For example, TCP port 80 is the default port assigned to web-serving applications. The source port in a segment or datagram header of a client request is randomly generated. As long as it does not conflict with other ports in use on the system, the client can choose any port number. This port number acts like a return address for the requesting application. The transport layer keeps track of this port and the application that initiated the request so that when a response is returned, it can be forwarded to the correct application. The requesting application port number is used as the destination port number in the response coming back from the server. The combination of the transport layer port number and the network layer IP address assigned to the host uniquely identifies a particular process running on a specific host device. This combination is called a socket[4].

2.1.2 Traffic flow analysis for network security

The nature of Internet traffic can be better understood by knowing the concept of the flow. Flow is the sequence of packets that belonged to certain network sessions between end to end hosts. Alternatively, flow is a series of packets that share the same source IP, destination IP, source port, destination port and the protocol. This is most commonly known as five-tuple IP flow, which is an aggregation of individual flows [43].

Traffic flow analysis is the process of measuring the bandwidth usage on a network and analyzing the data for the purpose of performance, capacity planning and making hardware improvement decisions. A TCP flow would start with a SYN packet and end with a FIN or RST packet. Whereas in UDP, it is difficult to define a flow, because UDP is a connection-less, unreliable datagram delivery protocol. Hence, there is no concept of a connection in UDP. However, a flow could still be defined by the controlling application specifying source or destination port number [43].

Network flow data is aggregated packet header data for a communication between a source and a destination. Communications are distinguished by the protocol-level information in the header and the proximity in time (i.e., a flow contains aggregated header information for all

[3] http://www.highteck.net/EN/Transport/OSI_Transport_Layer.html
[4] http://www.highteck.net/EN/Transport/OSI_Transport_Layer.html

packets that use the same protocol settings within a designated time window). There are several reasons that network flow data is a useful format for analyzing network traffic [36]:

- Network flow enables analysts to record the presence of a communication in a very small footprint, which means the data can be collected economically across a large network and stored for months to years.
- Network flow contains sufficient indicative information to allow network defenders to perform a variety of analyses to search for threats or context information that can help defenders understand what is going on.

The network traffic analysis module collects network traffic and bandwidth usage data from any flow enabled device on the network and provides threshold-based alerting to help us address network traffic problems before they impact users, applications and organizations. The most important information that we can collect from traffic analysis tools are:

1. Sender and receiver domains and countries
2. Protocols and services
3. Incoming and outgoing interface traffic and utilization
4. Bandwidth usage

These information help us to identify traffic flow patterns, analyze bandwidth consumption and isolate and resolve network bottlenecks. The network traffic analysis module also provides reports that can help us secure the networks by identifying potential DoS attacks.

2.2 Overview of DoS and DDoS attacks

DoS and DDoS attacks are a growing concern with serious effects for individual hosts and organizations of all sizes. Video and music streaming services, online games and any number of websites have all at one point been targets of a DDoS attack. So what is DDoS? Distributed Denial of Service (DDoS) attacks are used by criminal enterprises, politically-motivated cyber terrorists and hackers hoping to bring websites down for fun or profit while Denial of Service (DoS) attacks occur when a target machine is flooded with malicious traffic until resources are exhausted and the system goes down.

DoS and DDoS attacks are defined as the "explicit attempt to prevent the legitimate use of a service"[6] or a "Swiss army knife" [26] which can easily exhaust the communication resources of its victim within a short period of time. They do not only target individual websites or other servers at the edge of the network, they subdue the network itself. Attacks have begun to explicitly target the network infrastructure, such as aggregation or core routers and switches in a provider's network. But how do DDoS attacks work?

The goal of DoS attacks is to block network services by limiting the access to the nodes that provide these services in the network. This could be achieved by using up the entire network bandwidth or by consuming the resources available on the service provider nodes such as memory and CPU. The attacker simply sends high volumes of packets to occupy the entire channel bandwidth or breaks down the service by taking the entire processing capacity available on the service provider nodes. There are many reasons why DDoS attacks are so popular among attackers. Some of these include:

- There is an increasing number of connected devices that are poorly managed and allow attackers to take over them and make them part of their botnets.
- Using botnets to perform attacks allows the adversary to incur lower costs for performing a DDoS attack. In addition to lower costs, the ability to use botnets to perform DDoS attacks makes it more challenging to identify the individuals behind the attack.
- Given the relatively low effort and costs involved in performing this type of attack, it is not a problem for the attacker to make it continue until their goals are achieved or they choose to stop.

There are two main methods to launch DDoS attacks in the Internet. The first method is for the attacker to send some malformed packets to the victim to confuse a protocol or an application

running on it and the other method, which is the most common one, involves an attacker trying to disrupt a legitimate user's connectivity by exhausting bandwidth, router processing capacity or network resources [4]. The second method involves essentially network/transport-level flooding attacks which are under consideration of this thesis.

For many service providers it is not easy to understand when they are under the control of DoS or DDoS attack. With a growing number of DDoS attacks being observed across the internet, it is important to understand the risk they pose and the ways to defend against them. While there is no standard way to classify DDoS attacks, one of the systems in use divides them into bandwidth and application attacks:

- **Bandwidth attacks**: These DDoS attacks consume resources (e.g. network bandwidth) with a high volume of packets. Targeted routers and servers can be rendered unavailable to process valid transactions and can fail under the load. The most common form of bandwidth attack is a packet-flooding attack, in which a large number of seemingly legitimate TCP, UDP or ICMP packets are directed to a specific destination. Furthermore attacks like these might also spoof the source address and misrepresent the IP address that supposedly generated the request to prevent identification.
- **Application attacks:** These attacks use the expected behavior of protocols such as TCP and HTTP to the attacker's advantage by tying up computational resources and preventing them from processing transactions or requests.

2.3 DDoS attacks classification based on type and quantity

The number of DDoS attacks is increasing rapidly, which makes protecting against these threats an even bigger priority for all enterprises. In order to be better prepared for DDoS attacks, it is important to understand how they work and examine some of the most widely-used on network and transport layer. DDoS attacks come in a variety of flavors and they are not all the same in the behavior. On a very high level, a DDoS attack can be first divided into the following two categories:

- **Connection-based:** An attack that occurs once a connection between a server and a client has been established via certain standard protocols [44].
- **Connectionless:** An attack that does not require a session to be formally established before a sender (server) can send "data packets" to a receiver (client) [44].

Broadly speaking, they are classified based on the type and quantity of traffic used for the attack and the exploited vulnerability of the target. DDoS attacks are grouped into three categories: Volumetric Attacks, Protocol Attacks and Application Attacks.

1. **Volumetric attacks:** Volumetric attacks are by far the most common and simplest type of DDoS attacks. Also known as "floods attacks". The goal of this type of attacks is to cause congestion and send so much traffic that it overwhelms the bandwidth of the site. Attacks are typically executed using botnets, an army of computers infected with malicious software and controlled as a group by the hacker [44].
2. **Protocol attacks:** Protocol-based attacks primarily focus on exploiting a weakness in Layer 3 or Layer 4 of the OSI layer. The most common example of a protocol-based DDoS attack is the TCP SYN Flood, wherein a succession of TCP SYN requests directed towards a target can overwhelm the target and make it unresponsive [45].
3. **Application attacks:** Application attacks are the trickiest of the DDoS attacks as they are harder to identify and in some cases even mitigate. Application-layer attacks are characterized as the most sophisticated and stealthy attacks because they can be very effective with as few as one attacking machine generating traffic at a low rate. This makes these attacks very difficult to proactively detect with traditional flow-based monitoring solutions. Hackers leveraging application-type attacks are highly skilled and have deep knowledge of the intricate workings of the application or protocol [45].

It is important to note that while most common DDoS attacks broadly fall into these three categories, some attacks can also be a combination. The table below summarises the basic characteristics of the first approach to classify the DDoS attacks.

	Volumetric Attacks	Protocol Attacks	Application Attacks
What is it?	Attacks that use massive amount of traffic saturating the bandwidth of the target. Volumetric attacks are easy to generate by employing simple amplification techniques.	Attacks that render a target in-accessible by exploiting a weakness in the Layer 3 and Layer 4 protocol stack.	Attacks that exploit a weakness in the Layer 7 protocol stack. The most sophisticated of attacks and most challenging to identify/mitigate.
How does it cripple the target?	The sheer quantity of traffic generated by the attack can completely block access to the end-resource (a website or a service). The magnitude of the attack is commonly measured in bits or packets per second.	Protocol attacks consume all the processing capacity of the attacked-target or intermediate critical resources like a firewall causing service disruption.	Application attacks establish a connection with the target and then exhaust the server resources by monopolizing processes and transactions.
Examples	NTP Amplification, DNS Amplification, UDP Flood, TCP Flood	SYN Flood, Ping of Death	HTTP Flood, Attack on DNS Services

Table 2: Classification of DDoS attacks based on type and quantity [45]

All the attacks are utilizing weaknesses in the implementation of the TCP/IP protocols to make the attacked network stop working as intended. These attacks have been mostly launched using TCP, UDP, ICMP and DNS protocol packets. There are four types of attacks in this category:

- **Flooding attacks:** Attackers focus on disrupting legitimate user's connectivity by exhausting victim network's bandwidth (e.g., Spoofed/non-spoofed UDP flood, ICMP flood etc.) A flood attack involves zombies sending large volumes of traffic to a victim system to congest the victim system's network bandwidth with IP traffic. The victim system slows down, crashes or suffers from saturated network bandwidth, preventing access by legitimate users [4].
- **Protocol exploitation flooding attacks:** Attackers exploit specific features or implementation bugs of some of the victim's protocols in order to consume excess amounts of the victim's resources (e.g., TCP SYN flood, TCP SYN-ACK flood etc.) [4].
- **Reflection-based flooding attacks:** Attackers usually send forged requests (e.g., ICMP echo request) instead of direct requests to the reflectors; hence, those reflectors send their replies to the victim and exhaust victim's resources (e.g., Smurf and Fraggle attacks) [4].

- **Amplification-based flooding attacks:** An amplification attack involves the attacker or the zombies sending messages to a broadcast IP address, using this to cause all systems in the subnet reached by the broadcast address to send a reply to the victim system. The broadcast IP address feature is found on most routers; when a sending system specifies a broadcast IP address as the destination address, the routers replicate the packet and send it to all the IP addresses within the broadcast address range. In this attack, the broadcast IP address is used to amplify and reflect the attack traffic, and thus reduce the victim system's bandwidth [34].

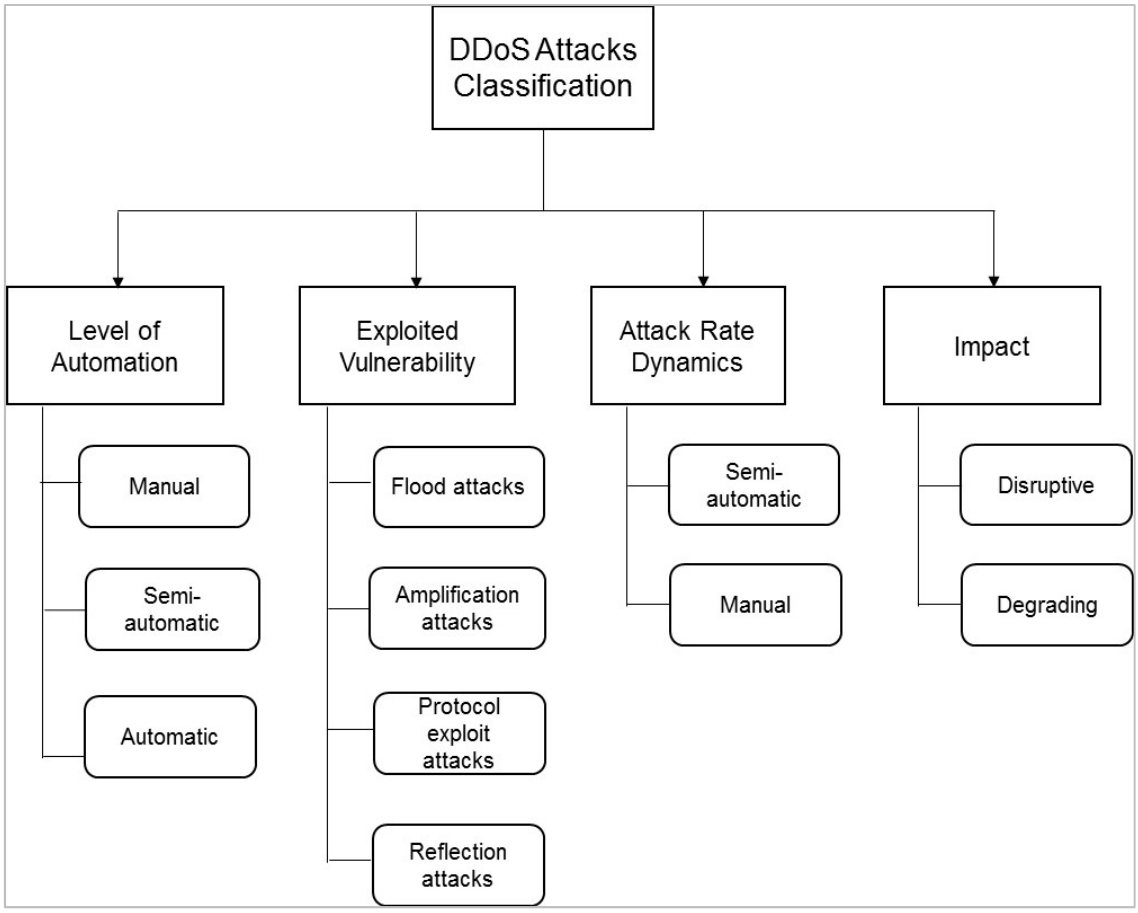


Figure 4: Classification of DDoS Attacks

2.3.1 DDoS attacks classification based on protocol vulnerabilities

In this section, we will discuss about the original design principles of the Internet and their implications in terms of DoS attacks. IP networks were originally designed to provide a best-effort, packet-switched service, where users share all the resources and one user's service can be disturbed by other users' behavior. By occupying most of the shared resources, bandwidth attacks can disrupt service for legitimate users.

One of the design principles is that the Internet should keep the core networks simple and push any complexity into the end hosts. This means that intermediate routers, especially core routers, only need to deliver IP packets without needing to understand services above the network layer. Most changes to the Internet are implemented at the end hosts. This encourages the development of new protocols and new applications. However, this also means that core routers do not have resources to implement sophisticated applications, for example, mandatory

authentication schemes. The lack of authentication at the network layer leads to a serious problem, known as IP spoofing[5]. Without an integrity check for each IP packet, attackers can spoof any field of an IP packet and inject it into the Internet. For the same reason, routers generally do not have packet-tracing functions, for example, keeping all previous connection records. In practice, this cannot be done due to the huge amount of traffic that needs to be stored. Therefore, once an IP packet is received by the victim, there is no way to authenticate whether the packet actually comes from where it claims to be coming from.

Another design principle is that packets can travel on any path between the source and the destination. This makes the Internet extremely robust in comparison to traditional telephone networks. IP packets are forwarded based on their destination address, rather than a predefined path. Many factors, such as delay on a link, can contribute to the changeability of the path a packet is traveling. Hence, the set of IP addresses that appear at a given interface of a router can be highly variable. If a router receives a packet from a source that has not been seen before, then the router has no way of knowing whether this is a spoofed packet, or a legitimate packet that is following a new route as a result of congestion or failure elsewhere in the network. While this flexibility helps make the Internet robust, it also makes IP address authentication difficult.

It is already known that TCP/IP protocols are used as the backbone of the Internet transmission structure. They constitute an important component of the implementation of system. Because of their fundamental importance and necessary usage, these protocols are a prime target for exploitive attacks. When TCP/IP protocols were first being developed for communication over a network, security concerns were minimal for these protocols as access to the network itself was highly restricted.

DDoS attacks can do much more harm than a simple DoS attack. Their size is often tremendous and can take down whole networks or data centers, which makes them difficult to combat. In these types of DDoS attacks, malicious traffic (TCP/UDP) is used to flood the victim. On the basis of TCP/IP Protocol vulnerabilities, DDoS attacks are classified as shown in Figure 5.

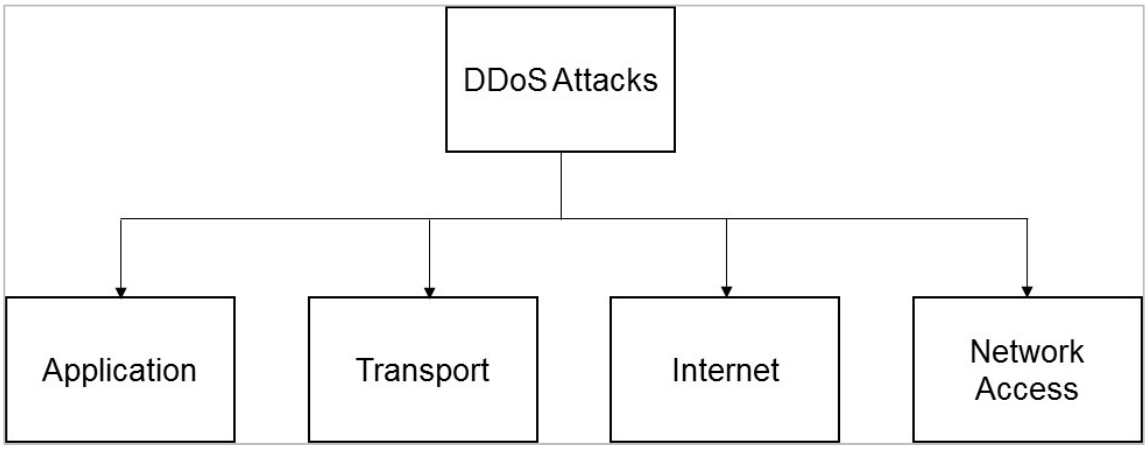


Figure 5: DDoS attacks classification based on Protocol vulnerabilities [23]

Application layer: This type of attack, specifically targets weaknesses in an application or server with the goal of establishing a connection and exhausting its processes and transactions. These sophisticated threats are harder to detect because not many machines are required to attack, generating a low traffic rate that appears to be legitimate [44]. The most common L7 attack types are HTTP/S flooding, Telnet DDoS, SSH, DNS floods.

[5] IP spoofing refers to creating an IP packet containing fake information. IP source address spoofing occurs when an IP packet is generated using a different source IP address than the actual address that is assigned to the source computer.

Transport layer: These types of attacks are usually encompassed of volumetric attacks that aim to devastate the target machine, denying or consuming resources until the server goes offline. In these types of DDoS attacks, malicious traffic (TCP / UDP) is used to flood the victim [23]. The major categories of DDoS attacks under transport layer are SYN flooding, UDP and TCP flooding which will be analyzed in the next section.

Internet layer: This type of attack occurs due to vulnerability in Internet layer protocols of the TCP/IP model. The most common attacks of this category are following:

- **Smurf attack:** In a Smurf attack, an attacker broadcasts a large number of ICMP packets with the victim's spoofed source IP to a network using an IP broadcast address. This causes devices in the network to respond by sending a reply to the source IP address.
- **ICMP flooding attack:** A server is flooded with ICMP echo requests from multiple spoofed IP addresses. As the targeted server processes and replies to these requests, it is eventually overloaded and unable to process valid ICMP echo requests.

Network access layer: These types of attacks exploit the weakness of network layer and its protocols. Following are the major types of DDoS attacks falls under this category [23]:

1. **VLAN hopping:** VLAN hopping is a computer security exploit, a method of attacking networked resources on a Virtual LAN (VLAN).
2. **MAC flooding:** MAC flooding is a method engaged to compromise the security of network switches.
3. **DHCP attack:** Attacker avert hosts from gaining access to the network by refuting them an IP address by overwhelming all of the available IP address in the DHCP Pool.
4. **ARP poison:** Address Resolution Protocol (ARP) poison attacks require the attacker to have access to the victim's LAN. The attacker deludes the hosts of a specific LAN by providing them with wrong MAC addresses for hosts with already-known IP addresses. This can be achieved by the attacker through the following process: The network is monitored for "arp who-has" requests. As soon as such a request is received, the malevolent attacker tries to respond as quickly as possible to the questioning host in order to mislead it for the requested address.

2.3.2 DDoS attacks on the rise

As we already know DDoS attacks are not all the same. While the end result is to consume all of a server or site's resources such that legitimate users are denied service, there is a subtle difference in how these attacks are perpetrated. Attacks on layers 3 and 4 try to exhaust the network resources of the victim, in particular trying to saturate the connection or the ability of network equipment to handle the arrival of so many connections. On layer 7 attacks are not easy to detect because the malicious requests often imitate the ones of legitimate users of the application, which can make very difficult the distinction between the real and malicious traffic.

Very often, the attacks are more sophisticated. Network connections are based on protocols. Normally the TCP and UDP protocols require data to be divided into smaller units called packets, having a header indicating the destination IP. The difficulty of recognizing attacks based on network and transport layer is an existing problem. Early works created taxonomies of network attacks. These taxonomies attempt to collect information on known attack strategies and make a list so that any type of attack can be recognized after it has occurred.

Network layer is responsible for sending individual packets from source to destination. It consists of different kinds of devices which make the core network infrastructure more complicated, so different types of attacks can easily come to the foreground. The majority of DDoS attacks target the network and transport layers. Such type of attacks occur when the amount of data packets and other traffic overloads a network or server and consumes all of its available resources. Some of them are as follows:

TCP SYN flooding is the most common example of network DDoS attack that exploits part of the normal TCP three-way handshake to consume resources on the targeted server and causing network saturation. This happens whenever a client wants to connect to the server. Client requests connection by sending SYN message to the server. The server responds to the client by sending a SYN-ACK message and the connection is established when the client responds with an ACK message. The Figures below show a healthy TCP handshake and a typical TCP SYN flooding attack.

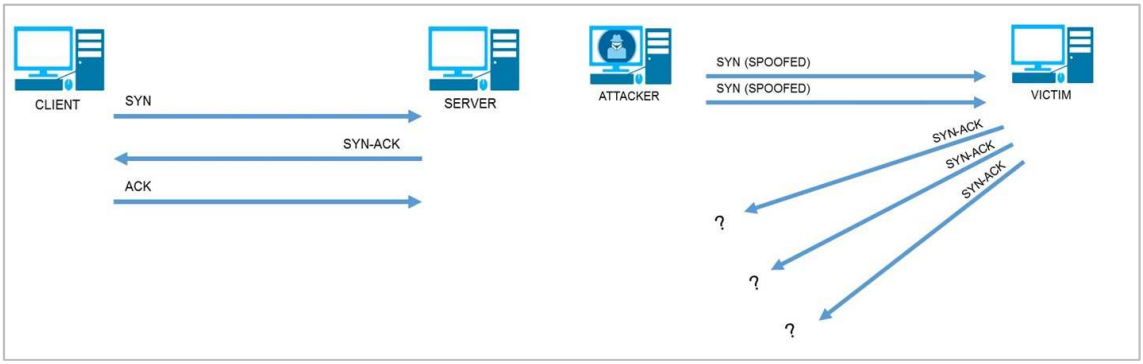


Figure 6a: A healthy TCP handshake Figure 6b: A common TCP SYN flood attack

A TCP-SYN flood attack occurs when attackers flood the target system with SYN requests. The target server handle them in the normal way, dedicating a port, internal memory and usually a CPU process to listening for the SYN-ACK. However, in a TCP-SYN flood attack, the SYN-ACK is never sent and the target server is left hanging, listening for a message that will never come. This hanging persists until a clock timeout occurs and the Transmission Control Block (TCB) is freed. During this hanging period, if enough TCP-SYN requests have flooded, the server will be unable to grant a new TCB to a legitimate request and the server is seen as unresponsive.

During this time, the server cannot close down the connection by sending an RST packet and the connection stays open. Before the connection can time out, another SYN packet will arrive. This leaves an increasingly large number of connections half-open – and indeed SYN flood attacks are also referred to as “half-open” attacks. Eventually, as the server’s connection overflow tables fill, service to legitimate clients will be denied and the server may even malfunction or crash.

ICMP flooding is a common DDoS attack in which an attacker takes down a victim's computer or a network by overwhelming it with ICMP echo requests, also known as pings. The typical goal of attack is to flood the victim's network with request packets, knowing that the network will respond with an equal number of reply packets. Normally, ping requests are used to test the connectivity of two computers by measuring the Round Trip Time (RRT) from when an ICMP echo request is sent to when an ICMP echo reply is received. During an attack, however, they are used to overload a target network with data packets. The success of the attack is dependent on attackers knowing the IP address of the target. By this way, attacks can be broken down into three categories [56].

- **A targeted local disclosed ping flood** targets a single computer on a local network. An attacker needs to have physical access to the computer in order to discover its IP address. A successful attack would result in the target computer being taken down.
- **A router disclosed ping flood** targets routers in order to disrupt communications between computers on a network. It is reliant on the attacker knowing the internal IP address of a local router. A successful attack would result in all computers connected to the router being taken down.
- **A blind ping flood** involves using an external program to uncover the IP address of the target computer or router before executing an attack.

Smurf flood attacks are another ICMP based attack type. However, is an indirect type of attack in which the server systems are flooded with ICMP echo replies rather than ICMP echo requests. This is done when the attacker spoofs the IP source address in its initial ICMP echo request to be the IP address of its intended attack victim. The attacker can then broadcast these spoofed echo request messages to whole networks/sub-networks of legitimate devices. These legitimate devices then generate and send ICMP echo reply messages to the spoofed source IP address, which is the address of the intended victim, and an attack occurs. In this way a DDoS attack can actually originate from a single root source.

UDP flooding attack is triggered by sending a large number of UDP packets to random ports on the victim's system. The system will notice that no application listens at that port and reply with an ICMP destination unreachable packet. Subsequently, if a large number of UDP packets are sent, the victim will be forced to send numerous ICMP packets. In most cases, these attacks are accomplished by spoofing the attacker's source IP address. Most modern operating systems now limit the rate at which ICMP responses are sent, minimizing the impact and mitigating this type of DDoS attack.

Application layer DDoS attacks are also on the rise and their goal is to exhaust the resources of the web pages which are generated on the server side and delivered in response to HTTP requests. These attacks, often called layer 7 attacks, target not only applications but also the network and bandwidth. The most common types of application layer DDoS attacks include those targeting DNS services, HTTP and HTTPS.

In an application layer, attackers can send malicious traffic either to a UDP socket or through a TCP connection, using both network bandwidth and resources on the victim network service. A common form of application layer DDoS is a reflective DNS amplification attack. During this attack, a malicious client sends a DNS query to an open DNS resolver with a spoofed source IP matching the victim server. The DNS resolver then sends (reflects) a response, often several times larger than the query, to the victim server due to the spoofed address.

In an HTTP flood DDoS attack, the attacker exploits seemingly-legitimate HTTP GET or POST requests to attack a web server or application. HTTP floods do not use malformed packets, spoofing or reflection techniques and require less bandwidth than other attacks to bring down the targeted site or server. The attack is most effective when it forces the server or application to allocate the maximum resources possible in response to each single request [57].

Another attack which operates by utilizing partial HTTP requests is the Slowloris which functions by opening connections to a targeted web server and then keeping those connections open as long as it can.

Slowloris is a highly-targeted attack, enabling one web server to take down another server, without affecting other services or ports on the target network. This can be true by holding as many connections to the target web server open for as long as possible. It accomplishes this by creating connections to the target server but sending only a partial request. Slowloris constantly sends more HTTP headers, but never completes a request. The targeted server keeps each of these false connections open. This eventually overflows the maximum concurrent connection pool, and leads to denial of additional connections from legitimate clients [57]. The attack occurs in four steps [58]:

1. The attacker opens multiple connections to the targeted server by sending multiple HTTP request headers.
2. The target opens a thread for each incoming request, with the intent of closing the thread once the connection is completed. In order to be efficient, if a connection takes too long, the server will timeout the exceedingly long connection, freeing the thread up for the next request.
3. To prevent the target from timing out the connections, the attacker periodically sends partial request headers to the target in order to keep the request alive.
4. The targeted server is never able to release any of the open partial connections while waiting for the termination of the request. Once all available threads are in use, the server will be unable to respond to additional requests made from regular traffic, resulting in denial-of-service.

The table below, summarises the common attacks which are detected at the network, transport and application layer, their impact on high-speed networks as well as some ways to mitigate the attacks.

Attack Possibilities by OSI Layer						
OSI Layer	**Protocol Data Unit (PDU)**	**Layer Description**	**Protocols**	**Name of Attack**	**Potential Impact of DoS Attack**	**Mitigation Options for Attack Type**
Network (3)	Packet	Dedicated to routing and switching information to different networks. LANs or internetworks	Uses the protocols IP, ICMP, ARP, RIP and uses routers as its device	ICMP Echo Request Flood, IP Packet Fragment Attack, SMURF, IGMP Flood, Ping of Death	Can affect available network bandwidth and impose extra load on the firewall	Rate-limit ICMP traffic and prevent the attack from impacting bandwidth and firewall performance
Transport (4)	Segment	Ensures error-free transmission between hosts: manages transmission of messages from layers 1 through 3	Uses the protocols TCP & UDP	TCP SYN Flood, TCP Spoofed SYN Flood, TCP SYN ACK Reflection Flood, TCP ACK Flood, TCP Fragmented Attack, UDP Flood, UDP Fragment Flood	Reach bandwidth or connection limits of hosts or networking equipment	DDoS attack blocking, commonly referred to as blackholing, is a method typically used by ISPs to stop a DDoS attack on one of its customers. This approach to block DDoS attacks makes the site in question completely inaccessible to all traffic, both malicious attack traffic and legitimate user traffic.

Application (7)	Data	Message and packet creation begins. DB access is on this level. End-user protocols such as FTP, SMTP, Telnet, and RAS work at this layer	Uses the Protocols FTP, HTTP, POP3, SMTP and its device is the Gateway	Distributed DNS Amplification Attack, DNS Flood, HTTP(S) GET/POST Flood, DDoS DNS	Reach resource limits of services Resource starvation	Application monitoring is the practice of monitoring software applications using a dedicated set of algorithms, technologies, and approaches to detect zero day and application layer. Once identified these attacks can be stopped and traced back to a specific source more easily than other types of DDoS attacks

Table 3: Attacks possibilities based on network/transport layer [47]

2.4 DDoS in the IoT: Mirai and other botnets

Internet of Things is the next big evolutionary step in the world of Internet. Referring to the IoT is defined as "the interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data" [48]. The increasing popularity of the IoT has made “smart” devices a powerful amplifying platform for cyber-attacks. It is common for these devices to have poor security standards such that their remote administration ports are publicly accessible to brute force, the ports are protected with vendor default passwords (e.g. admin/1234) and they don’t have any antivirus solution in place to prevent malware infections.

Securing connected devices becomes more and more important due to the fact that actual devices have only weak security features by design. The attractiveness for attacking IoT devices is simple, the devices are permanently online and they have no antivirus protection or malware scanners or protection mechanisms. These basic vulnerabilities expose IoT devices to the most unsophisticated attacks such as spoofing, simple intrusion attacks etc.

In October 2016, the strongest attack in the short history of the IoT appeared in the Internet scene. The Mirai malware was used to perform a strong DDOS attack against a significant DNS-provider called Dyn. This attack also had significant impact and side effects on other large sites such as OVH, GitHub and Amazon which were temporarily unavailable [48]. Before analyzing further the Mirai botnet, let’s see what it is and how it works.

Mirai is a botnet -a malicious software application that is designed to gain unauthorized access to Linux- powered devices and conscript them into a distributed infrastructure of clients. Once

enlisted, these machines have the capability to perform a variety of DoS attacks against a target dictated by the attacker [49]. When people are referring to Mirai, they often talk about the emerging threat caused by IoT devices. This doesn't account for 100 percent of Mirai activity but certainly there are some aspects that make these "smart" devices attractive to attackers.

- Most people have a personal computer but they also have multiple Internet-enabled devices. IoT devices have historically not been very secure.
- Consumers are less likely to secure their IoT devices than their personal computer.
- With the proliferation of the IoT, the pool of possible botnet nodes is growing and become more and more attractive to attackers.
- Unlike personal computers, IoT platforms are generally identical.

In order to gain access to IoT devices, Mirai does not exploit any software vulnerabilities but it tries to guess Telnet login credentials for computers accessible via Telnet from the Internet. Some of the username and password combinations like "admin / admin" and "root / root" which used from these devices, are really bad choices and simply accessible. In particular, Mirai sends TCP SYN packets towards Telnet ports 23, 2323 or 103 and sets the TCP initial sequence number equal to the destination IP of the targeted host.

DDoS attacks are usually high-traffic events, measured in Gbps (gigabits per second) or PPS (packets per second). 20-40 Gbps is usually enough to shut down most network infrastructures. The size of the Mirai botnet isn't really what's remarkable about it -the latter attack reached a peak volume over 665 Gpbs and 1.2 Tbps [50]- there are many other botnets operating now that are several times its size. It's the fact that Mirai has mainly infected embedded devices such as DVRs and CCTV cameras that has drawn notice, along with the fact that the botnet has been part of two of the larger DDoS attacks ever seen. The malware scans the Internet for devices with Telnet running on an open port and using default credentials, and then connects and installs itself on the device. Each new infected device then starts the scanning process again [51].

Mirai is the most popular IoT botnet that was involved in large scale attacks but not the only one. Leet botnet and the Amnesia botnet have also been detected by security firms while targeting IoT devices.

Leet is being regarded as the winner of the most powerful DDoS attack of 2016 with a humongous speed of 650 Gbps, noticed recently by Imperva[6] network [52]. Leet's name comes from a signature within the packets; In the TCP header options of these packets, the values were arranged so they would spell '1337'. To the uninitiated, this is leetspeak for 'leet', or 'elite' [53].

The attack came in two different waves, both which attacked Imperva's network after the attacker was unable to determine the target's real IP address, which had been hidden behind the company's mesh of proxy servers. The first wave peaked at 400 Gbps and lasted only 20 minutes. But the attacker came back five minutes later with a larger DDoS cannon that thrusted more than 650 Gbps of junk traffic at the Imperva network. This second attack lasted only 17 minutes, as the attacker realized it couldn't bring down its target. Compared to the attacks on OVH, KrebsOnSecurity, and Dyn, which lasted days, this was insignificant. Furthermore, researchers found three main differences in the way the botnets operated [53].

1. The Mirai malware and its botnets aren't built and don't feature the technical capabilities to launch large SYN attacks, as this attack was.
2. Junk traffic packets sent out by Mirai botnets are all hardcoded with several TCP options that were not present in the packets observed in this attack.
3. The content of each junk traffic packet sent out during a Mirai DDoS attack is made up of random-generated strings. For this attack, the Leet botnet had taken content from actual system files, embedded the content in the DDoS attack's TCP packets and sent it out towards Imperva's network.

[6] https://www.imperva.com/

In March 2016, the Amnesia botnet targets an unpatched remote code execution vulnerability that was detected in Digital Video Recorder (DVR) devices which made by TVT Digital and branded by over 70 vendors worldwide. Amnesia exploits this remote code execution vulnerability by scanning for locating and attacking vulnerable systems. A successful attack results in Amnesia gaining full control of the device. Attackers could potentially harness the Amnesia botnet to launch broad DDoS attacks similar to the Mirai botnet attacks.

In addition, the Amnesia malware is the first Linux malware to adopt virtual machine evasion techniques to defeat malware analysis sandboxes. Virtual machine evasion techniques are more commonly associated with Microsoft Windows and Google Android malware. Similar to those, Amnesia tries to detect whether it's running in a VirtualBox, VMware or QEMU based virtual machine and if it detects those environments it will wipe the virtualized Linux system by deleting all the files in file system. This affects not only Linux malware analysis sandboxes but also some QEMU based Linux servers on VPS or on public cloud [54].

Much of the responsibility for DDoS attacks often belongs to users who practice poor security behaviors and system administrators who fail to deploy adequate safeguards. In the case of IoT botnets, a big mistake comes from vendors who distribute products with weak security, including default credentials and remote access capabilities. So vendors should provide automated security updates that would solve the problem. Solutions such as changing passwords are not unrealistic in the world of IoT devices. We need to enforce security practices and standards for IoT devices and distributors.

Chapter 3: Architecture and design

3.1 In a nutshell

Before DDoS attacks can be mitigated, they first have to be detected. Many papers have been written about the detection of DDoS attacks. A large number of detection mechanisms have been proposed and many techniques trying to identify DDoS attacks near the victim, near the attack sources and within transit networks.

DDoS attacks are notoriously difficult to detect on a timely basis and defend against. Most of the organizations use multiple threat detection mechanisms and tools for detecting these attacks. The detection of DDoS attacks is vital for the security management of edge networks so ISPs use NetFlow analyzers, firewall logs and SNMP-based tools which are the most successful and commonly deployed threat detection mechanisms. However there are cases where these approaches take a long time to implement and any change in the infrastructure makes the baseline obsolete and results in False Positives.

In this chapter, we present a summary of existing DDoS attacks detection methods. As DDoS attacks become one of the most threatening security issues, the need to detect this type of attack is increasing. Therefore, there is not a straightforward approach or method or unique solution to filter or block the offending traffic.

3.2 A honeynet architecture for detecting network attacks

This chapter articulates the general challenges associated with designing a system capable of detecting DDoS attacks. Specifically, we demonstrate all the installation steps and describe all the necessary configurations and resources needed to carry out the experimental analysis. In order to protect the network systems from attackers we must try to find methods to stop at least some part of the threats described in previous chapters.

An efficient way to monitor and infer threat activities online is to collect information from an architecture based on honeynet technology using detection capabilities. So that raw network pcap data will be investigated to gather statistical information about the attacks. Before starting to describe how we set up the network infrastructure for the honeynet, we need to mention the primary challenges which are identified as essential tasks for a general system in order to protect a network from DDoS attacks.

Sufficiently, monitoring network behavior is the first challenge to be met. As networks become more complex and more distributed, no single network device has a global network view and monitoring becomes infeasible. Furthermore, to effectively secure a system from attack, the security system must be able to accurately detect an attack before damage is done. Due to the imperfect nature of detection, an amount of error will invariably exist (for example to detect an event that doesn't occur -False Positive[7] type).

An effective system must be able to take action against an identified threat. The action taken must be able to mitigate the harmful effects of the attack and should provide some sort of attack memory should a similar attack occur in the future. Given a distributed environment, the action taken should include a broadcast of the attack signature to potentially affected network devices such that other network devices may stop or mitigate the attack. The attack mitigation technique should maximize effect on the targeted threat while minimizing the disruption to normal traffic.

All these core challenges should be faced up by developing a honeynet, a common way of collecting malware and/or any suspicious traffic from the Internet. The main idea for the practical part of this thesis was to create a virtual Linux host, running Ubuntu, connected to the Provider Edge Router (Figure 8) in order to receive the packet traffic for specific unused IP

[7] False positive type: Authorized users seem to be attackers (problem in anomaly detection systems). Otherwise false negative where attackers seem to be authorized users (problem in signature based IDS).

ranges. Figure 7 shows the various modules and techniques used of the proposed system as well as the description of all these modules:

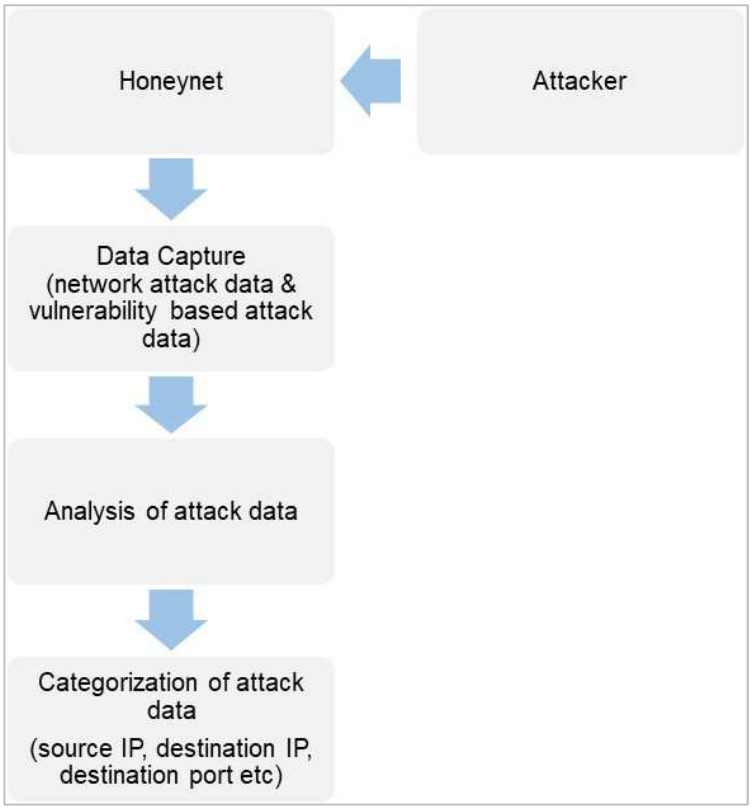


Figure 7: Various modules of the proposed design

It's often more useful to capture packets using tcpdump, one of the most popular network packet analyzers, rather than Wireshark. The major function of tcpdump is to monitor packets on the attached network and dump headers and payloads of packets to a human-readable format. In order to monitor the network traffic, tcpdump sets the network interface in the promiscuous mode to capture all the packets on the attached network. It uses packet capture library (libpcap) to retrieve packets from the network interface.

Tcpdump is a simple and easy to use tool. It supports many command line options for flexibility and ease of convenience. The output generated using tcpdump depends on the options used by the user and allows to extract particular kinds of network traffic. Furthermore, tcpdump can capture TCP, ICMP, UDP and ARP packets and give us the opportunity to identify the source/destination IP address, source/destination port and the type of packet for all network and transmission protocols. One more important feature of tcpdump is to allow the user to filter packets and select only those that match a certain rule or criteria.

The next phase is the analysis phase. A key part of this process is incident classification, which involves understanding the type of attack and the damage it is causing. It is important to perform the analysis with as little impact as possible on business functions. Although the step of detection phase could be done with tcpdump, we used Wireshark –a malware analysis tool- because of the GUI access. The results of this analysis will help to determine the most appropriate reaction techniques for the specific incident.

The open source software Wireshark has been built over the pcap tool including functions for network traffic analysis. It is widely used in network troubleshooting, analysis and protocol development by network professionals. Wireshark is a convenient user interface program that allows direct and easy access to the network packets, captures live traffic and accepts wide range of protocols, such as TCP, UDP, ARP, HTTP and etc.

It also displays packets with source and destination IP addresses, protocol, frame size and additional information about each packet. Headers are decoded and can be read by the user. This software makes easy the way to create custom Wireshark profiles for specific tasks relating to network protocol or packet analysis and troubleshooting (i.e. network scanning detection, unauthorized or anomalous network traffic identification).

Wireshark cannot alter the flow of packets within the network but can only view the traffic activity in any network. It uses two different types of filters. The Capture filter is used to select the data to be recorded in the logs and the Display filter which is used to search for information from the captured logs.

In this regard, Wireshark can be used in identifying and categorizing (last phase of detection) various types of attack signatures. Captured packets can reveal the signatures of attacks and this information can enable the users to recover the systems from damages caused by the attackers. Network attacks can be mostly identified by observing the incoming and outgoing traffic, because unusual behavior is resulted from suspicious patterns of packets. The contributions of this thesis are as follows:

- we show that a packet analyzer like Wireshark can be leveraged to identify certain types of network attacks that result in unusual activities and
- we present case studies for typical network attacks by using Wireshark

So far we described the power of tcpdump and Wireshark to get a deeper look on how we can use them in order to capture and analyze a network traffic. NetFlow analyzer is the most commonly used tool and remains the most effective way of identifying possible anomalies or detecting malware activities. It provides real-time network monitoring and classification of well-known DDoS attacks, viruses and worms.

Each packet that is forwarded within a NetFlow enabled switch or router is examined for a set of IP packet attributes. These attributes are used as the IP packet's identity of the packet to determine if the packet is unique or similar to other packets. The IP packet attributes that are inspected by NetFlow are the following:

- IP source address
- IP destination address
- Source port
- Destination port
- Protocol type (TCP/UDP/ICMP)
- Class of service priority (value that can be used by QOS router or switch interface)

All packets with the same source and destination IP address, source and destination ports, protocol interface and class of service are grouped into a flow. A NetFlow report does not provide any information about the content of the packets moving through the network but only information based on the volume of the traffic that was exchanged between the endpoints. Typically is configured on the network edges to provide insight into the traffic getting into the network.

NetFlow can be used to track application and network usage and is often used by ISPs to calculate how much network resources each customer is using. It can be used to get a quick glimpse of how changes made to the network infrastructure impact the network. In addition to this, NetFlow can also be used to detect network anomalies and security vulnerabilities. If an Internet worm is communicating on a specific port or is communicating in a certain pattern, one can often with high certainty detect the machines infected with the worm. This makes NetFlow a powerful tool when protecting the local network.

Due to detailed information contained within NetFlow packets, admins and engineers can successfully utilize a NetFlow analyzer to identify bandwidth usage issues, network bottlenecks and application utilization of network resources. The most common reasons that lead ISPs to use NetFlow analyzers to their network are the below:

1. Network Managers are able to acquire a great deal of information with a NetFlow analyzer and compile detailed reports which help them to develop future projects and investments.

2. NetFlow analyzers provide comprehensive visibility into networks, giving engineers the ability to account for bandwidth usage down to the user and or IP level.
3. Software vendors are able to capture NetFlow packets with proprietary systems and use it to present network information with the means of their software suite's reporting and analysis tools. A NetFlow analyzer extends the ability of third party vendors to acquire and analyze network data efficiently.

For the implementation of the idea above, we routed towards the virtual host unused IP ranges. That means the virtual host is responsible for collecting and analyzing network flows from packets having destination addresses these IPs.

A very important step before starting looking for malicious traffic is the routing configuration on the PE router, which has already been connected to the virtual host (1GbE interface connectivity) in order to advertise all these IPv4 ranges to the rest of the network via IGP[8] (OSPF[9] and IS-IS[10]) This was achieved by means of static routes on the PE router side, which were then redistributed to the rest of the network via IGP.

Before putting our honeynet system into action in the real network environment, its functionality and usefulness were tested through packet sniffing. Those included the open source emulator PuTTY which can be used as a client for SSH and Wireshark, an open source network packet analyzer used for network analysis and troubleshooting. After conducting a variety of several tests including pinging, tracerouting, scanning and connection attempts, it was verified that the virtual host was perceived as real server. Figure 8 illustrates an overview of the implemented honeynet topology.

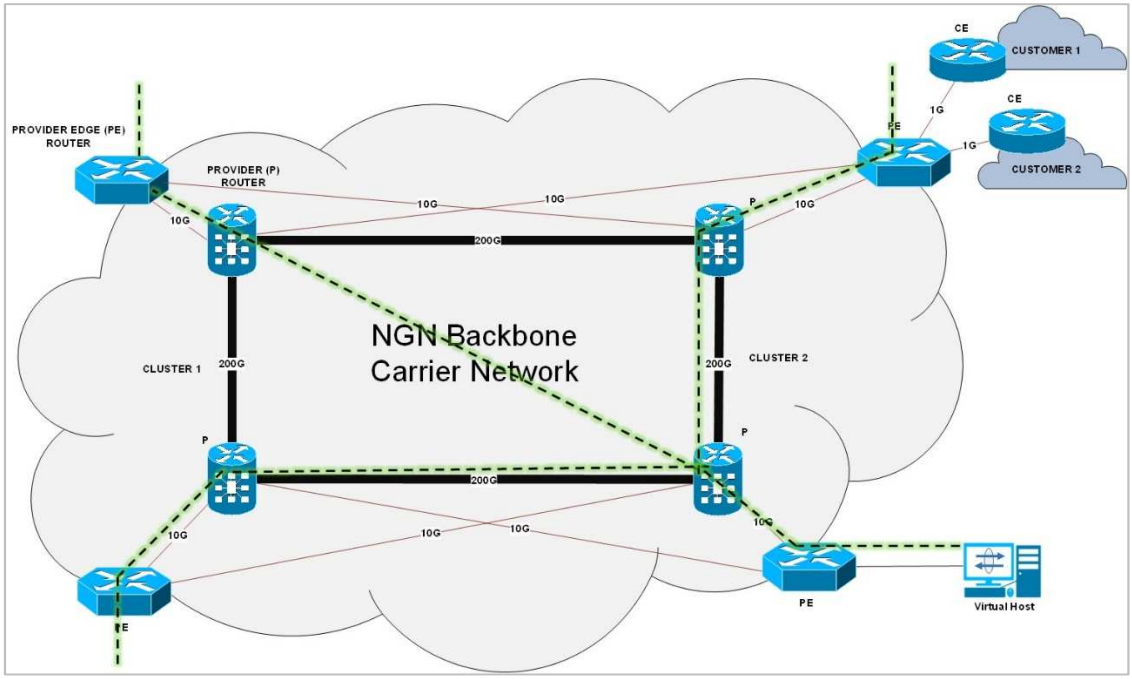


Figure 8: Implemented honeynet topology

[8] Interior Gateway Protocol is a type of protocol used for exchanging routing information between gateways (commonly routers) within an autonomous system. This routing information can then be used to route network-layer protocols like IP. https://en.wikipedia.org/wiki/Interior_gateway_protocol

[9] Open Shortest Path First is an IGP for routing IP packets solely within a single routing domain, such as an autonomous system. It gathers link state information from available routers and constructs a topology map of the network. https://en.wikipedia.org/wiki/Open_Shortest_Path_First

[10] Intermediate System to Intermediate System (IS-IS) is a routing protocol designed to move information efficiently within a computer network, a group of physically connected computers or similar devices. It accomplishes this by determining the best route for data through a packet-switched network. https://en.wikipedia.org/wiki/IS-IS

Honeynet has been up and running for a period of approximately four months. From the first day of our honeynet deployment, we started receiving a great amount of network probes targeting our network. Until the last day of our study, thousands of network access attempts were recorded, constituting a solid ground for analyzing and assessing the results of our study.

An important area of Internet research focuses on monitoring of network systems. Particularly in security-related fields, allocated but unused IP addresses (Dark IP addresses) are considered a valuable resource which can enable the collection and analysis of attack traffic. By deploying Intrusion Detection Systems, monitoring systems or honeynets, researchers can collect vast amounts of traffic data.

In a nutshell, Dark IP space is a range or ranges of IP addresses which are routed, allocated (by IANA) but not in use. These addresses are of value because any packet sent to them is a possible attack and subject for analysis. A Dark IP address range is routable on the Internet and can be used as a method for detecting network security-related incidents.

It is observed that the number of anomalous packets generated by worms or DDoS attacks is dramatically increasing. These packets typically aim to exploit services vulnerabilities. In order to minimize these threats, it is necessary to have a system in place that has the ability to detect zero-day attacks and block them. Today many organizations usually use honeynets to analyze attacks and vulnerabilities and learn more about the techniques and motivations of the attackers. A honeynet is one of the security resources which has value in being probed, attacked and compromised. There is no production value for honeypot but it can collect information about attacking. This information is valuable for the field of network security analysis.

Honeynets are assigned one or more unused IP addresses to monitor and usually listen on several UDP/TCP ports for incoming traffic. The listening UDP/TCP ports can be assigned fully functional services (e.g. HTTP, SSH, etc.) to provide some interactivity and thus permit the capture of complex attack attempts or just be used to capture network packets. There are several ways to deploy honeynets on live networks, the main goal being to maximize the collected information while limiting detection by attackers.

In order to implement the experimental scenario, it is very important to accept the following assumptions:

1. Allow all packets from external networks to the Dark internal IP addresses.
2. Deny all packets sent from the Dark internal IP addresses. In our case, server only accepts the traffic from external networks and doesn't answer to the requests.
3. Record a log file for all incoming packets from external networks in order to be analyzed from Wireshark.

All these rules have already been configured at the router side which is connected with the server via Ethernet interface. The anomaly packets which captured by Dark IP networks are divided into three categories:

- Port scanning or Host scanning are well-known cracker behaviors. They try to confirm the presence of hosts, searching for open ports for their attacks. Scanning is not limited to human crackers. A worm may send attack packets that exploit a bug in an operating system or in some network software. Their goal is to spread infected machines on the network.
- A backscatter of DDoS packets occurs when a host or a server is attacked and the IP address is spoofed. In most of the cases, a backscatter packet has a TCP flag of SYN/ACK in response to an attack of TCP SYN. When a Dark IP address observes this kind of packet, it indicates that a DDoS attack is being carried out somewhere on the Internet.
- The packets due to improper configuration are usually caused by human mistakes. The security risk of these packets is low. However, these packets are sometimes detected erroneously as DoS or scanning packets.

3.3 Network transit security policy

Security threats are an ongoing global challenge for IT professional experts and consumers. While it is important for customers to take adequate measures to protect themselves, ISPs are primed to take the lead. Cyber security is becoming an increasing concern for financial and governmental institutions, but it's always been a top priority for service providers. In a survey done by the Internet Services Providers Association, it was reported that over 90% of ISPs come under some form of attack, and 85% of those surveyed said it was the responsibility of ISPs to take a "proactive role in cyber security" [33].

In order for ISP's customers to feel confident, they need to know if their data are secure. On the other hand service providers need to ensure that they share information with relevant organizations and authorized users. This requires communication processes and policies. As we mentioned in previous chapter, one of the biggest cyber threats an ISP faces are DDoS attacks, where end-users are denied access to Internet applications. There are several reasons for ISPs to make them protect their customers from DDos attacks:

1. DDoS attacks are on the increase and can hurt their customers, especially if the attack lasts hours.
2. They become more complex: The availability of new tools means even relatively basic hackers can launch sophisticated attacks against service providers.
3. DDoS attacks are harder to detect: Hackers are constantly revising their techniques and many are working with the support of various governments and terrorist organizations.
4. The Internet of Things (IoT) or smart devices are escalating at an unforeseen rate.

Network policy deals with general network issues, subdivided into high-level and low-level policy. High-level policy deals with "why" and low-level policy deals with "how" to place administrative controls on the network. On the high-level policy, companies normally stipulate which applications can be used on the network, which applications can communicate with the outside world, which applications can talk to local clients from the outside and which conditions must be met for exceptions to be allowed. Low-level policy details specifically which commands will be used on the firewalls to actually lock them down.

This section describes the security policy of a high-speed transit network and what rules and procedures are been driven from ISPs when hosts are being compromised. As a backbone network, the policy is determined by the basic properties of an IP network, where control is at the edge routers. Hosts determine when and where to send packets and initiate traffic flows. They may become compromised and send large amounts of traffic to other hosts or they may compromise additional hosts, leading to potential chaos on the network. The fundamental security problems of an IP network therefore lie at the edge, in the hosts attached to the network and not in the network itself.

The backbone network has the means to apply at least some control. It is possible for the network to block traffic on particular ports or to block all traffic from particular IP addresses. In today's high performance internetworks, organizations need to implement packet forwarding and routing according to their own defined policies in a way that goes beyond traditional routing protocol concerns.

By using policy-based routing, customers can implement policies that selectively cause packets to take different paths. The benefits that can be achieved by implementing policy-based routing in the networks include:

- ISPs and other organizations can use policy-based routing to route traffic originating from different groups of users through different Internet connections across the policy routers.
- Organizations can provide Quality of Service (QoS)[11] to differentiated traffic by setting the precedence or Type of Service (ToS)[12] values in the IP packet headers at the

[11] Refers to a network's ability to achieve maximum bandwidth and deal with other network performance elements like latency, error rate and uptime. Also involves controlling and managing network resources by setting priorities (high, low,

periphery of the network and leveraging queuing mechanisms to prioritize traffic in the core or backbone of the network.

- In addition to the dynamic load sharing capabilities network managers can implement policies to distribute traffic among multiple paths based on the traffic characteristics.
- Policy-based routing allows the network administrator to classify traffic using ACLs and then set the IP precedence or ToS values, thereby tagging the packets with the defined classification.

With some types of DoS attacks, there's not much we can do to stop the flow of the attack, especially in DDoS attacks in which the attacker is spoofing the source IP address. In this situation, ISP's first concern should be limiting the impact of the attack on the network, which can be done with rate limiting. Rate limiting is a defense against DDoS attacks and is able to assign a bandwidth restriction to a category of traffic, such as ICMP, UDP etc. A rule-based policy limits the incoming traffic from specific sources and sets the priority of traffic to control the volume of traffic for a specific period (war time period where the rate of attacks is increasing).

Port-based rate limiting is used from ISPs to specify the maximum number of bytes a given port can send or receive. The port drops bytes that exceed the limit they specify. This limit is determined in peace time period of the network, where ISPs could find the normal rate of traffic on a specific port using NetFlow analyzer. With this policy, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the router blocks all traffic until the traffic rate drops below the rising suppression level.

At the end of this section, it is important to say that all customers need to be bound by an Acceptable Use Policy (AUP)[13]. Network abusers typically target hosts with no existing or poorly written AUPs to further threaten the service provider when they try to take action against them, making it highly important that AUPs are reviewed and updated regularly. The AUP contract determines the guidelines that customers must obey when connected to the network, including what kind of traffic is permitted and what sanctions will be imposed if the AUP is violated. Any violations of the AUP will show up on abuse reports, enabling abuse desk managers to identify categories of abuse that are frequently seen so that they can update the AUP. Especially for big customers there is a contract between them and ISP which states that in case of a security attack, customer is submitted in blackholing situation for forty minutes until the attack is being solved.

---

medium) for specific types of data on the network. QoS is exclusively applied to network traffic generated for video on demand, IPTV, VoIP, streaming media and online gaming.

[12] Is a field in the IPv4 header and has various purposes. The ToS field could specify a datagram's priority and request a route for low-delay, high-throughput or highly-reliable service. Based on these ToS values, a packet would be placed in a prioritized outgoing queue or take a route with appropriate latency, throughput or reliability.

[13] An acceptable use policy (AUP) is a document stipulating constraints and practices that a user must agree to for access to a corporate network or the Internet.

Chapter 4: Experimental analysis

4.1 Overview of the cyber-attacks chain

This chapter describes the analysis of the attack patterns found on the honeynet and presents an overall statistical analysis of the results gathered from the combination of tcpdump and Wireshark packet analyzers.

When dealing with attacks against large-scale networks, many people might not realize that the actual infection is only one part of a chain of events leading up to a network breach. In this section, we're going to analyze the attack chain before presenting the experimental results of honeynet.

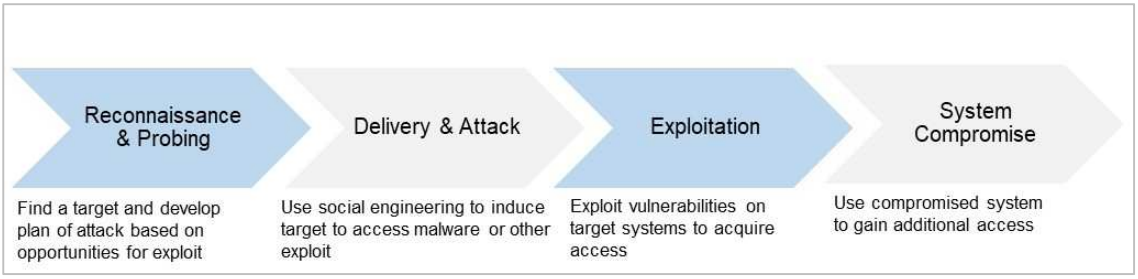


Figure 9: Cyber-attacks chain reaction [https://www.alienvault.com/blogs/security-essentials/defend-like-an-attacker-applying-the-cyber-kill-chain]

The objective of reconnaissance is the attackers to get information about the architecture of the target organization and gather as much as possible valuable data. From the outside, they learn about the resources of networks in order to determine whether it is worth the effort. Ideally, they want a target that is relatively unguarded and with valuable data.

All the information that collected in the early stages itself is important. The relevance of information may vary with respect to target entity. However, the following can be considered as the best information.

- Open ports and services
- IP blocks
- Domain names
- OS and server information

Network reconnaissance relates to the act of gathering information about the target's network infrastructure, the devices that reside on the network, the platform used by such devices, and the ports opened on them, to ultimately come up with a brief network diagram of devices and then plan the attack accordingly. Network scanning activities can be as follows:

- Scanning for live machines
- Port scans
- Detecting for additional IP protocols

An attacker would want to map out the live machines on the network rather than performing any activity with an assumption that all the machines are live. The ping sweep is the most common technique to ping an IP address in order to identify whether it is alive or not. However, in case the network supports ICMP-based traffic, we can detect this attack by looking for large number of ping requests going to a range of IP addresses on your network. This is highlighted in the following screenshot, using the Wireshark filter: icmp.type == 8 || icmp.type == 0 on a specific target 94.65.215.0/24, where icmp.type 8=ECHO Request and icmp.type 0=ECHO Reply.

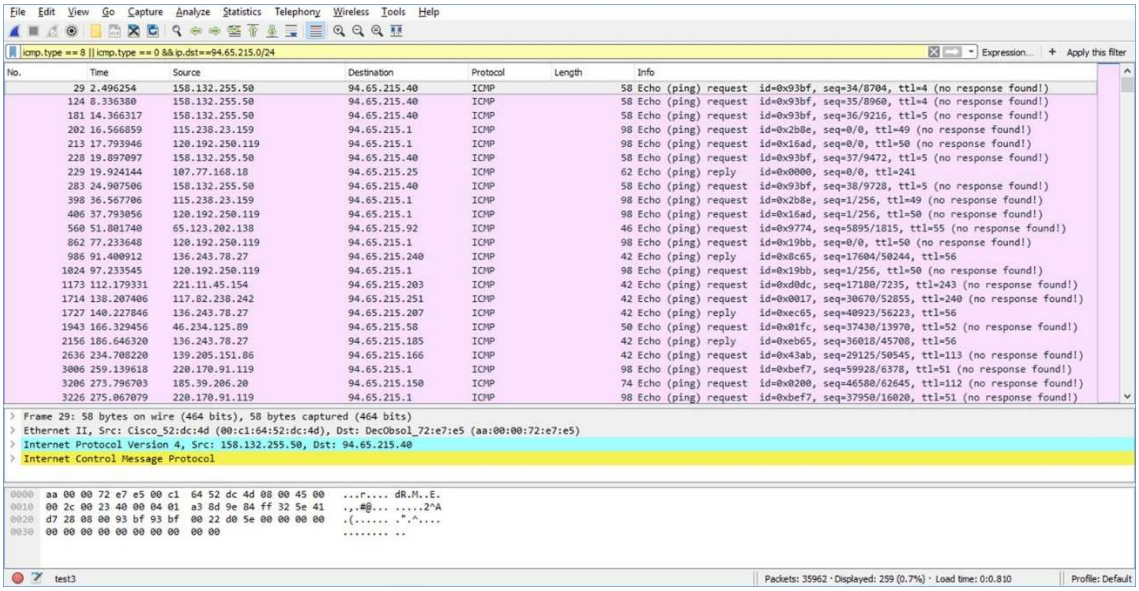


Figure 10: Ping sweep example (screenshot from Wireshark)

Once available hosts on a network have been found via networking scanning, port scanning can be used to discover the services in use on specific ports. As previously mentioned, TCP and UDP are frequently the protocols used in port scanning. There are several methods of performing TCP scans used by attackers that can be detected using Wireshark.

In a TCP connect scan, a client/attacker sends a SYN packet to the server/victim on a range of port numbers or on a specific port. For the ports that respond to SYN/ACK, the client completes the 3-way handshake by sending an ACK and then terminates the connection by sending an RST to the server/victim, while the ports that are closed reply with RST/ACK packets to the SYN sent by the client/attacker. The following screenshots display an example of SYN scan where a possible attacker with IP address 119.42.83.139 sends SYN packets to the target 94.65.215.0/24 trying to scan all the hosts on a specific port 80. To detect this scan using Wireshark, we can filter the traffic based on tcp.flags.syn == 1 and tcp.flags.ack ==0 && tcp.dstport==80 && ip.src==119.42.83.139 && ip.dst==94.65.215.0/24

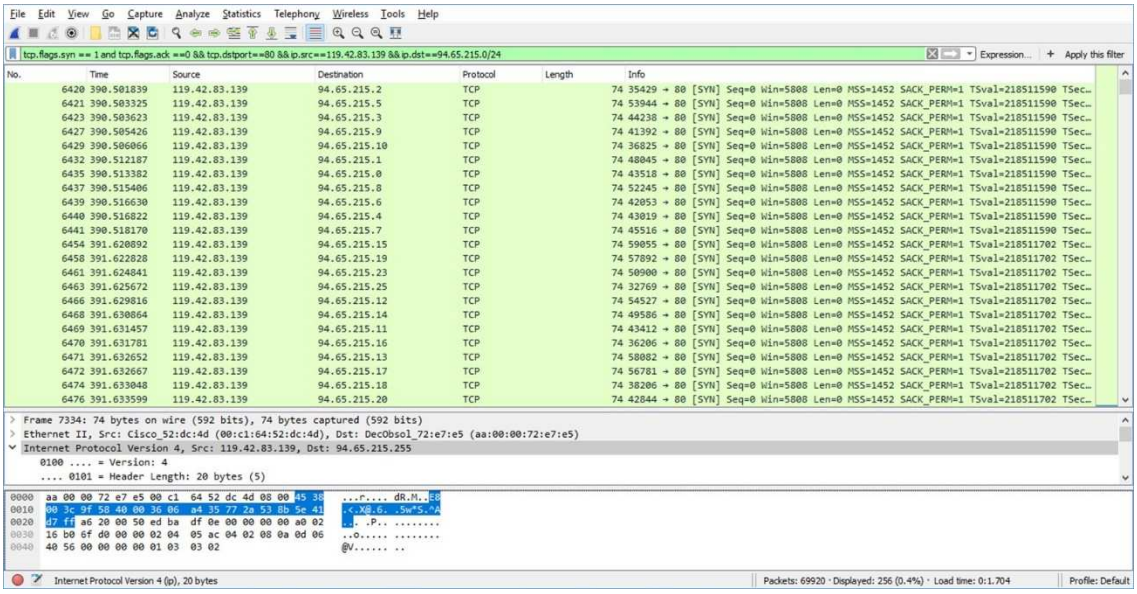


Figure 11a: SYN scan on a specific port 80 (screenshot from Wireshark)

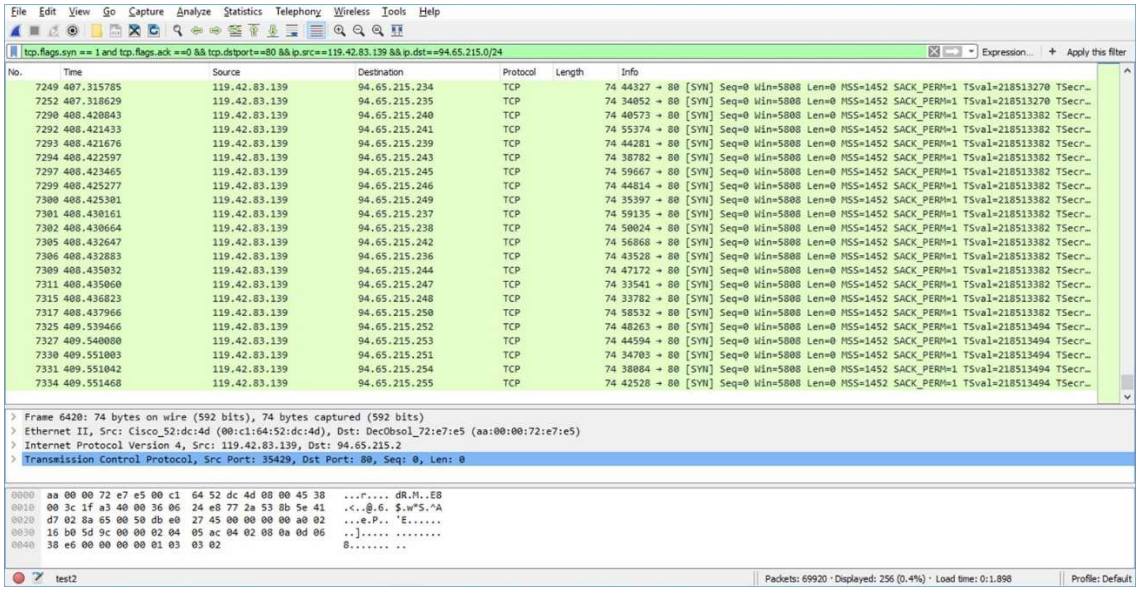


Figure 11b: SYN scan on a specific port 80 (screenshot from Wireshark)

Another indication is the comparison between the TCP packets which are sent to a specific tcp.port==80 and the total packets which are sent at any port tcp.port >=1024 (well-known ports) and out any traffic to tcp.port!=80 in a specific data capture (Statistics | I/O Graph).

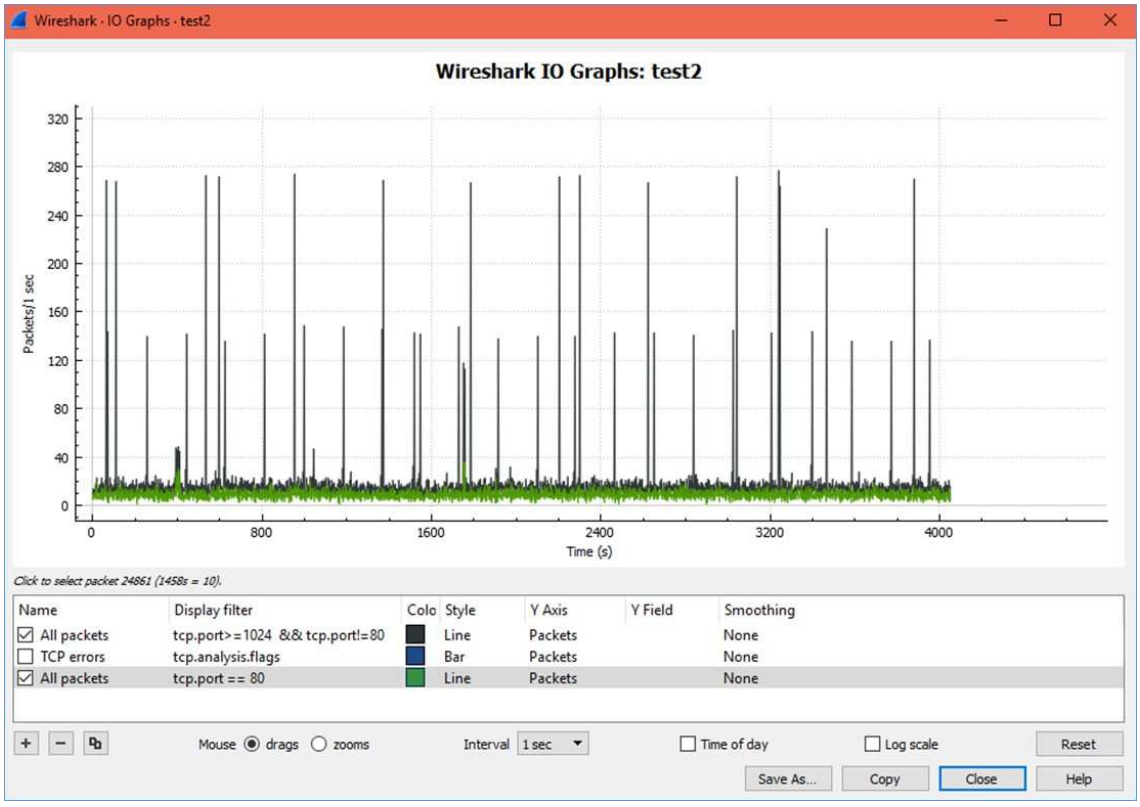


Figure 12: I/O graph of TCP packets (screenshot from Wireshark)

The next stage of an attack is the delivery. This is where the criminals craft a tool to attack their chosen target, using the information they have gathered from the first stage. If the attackers will find an open port then determine which methods to use in order to deliver

malicious payloads. As more and more vulnerabilities are being reported, organizations are forced to spend an increasing amount of resources to stay properly informed about vulnerabilities affecting their network infrastructure and applications.

After the attackers determine an interesting target and they've redirected him to the attack server, they deploy an exploit against a vulnerable application or system, typically using an exploit kit or weaponized document. This allows the attack to gain an initial entry point into the organization. So, if we want to prevent it from happening, we need to considerate the following:

- To run an exploit, an attacker needs a vulnerability.
- To find a vulnerability, the attacker needs to fingerprint all services which run on the machine (find out which protocol they use, which programs implement them and preferably the versions of those programs).
- To fingerprint a service, the attacker needs to know that there is one running on a publicly accessible port.
- To find out which publicly accessible ports run services, the attacker needs to run a port scan.

Understanding the attack chain means that we know that while there are multiple ways in which our networks could be compromised, there are also multiple ways to disrupt the actual attack. It's very important to understand what methods attackers will use to find the holes in networks and exploit their vulnerabilities. By identifying steps in the attack chain, we can deploy appropriate defenses at each stage to prevent breaches from happening in the first place.

4.2 Data analysis and results

This section presents the attack patterns found on the honeynet and by displaying some facts and analyzing different protocols Telnet, SSH, WEB, FTP in which Telnet is mainly used in IoT devices, so this honeynet is also referred as IoT honeynet mechanism.

The honeynet has been deployed for around the time of four months and captures the details of number of unique IPs, intersection of IPs among these protocols, the most attacked protocol, the least attack protocol etc. All these results will be presented in graphical forms (were generated from Arbor Networks, Inc. platform[14]), graphical statistics charts and tables in this section.

Before starting to analyze possible attacks, the best way is first to study how or where the possible attackers begin to launch attacks. Most of the experiment results show that they normally start with information gathering about their targets through port scanning to find open ports and vulnerabilities of their victims.

An important fact is that we could observe malicious connections immediately after launching the honeynet in the network. Sometimes attackers could send packet traffic from forged source IP addresses which make the analysis harder to identify their original source addresses. Most common activities observed were TCP, UDP and ICMP port scans by attackers in order to check the vulnerabilities on the operating systems.

Port Scanning is often used legally by system administrators to verify the security level of their networks and by attackers to find the vulnerabilities. Port scanning doesn't mean scanning only TCP ports. Nevertheless, UDP port scanning is used more often as well. The basic attacks we observed from the log files were TCP, UDP and ICMP port scans. UDP scanning is slower than TCP scanning, therefore many system administrators usually do not secure these ports. Such scans are more useful for the attackers, thus they send empty UDP datagrams to the target port. If the port is closed, then the attacker will receive "ICMP Port Unreachable" message. If it is opened or filtered, then an error message will be sent back or incoming datagram simply will be ignored. Thus, the attackers can determine which ports are open or closed.

The capturing, processing, preservation and analysis of information obtained from the following tcpdump filter:

[14] https://www.arbornetworks.com/

```
matriantop@troy:~$ sudo tcpdump -s0 net 195.167.82.128/28 or net 212.205.155.176
/29 or net 94.65.215.0/24 or  net 195.170.21.128/25 or net 212.205.84.128/25 or
net 94.68.152.0/24 or net 94.68.153.0/24 -w test10.pcap
[sudo] password for matriantop:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 byt
es
```

Figure 13: Tcpdump filter for capturing data

As it is mentioned previously Wireshark's strong point is in effect its capability of understanding the structure of packets belonging to various network protocols and therefore displaying the content of different fields and the encapsulation details of very different networking protocols. This feature along with the integrated support for sorting and filtering network packets depending on various qualities and attributes, makes Wireshark a powerful and flexible software for network traffic analysis.

Given the nature and the time limitations of our experiment, we considered the most interesting and representative criteria to be the following:

- Connections per port and per service: Knowing the exact number of connections on each port, we can obtain useful statistics for the most commonly attacked ports. The same applies also for the most attacked services.
- Information gathered about attackers: We must know as much as possible about the attacker: the IP address, the country of origin and etc. In fact, this has been the main idea behind honeynet; to lure the attackers and gain information about them.

During the operation of honeynet, different kind of possible threats were suffered. Figure 14 shows the number of connection attempts per protocol type during the observation period.

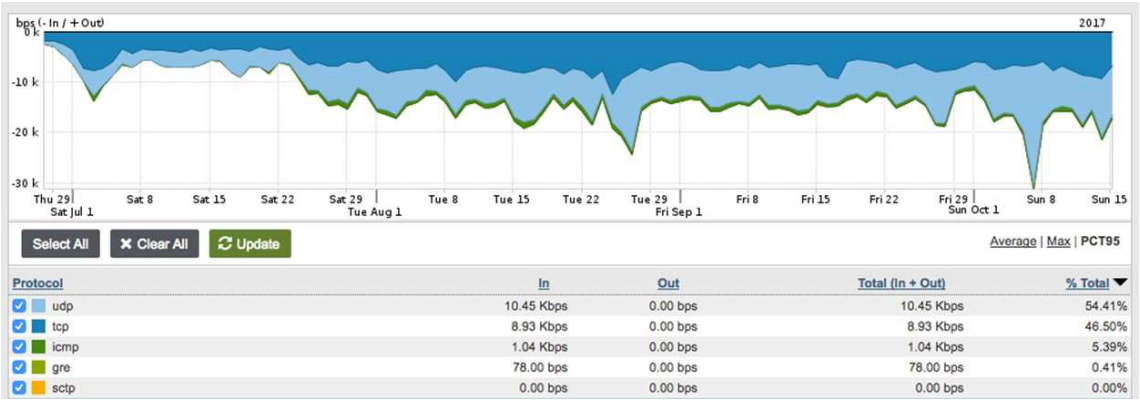
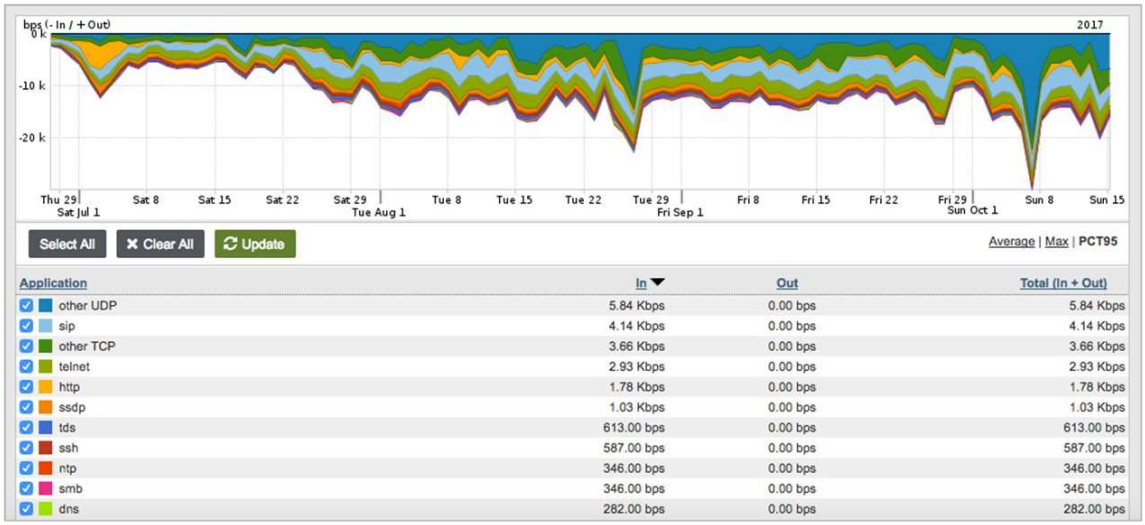


Figure 14: Total connection attempts per protocol type (Source: Arbor Networks, Inc.)

As it was expected, the highest percent of connections are UDP protocol related. This is because UDP, by design, is a connectionless protocol that does not validate source IP addresses and it is very easy to forge the IP packet datagram to include an arbitrary source IP address. When many UDP packets have their source IP address forged to the victim IP address, the destination server responds to the victim (instead of the attacker) creating a reflected DoS attack. The lowest percentage from the well-known protocols belongs to ICMP connections. Technically speaking, it wouldn't be precise referring to them as real connections since ICMP packets are not supposed to establish connections but basically control them through status and error messages.

More specifically, we would be extremely interested in acknowledging which are these ports that accept the most connection attempts. Knowing which are these ports is crucial as it can help network administrators in applying rules on firewalls and other network security related systems, restricting access to the services residing on these ports. Hackers are quite aware of the vulnerabilities of specific ports and these become their first targets in their attempt to break into a system.

The following Figures show the total of applications that received in comparison with the inbound traffic (per Kbps), the number of connections by destination port and protocol, practically demonstrating which the most targeted ports were during the period of the deployment. It is important to mention that the following statistics do not mean that there has been successful access because of the Dark IP networks but they are mainly based on unauthorized attempts and scans.



Application	In	Out	Total (In + Out)
other UDP	5.84 Kbps	0.00 bps	5.84 Kbps
sip	4.14 Kbps	0.00 bps	4.14 Kbps
other TCP	3.66 Kbps	0.00 bps	3.66 Kbps
telnet	2.93 Kbps	0.00 bps	2.93 Kbps
http	1.78 Kbps	0.00 bps	1.78 Kbps
ssdp	1.03 Kbps	0.00 bps	1.03 Kbps
tds	613.00 bps	0.00 bps	613.00 bps
ssh	587.00 bps	0.00 bps	587.00 bps
ntp	346.00 bps	0.00 bps	346.00 bps
smb	346.00 bps	0.00 bps	346.00 bps
dns	282.00 bps	0.00 bps	282.00 bps

Figure 15: Top attacked services ports of honeynet (Source: Arbor Networks, Inc.)

Figure 15 shows the top scanned TCP and UDP ports and consequently the service vulnerability on each port. This indicates that the honeynet has been continuously scanned by attackers and the rate of the connection attempts on the network services SIP (5060), TELNET (23, 2323) and HTTP (80) was higher.

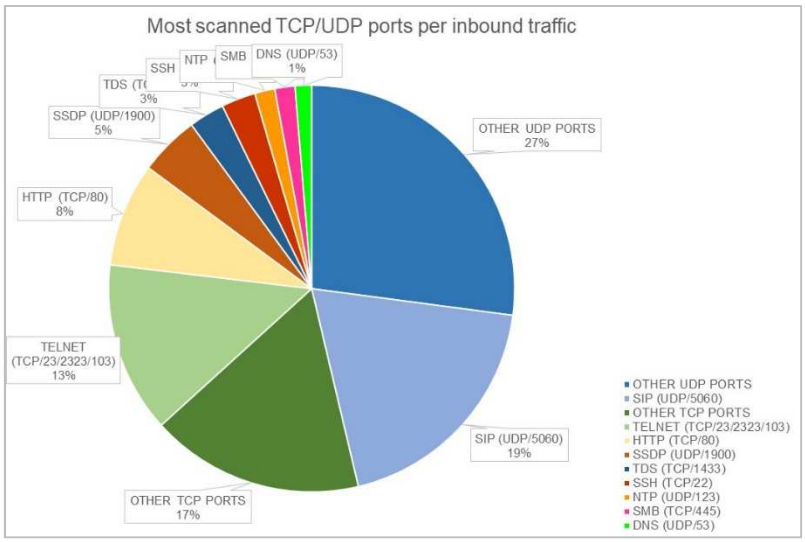


Figure 16: The graph of most scanned TCP/UDP ports

The Figure below displays the most commonly used UDP services. The UDP ports included in the most accessed ports are 5060, 123 and 53. The UDP port 5060 is by default associated with the Session Initiation Protocol (SIP) supporting sessions involving streaming video, voice calls, instant messaging application, Voice over IP (VoIP) and even online gaming.

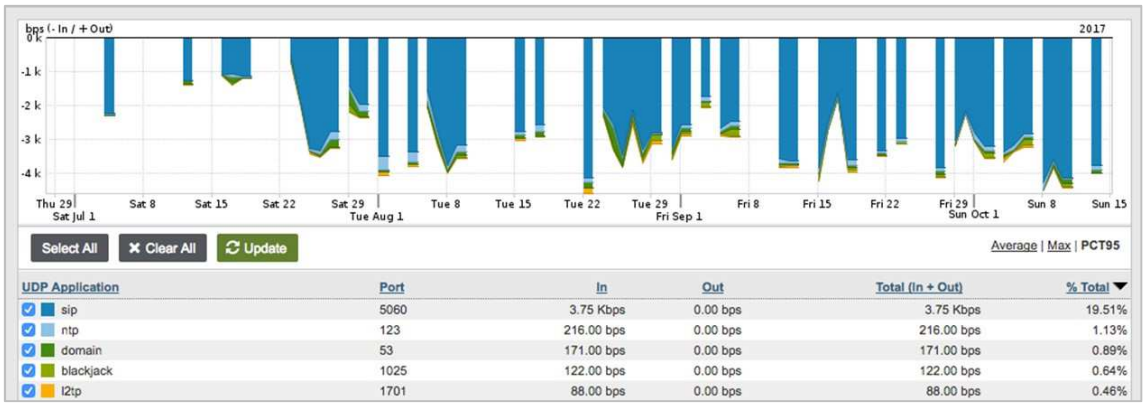


Figure 17: Top UDP attacked ports of honeynet (Source: Arbor Networks, Inc.)

The Network Time Protocol (NTP), typically UDP port 123, is one of the oldest networking protocols and is used for clock synchronization between computer systems. NTP DDoS is an amplification attack that relies on the use of publicly accessible NTP servers to overwhelm a target system with UDP traffic. It can be used easily in "reflection attacks" to initiate DDoS attacks. A remote attacker can send a carefully crafted packet that can overflow a stack buffer and potentially allow malicious code to be executed with the privilege level of the ntpd process.

In the third position of the list above, is the UDP port 53 DNS (Domain Name System) which can cause a DoS attack with DNS queries. Another method is to perform a ping of death or a TCP SYN flood attack. The idea behind this is to overwhelm server resources (CPU and memory) to stop it responding to queries. Though DNS servers are protected by firewalls, if care is not taken to block DNS UDP ports from non-trusted networks, it exposes the name resolution system to this attack.

Among other information, an investigation that the transport layer will return is the port each packet is directed to. Associating well-known ports to their correspondent protocols, it is possible to determine the application layer protocols that the attackers most tried to exploit. The most commonly scanned ports are the TCP ones where vulnerable popular services accessed via the Internet reside such as HTTP, SSH, TELNET. As we can see from the following screenshot, most of the TCP connection attempts were addressed to destination TCP ports 23, 80, 1433, 22, 445 and 8080.

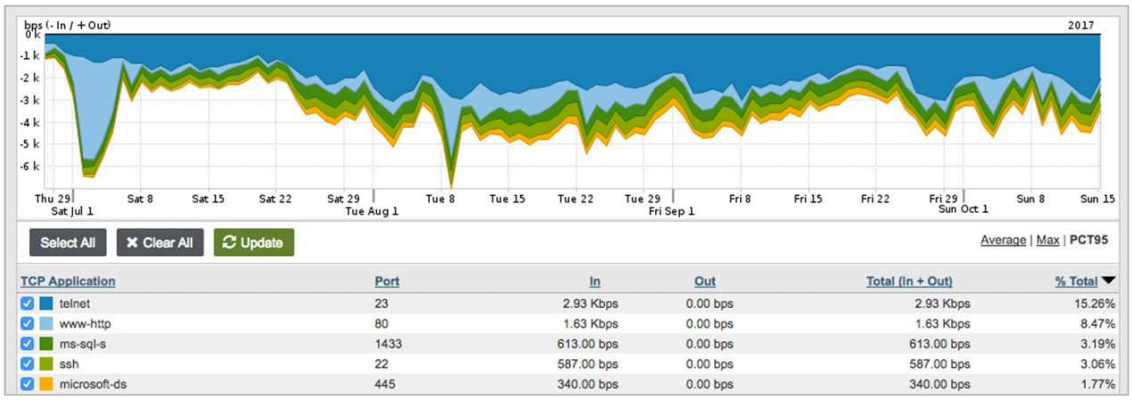


Figure 18: Top TCP attacked ports of honeynet (Source: Arbor Networks, Inc.)

From the results, we can see that the most probed port is TCP port 23 which is assigned to the Telnet daemon. Telnet is a first class means for intruders to gain unauthorized access to a computer system. From the log files of the shell script emulating a Telnet service on this port, the most common combinations of username and password are used when probed by the emulated service, included the words root and admin.

TCP port 80 is the port used by default by the Hyper Text Transfer Protocol (HTTP) on web servers. On this port, the servers are listening for any web traffic from clients. TCP Port 80 constitutes one of the most frequent target ports for hackers and there is a variety of attacks and worms to which Web servers are vulnerable. Many hackers send packets to this port for scanning, DoS Attack, setting the Backdoor or other purposes.

TCP port 22 which is listed lower constitutes the secure alternative to Telnet and other insecure remote connection protocols. Secure Shell (SSH) is an encrypted network protocol offering confidentiality and integrity of data and creating a secure channel over insecure networks like the Internet. The greatest security risk regarding SSH occurred when an adversary manages to log in as the system's administrator. This can be achieved via brute force attack especially when the password is easy to guess.

The service running on this port is Microsoft-DS which is used for resource sharing on Windows and other samba based connections. In general, TCP port 445 is one of the most commonly attacked ports, being an easy target for hackers who are searching for unsecure resource shares on Windows systems.

Despite the limited time of the honeynet deployment, we can see that our statistics regarding the top scanned ports are quite similar to the ones expected according to recently researches.

After examining the statistics deriving from the type of connections by protocol that were attempted to our server, it would be rather interesting to examine in total the number of connection attempts per unique IP address, countries or regions of origin and the conclusions we can draw from these data.

Analysis of the header of the third layer of the packets shows that France and China are the most representative sources of the attacks to the honeynet while the IP addresses that most attacked the network are from Russian Federation. Figures 19 and 20 show the relative contribution of the most frequent countries and IP address for the total traffic, along with the ports to where each IP sent its packets.

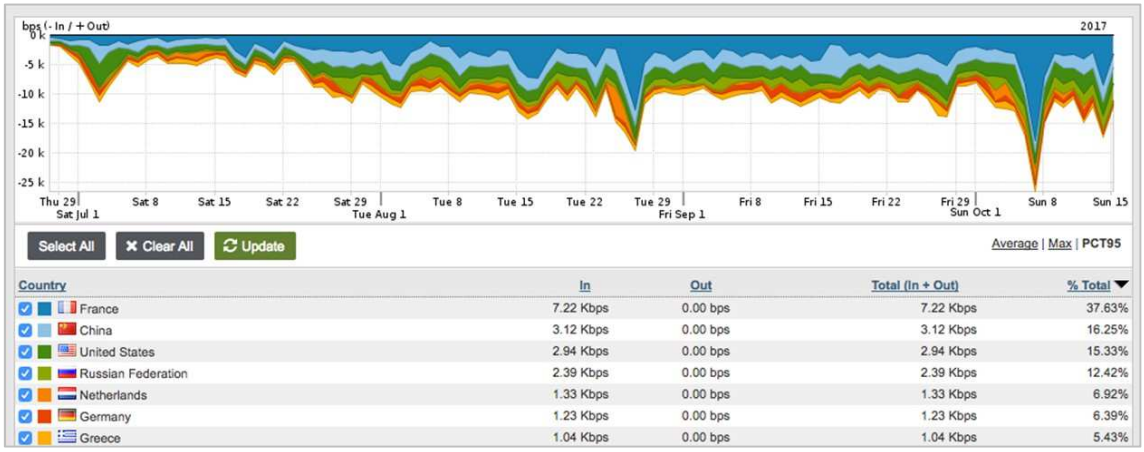
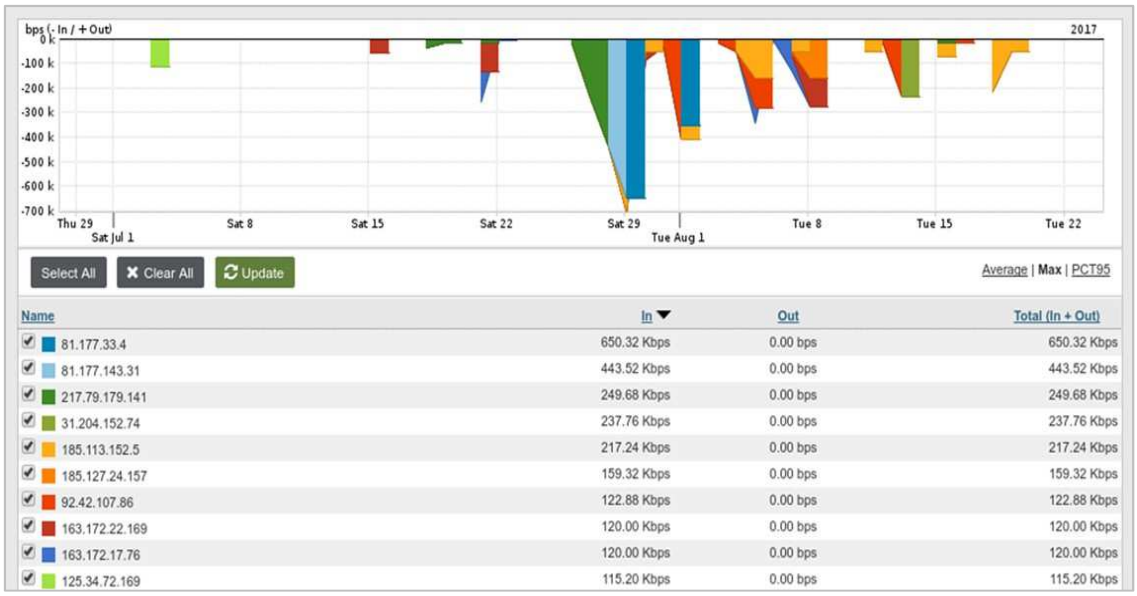


Figure 19: Countries where the most attacks originated (Source: Arbor Networks, Inc.)

We observed several unique IP addresses originating from different countries across the globe. Among these countries the highest numbers of attacks were received from France, China and United States of America while the least attempts were generated from Greece (5,43%).

During the analysis of layer 3, it became evident the similarity of the IP addresses from the countries France and Russian Federation. This is a strong indication that the hosts using those IP addresses probably belong to the same subnetwork and therefore it either corresponds to a network with several infected devices (for example as part of a botnet) or to a network setup for malicious purposes (Table 4).



Name	In	Out	Total (In + Out)
81.177.33.4	650.32 Kbps	0.00 bps	650.32 Kbps
81.177.143.31	443.52 Kbps	0.00 bps	443.52 Kbps
217.79.179.141	249.68 Kbps	0.00 bps	249.68 Kbps
31.204.152.74	237.76 Kbps	0.00 bps	237.76 Kbps
185.113.152.5	217.24 Kbps	0.00 bps	217.24 Kbps
185.127.24.157	159.32 Kbps	0.00 bps	159.32 Kbps
92.42.107.86	122.88 Kbps	0.00 bps	122.88 Kbps
163.172.22.169	120.00 Kbps	0.00 bps	120.00 Kbps
163.172.17.76	120.00 Kbps	0.00 bps	120.00 Kbps
125.34.72.169	115.20 Kbps	0.00 bps	115.20 Kbps

Figure 20: Top 10 external talkers of honeynet (Source: Arbor Networks, Inc.)

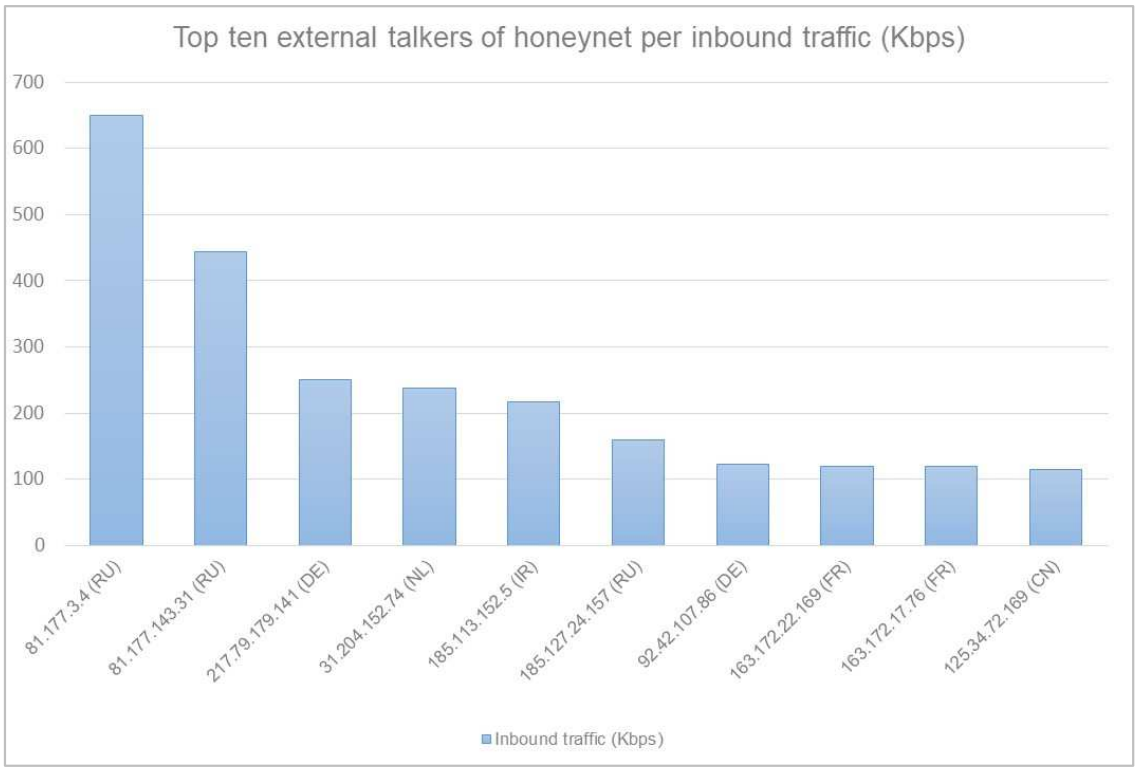


Figure 21: Vertical bar chart depicting the inbound traffic per unique IP address for the top 10 countries along with their codes

The following table displays the top ten IP addresses connected to the honeynet ordered by volume of connections and accompanied by geo-location and other useful information.

ID	IP address	Inbound traffic (Kbps)	City	Country name	Code	Longitude	Latitude
1	81.177.33.4	650.32	Sochi	Russia	RU	39.7257	43.5992
2	81.177.143.31	443.52	Sochi	Russia	RU	39.7257	43.5992
3	217.79.179.141	249.68	Dusseldorf	Germany	DE	6.7762	51.2217
4	31.204.152.74	237.76	Rotterdam	Netherlands	NL	4.4792	51.9225
5	185.113.152.5	217.24	Ahvaz	Iran	IR	48.6693	31.3203
6	185.127.24.157	159.32	Moscow	Russia	RU	37.6156	55.7522
7	92.42.107.86	122.88	Koeln	Germany	DE	6.95	50.9333
8	163.172.22.169	120.00	Paris	France	FR	2.3488	48.8534
9	163.172.17.76	120.00	Paris	France	FR	2.3488	48.8534
10	125.34.72.169	115.20	Beijing	China	CN	116.3972	39.9075

Table 4: Top 10 IP addresses connected to the honeynet system sorted by the inbound traffic

As we can see from above, half of the connection attempts targeting the honeynet were launched from France, United States, China and Russian Federation. However, the IP address with the highest number of connection attempts to our system originated from Russia with approximately 650.32Kbps inbound traffic. Following Russia, in the third position comes the United States and in forth position comes Netherlands with an IP address 31.204.152.74 which is blacklisted for malicious activity in two lists[15]. In the last position comes an IP address 125.34.72.169 which is also blacklisted for malicious activity in two lists[16].

Finally, the last statistics given are those concerning the number of connections per destination IP as well as the most commonly seen packet lengths. Regarding information provided in Figure 22, we can see a list of IP addresses that the attackers most tried to exploit or scanned during the experimental implementation.

[15] The information was taken from online source: https://whoisip.ovh/blacklist/31.204.152.74

[16] The information was taken from online source: https://whoisip.ovh/blacklist/125.34.72.169

Figure 22: Top 10 internal talkers of honeynet (Source: Arbor Networks, Inc.)

Figure 23 shows the most frequent packet lengths. The highest value of packet length is 46 bytes, which is purely IP and TCP packet lengths. The packet lengths of size 52 bytes refer to the TCP pure ACKs due to the length of IP and TCP along with 12 bytes of options length in the TCP/IP stack. Another most common size is 1500 bytes which refers to the maximum size of IP packet in Ethernet.

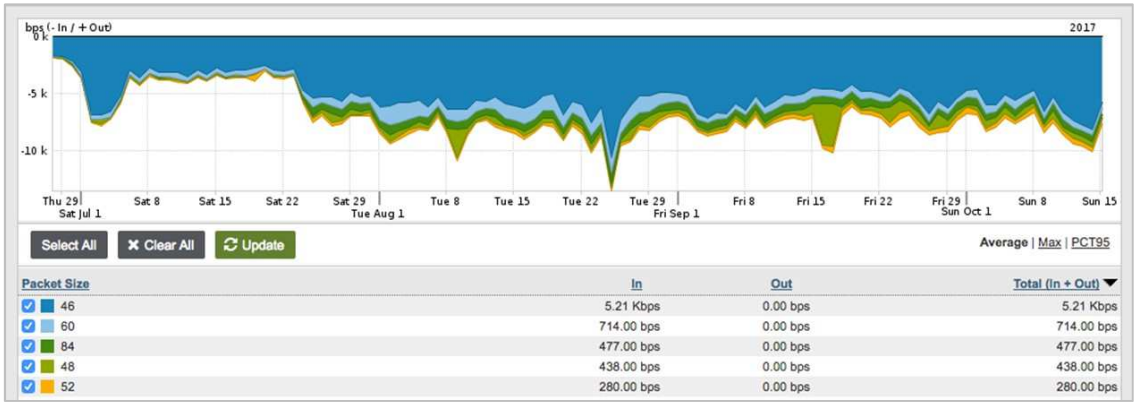


Figure 23: Top frequent packet lengths (Source: Arbor Networks, Inc.)

4.2 Fingerprints of network attacks

It's common knowledge that network abuse is on the increase. Many of these attacks come from inside ISPs network, as a result of compromised hosts within their network. In order to increase ISP security and deal with this type of network abuse in a more efficient manner, ISPs need layers of security in place that include robust threat intelligence sharing, more advanced detection and response technology, and true integration and automation capabilities.

Traditional intrusion detection relies on the inspection of individual packets, which are scanned for suspicious patterns or activities. However, the massive increase of link speeds and throughputs, especially in large networks, makes this approach ineffective. Today's attacks generally fall into two categories; DDoS attacks and hacks that steal data, such as SQL injection or other command injection attacks. DDoS attacks that target the network layer use a variety of techniques and ISPs need to defend against all of them.

Every day, network operators and security engineers battle emerging attacks. To effectively protect their networks, they must understand the true nature of the threats and implement the processes which best counter these threats. A very useful idea for detecting attacks is to create traffic behavioral “fingerprints” for known and unknown emerging threats, before signatures can be created for IPs.

The creation of “fingerprints” begins with cutting-edge techniques, processes and methods to better identify the most damaging composite threats. This model is used to spot network attacks and automatically generate a profile, or "fingerprint" of the attack in a standard data file format called pcap. That fingerprint information is passed along to other service providers closer to the source of the attack, which can then block the source of the traffic.

According to the Figure 24, given a list of the most important (possible) attacks that took place in the experimental implementation of honeynet. The majority of these attacks were blind, thus they randomly scanned honeynet IPs and tried to compromise the open ports which are more vulnerable and exploitable. Statistics of the targeted and blind attacks show that Mirai botnet was the most serious attack with the highest percentage of 29,4% while in the second position came the SSDP reflective/amplified attack with a percentage of 16,6%.

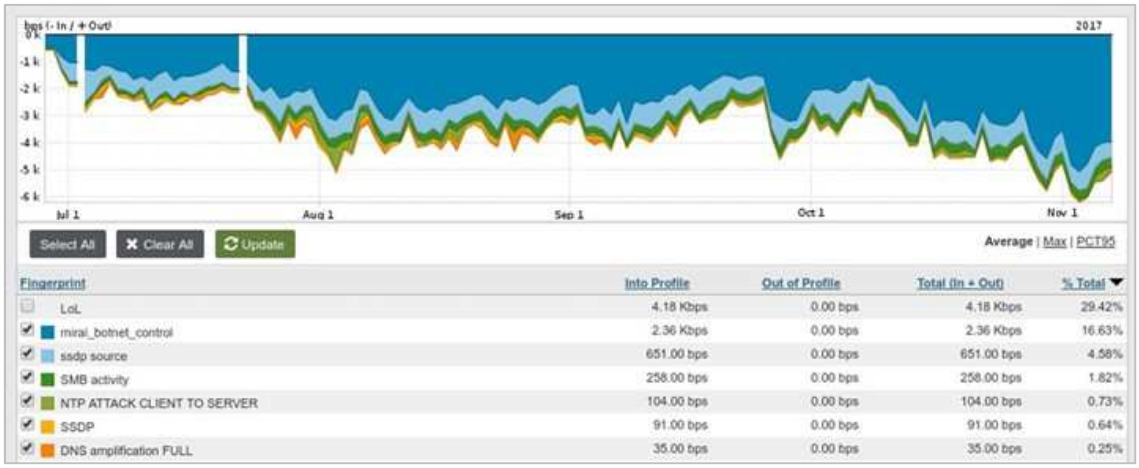


Figure 24: List of the top network attacks of honeynet (Source: Arbor Networks, Inc.)

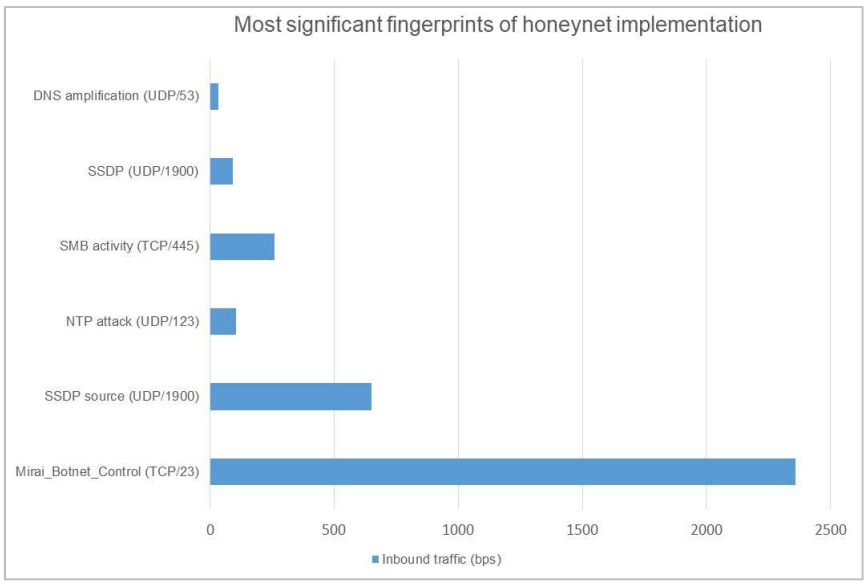


Figure 25: Inbound traffic of attacks by type

Different application layer protocols have different fields and information and can be used to exploit different vulnerabilities. Thus, each protocol requires a specialized investigation on its fields. In this section, inspection and analysis of payload is performed on some of the protocols the attackers tried to exploit.

4.2.1 The Mirai analysis: case study

Recently, the code of Mirai botnet was made available to the public. Mirai can be considered a kind of malware that infects IoT devices running BusyBox. The potential collateral impact of DDoS attacks launched by the Mirai botnet can be highly significant, depending upon the target selection and efficacy of a given attack. Outbound DDoS attacks launched by Mirai bots can cause significant network performance issues or outages for broadband access network operators. To understand this, let us consider a very simple example of creating a signature.

Fingerprint	Signature
Mirai_botnet_control	proto tcp and (src port 1024..65535 and (dst port 23 or dst port 103 or dst port 2323))

Figure 26: Signature for the fingerprint Mirai_botnet_control

Mirai botnet is the threat of honeynet with the growth inbound traffic. This is a result after a lot of scans on TCP ports 23, 2323 or 103. Although most Mirai infections occur through TCP ports 23 and 2323, Mirai also relies on other TCP ports to commandeer devices, for example port 7547, which ISPs use to remotely manage customers' broadband routers. In this case study we will focus on the dest_port 23 for Telnet to create our correlation search.

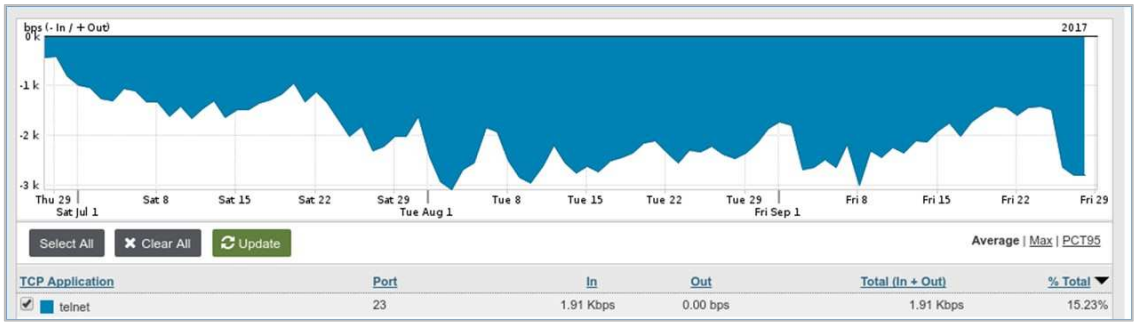


Figure 27: Significant increase of Telnet traffic on port 23 (Source: Arbor Networks, Inc.)

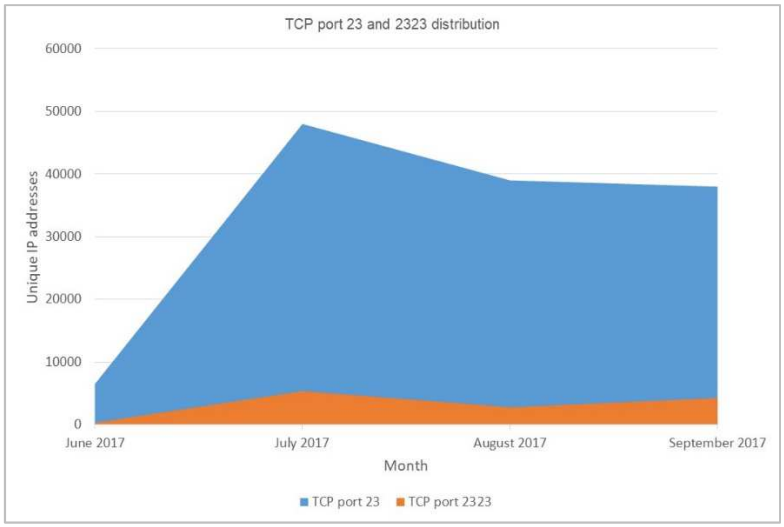


Figure 28: Significant increase of Telnet traffic on port 23 and 2323

One of the main characteristics of Mirai is that it works by brute forcing, over Telnet, weak and default credentials on devices. Once it gains controls of the device, it reports the infection to a command and control server and the device is now part of a botnet. Once many owners of devices directly and indirectly connected to the Internet never change the default username and password, so that Mirai could infect million devices. Figure 28, shows the distribution of port 23 and port 2323 over the period of four months. It can be observed that the scanning activities of Mirai have a significant peak on July.

The following example, shows the attempts of the source IP address 80.82.70.26 to scan all the hosts from network 94.68.152.0/24 by sending SYN packets on TCP port 23 in order to check if the port is opened and get access if the server was corresponded.

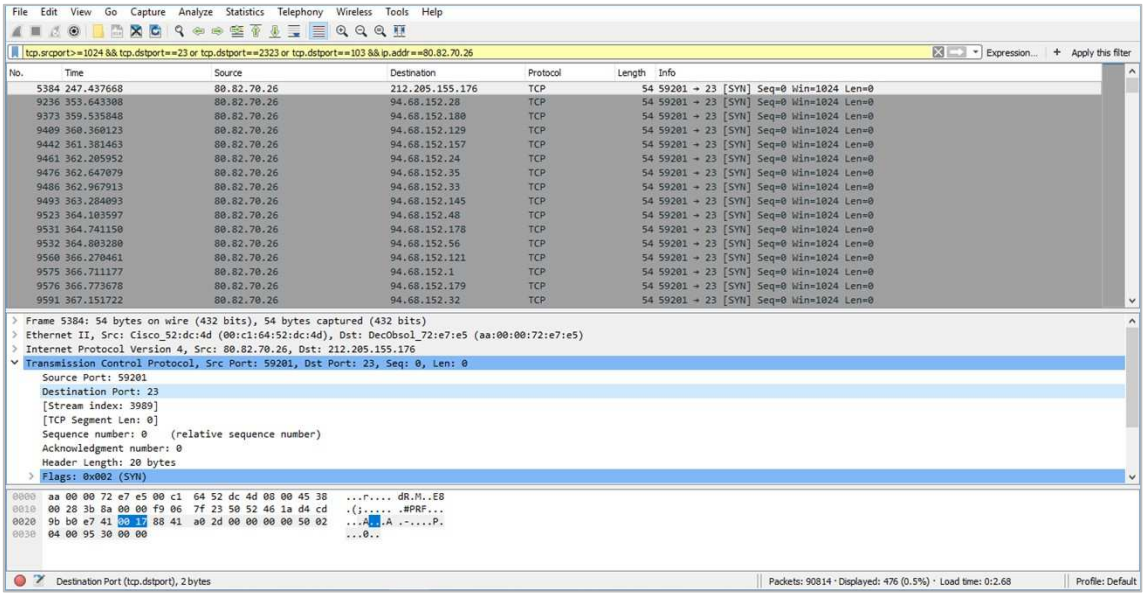


Figure 29: Example of scanning TCP ports 23, 2323,103 (screenshot from Wireshark)

In order to inhibit scanning for vulnerable IoT devices, it is possible for broadband access network operators to implement ACLs at an appropriate point in the network topology to prohibit high-port TCP traffic destined for TCP/23, 2323,103 on their customer access networks. Such policies would typically be implemented as ingress ACLs on the core interfaces of broadband customer aggregation gateways.

4.2.2 SSDP source analysis: case study

A Simple Service Discovery Protocol (SSDP) attack is a reflection-based DDoS attack that exploits Universal Plug and Play (UPnP) networking protocols in order to send an amplified amount of traffic to a targeted victim, overwhelming the target's infrastructure and taking their web resource offline. SSDP allows UPnP devices to send and receive information using UDP port 1900 (for M-SEARCH requests and for Notify packets) and is attractive to attackers because of its open state that allows spoofing and amplification.

The SSDP DDoS attack falls into the same category as the DNS and NTP amplified DDoS attacks where attackers use a smaller botnet that spoofs their victim's IP addresses. Attackers use that botnet to query any access point which has the UPnP service open to the internet. The SSDP source attack was composed of UDP packets with source port 1900 and can be divided into the following two main parts [40]:

1. Scan phase: The attacker sends an M-SEARCH request as discover packet to a range of IPs. The UPnP enabled device responds to the request with the HTTP location of its device description file. With these response messages the attacker gathers an IP list of vulnerable devices.

2. Attack phase: The attacker sends a spoofed UDP M-SEARCH packets (containing the IP address of the victim) to the various devices found. The spoofed M-SEARCH packets with ssdp:all (search for all UPnP devices and services) or ssdp:discover is sent and each device replies with an amplified answer UPnP:rootdevice that contains all the services it provides.

But how does a SSDP attack work? Under normal circumstances, the SSDP protocol is used to allow UPnP devices to broadcast their existence to other devices on the network. For example, after connecting a printer that supports UPnP, the printer gets an IP address from the DHCP server and using SSDP notifies that it is available by sending a multicast UDP packet from port 1900. Multicast address then tells all the hosts on the network about the new printer. Once a host hears the discovery message about the printer, it makes a request to the printer for a complete description of its services. The printer then responds directly to that host with a complete list of everything it has to offer.

Technically devices respond if the Search Target (ST) header field of the M-SEARCH request is “UPnP:rootdevice (search for root devices)” followed by a “uuid” that exactly matches the one advertised by the device or if the M-SEARCH request matches a device type or service type supported by the device. Here are the six steps of a typical SSDP DDoS attack [41]:

1. The attacker conducts a scan looking for UPnP devices that can be utilized as amplification factors.
2. As the attacker discovers networked devices, they create a list of all the devices that respond.
3. The attacker creates a UDP packet with the spoofed address of the targeted victim.
4. The attacker then uses a botnet to send a spoofed discovery packet to each UPnP device with a request for as much data as possible by setting certain flags, specifically ssdp:rootdevice or ssdp:all.
5. As a result, each device will send a reply to the targeted victim with an amount of data up to about 30 times larger than the attacker’s request.
6. The target then receives a large volume of traffic from all the devices and becomes overwhelmed, potentially resulting in denial-of-service to legitimate traffic.

The Figure below shows an actual scan package sent from an attacker. The attacker sends an M-SEARCH request as discover packet to a range of IPs and waits for the UPnP devices to be responded to the request with the HTTP location of their device description file.

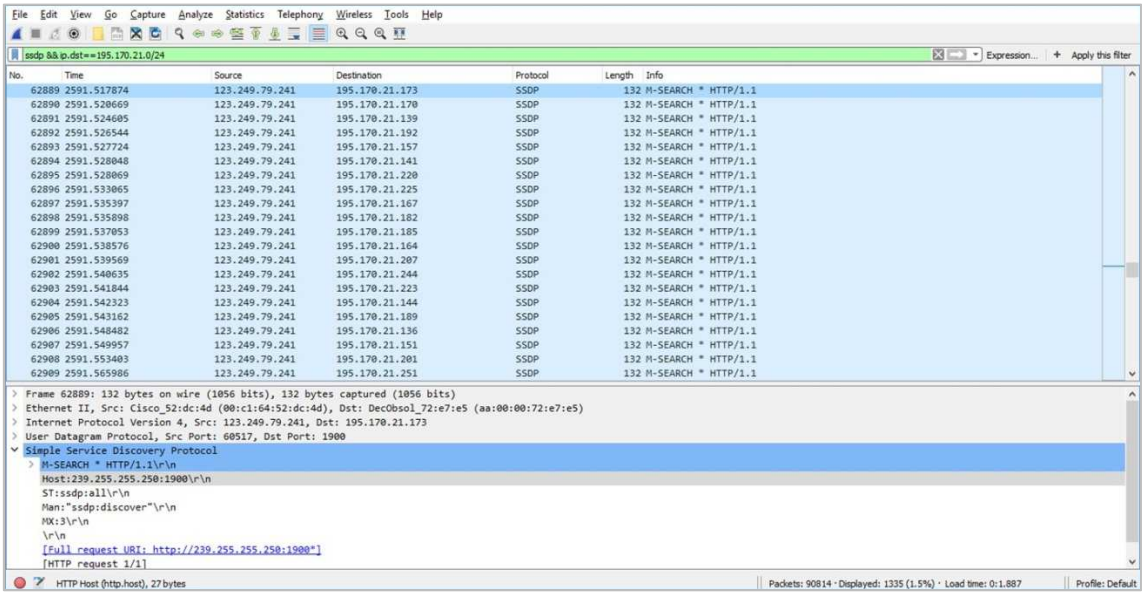


Figure 30: Example of SSDP M-SEARCH request for discover UPnP device

With this procedure the attacker gathers an IP list of vulnerable devices. The screenshot above shows an example of SSDP attack (M-SEARCH request) with no response. This is explained because the destination IP addresses are Dark IPs and they don't send back to source IPs any packet or response.

The following example illustrates that the attacker doesn't send an M-SEARCH request in order to discover UPnP devices but has spoofed the IP address of our network 195.170.21.138 and sends M-SEARCH responses.

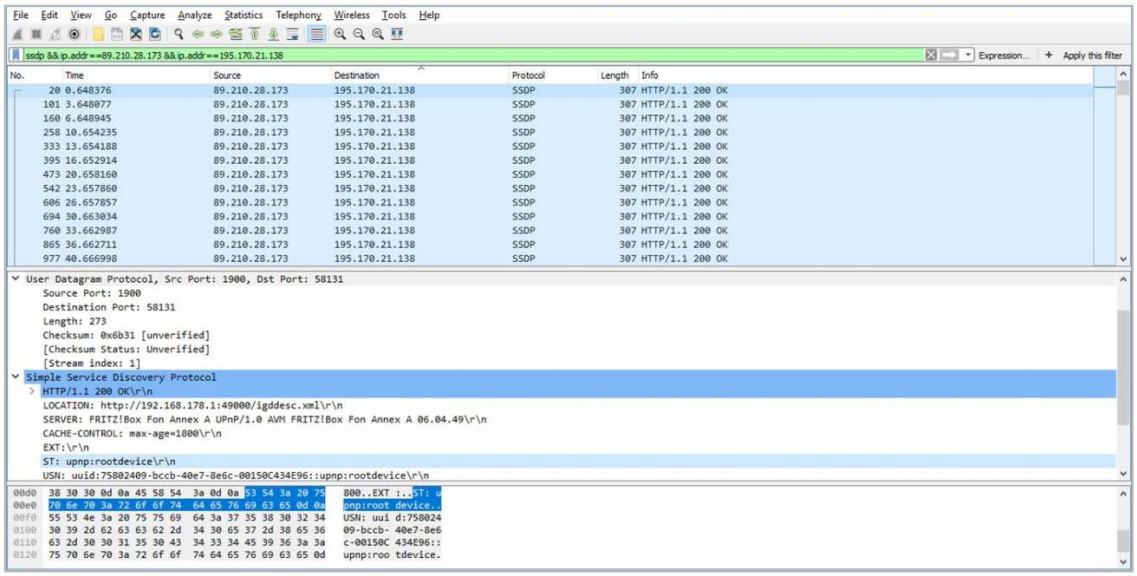


Figure 31: Example of SSDP response to M-SEARCH (without M-SEARCH request)

Figure 32: M-search reply for IP spoofed IP 195.170.21.138 (Screenshot from Wireshark)

As we can see from Figures 30 and 31 the original request is only 132 bytes but the targeted (spoofed) response message is 307 bytes. This is an amplification DDoS attack. A very interesting statistic graph comes up to show us the attempts for SSDP attack. In this case the victim 195.170.21.138 has been spoofed from attackers who claimed that they belong to our network 195.170.21.0/24 and forced it to response HTTP/1.1 200 OK without M-SEARCH request.

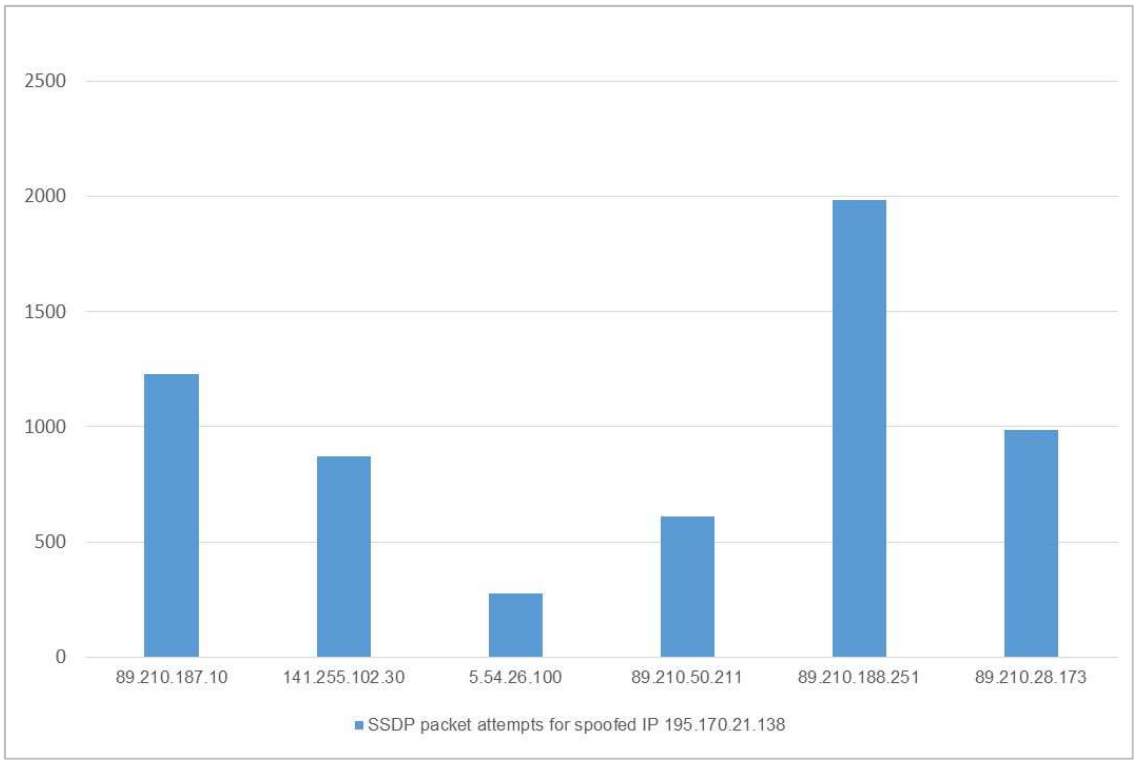


Figure 33: SSDP packet attempts for spoofed IP 195.170.21.138

Similar to other amplified reflexive attacks, detection and mitigation is easier than other direct attacks like SYN flood or GET flood. For network administrators and security experts, a key mitigation point is to block any incoming UDP traffic on port 1900 at the firewall. Provided the volume of traffic isn't enough to overwhelm the network infrastructure, filtering traffic from this port will likely be able to mitigate such an attack.

4.2.3 ICMP attacks analysis: case study

A ping (ICMP) flood is a DoS attack in which the attacker attempts to overwhelm a targeted device with ICMP echo-request packets, causing the target to become inaccessible to normal traffic. When the attack traffic comes from multiple devices, the attack becomes a DDoS attack. ICMP stands for Internet Control Message Protocol and is the most used protocol in networking technology. As a connectionless protocol, ICMP doesn't use any port number and works in the network layer. ICMP is commonly used for diagnostic purposes, error reporting or querying any server and right now attackers are using ICMP to send payloads. The DDoS form of a ping (ICMP) flood can be broken down into 2 repeating steps:

1. The attacker sends many ICMP echo request packets to the targeted server using multiple devices.
2. The targeted server then sends an ICMP echo reply packet to each requesting device's IP address as a response.

A classic way to discover hosts on the network is to send an ICMP echo request (type 8) which should prompt target hosts to respond with ICMP echo reply messages (type 0). ICMP Error Messages (Protocol/Port Unreachable) can be used to find out the open ports to an IP address or a LAN segment. Different types of scanners are available in the market that use ICMP to check whether a port is open or not. Usually ICMP packets are sent without any payload to each specified protocol on the target machine. If an ICMP Protocol Unreachable error message (code 2) is received, it means the protocol is not used.

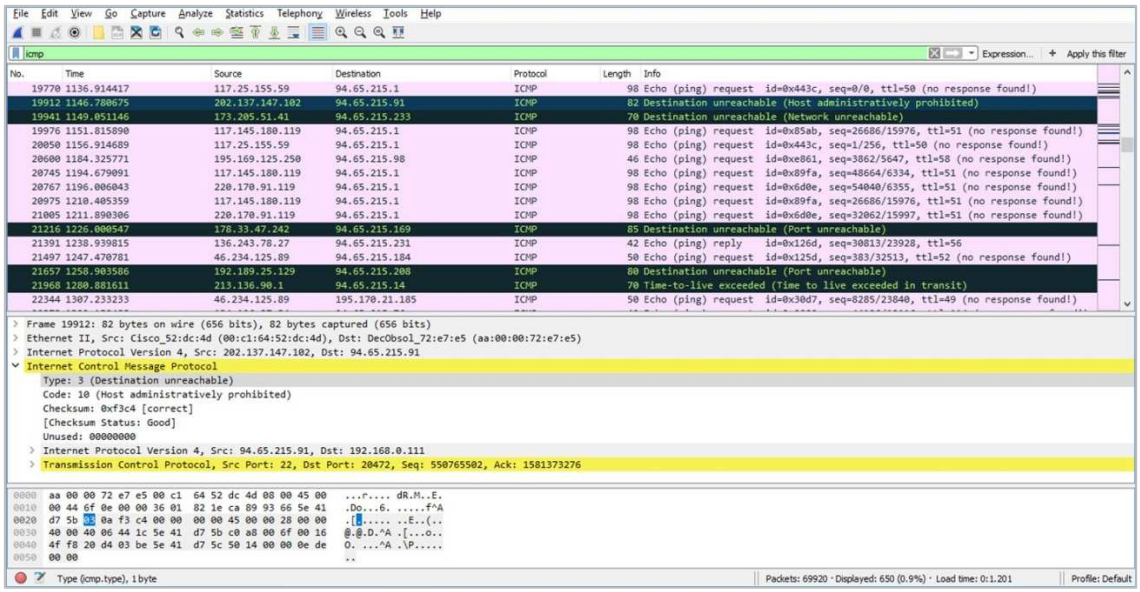


Figure 34: Example of Destination Unreachable messages (screenshot from Wireshark)

When an ICMP Destination Unreachable message is sent by a router, it means that the router is unable to send the packet to its final destination. The router then discards the original packet. There are different reasons why a destination might be not be reached. Some of these are, the source host could specify a non-exist address or the router hasn't the route to reach the destination. Destination Unreachable messages include Network unreachable (code 0), where a failure has occurred in the routing or forwarding of a packet. Another kind is Host unreachable message (code 1), which indicates a delivery failure, such as a wrong subnet mask and Port unreachable messages (code 3) are those when the port is not available. The ICMP Time-to-live exceeded message is sent by the router if the TTL field of the IP packet (expressed in hops or seconds) reaches zero. The table below shows the different types of ICMP Messages.

Type	Code	Description
0/8	0	Echo Reply/Echo Request
3	0-15	Destination Unreachable
4	0	Source Quench
5	0-3	Redirect
9/10	0	Router Advertisement
11	0-1	Time Exceeded
12	0	Parameter Problem
13/14	0	Timestamp Request/Timestamp Reply
17/18	0	Address Mask Request/Address Mask Reply

Table 5: ICMP Message Types [https://ciscohite.wordpress.com/tag/icmp-message-types/]

In any typical attack scenario, the attacker will first start with scanning activities in order to understand the environment of the target, gather information about the target so as to plan the attack approach and employ the right techniques for the subsequent attack phases. One of the most common and well understood techniques for discovering the range of hosts which are alive in the target's environment is to perform an ICMP sweep of the entire target's network range. An ICMP sweep involves essentially sending a series of ICMP request packets to the target network range and from the list of ICMP replies infer whether certain hosts are alive and connected to the target's network for further probing. This is illustrated in the next Figure.

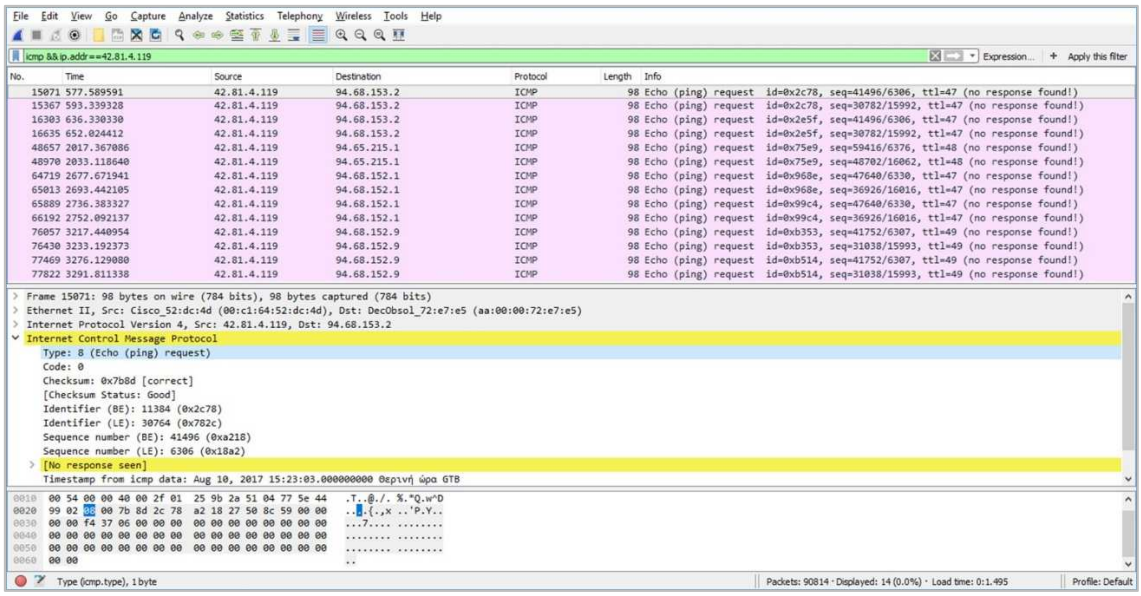


Figure 35: ICMP request packets to the targeted network 94.68.152.0/24 (screenshot from Wireshark)

In this case the attacker sends ICMP echo requests to all the hosts in a network, then he waits to collect the replies and determine which hosts are alive. In a normal ICMP echo request, the packet length would be 42 bytes (and no 98 bytes), where the data length is 0 and if we append any data in to the ICMP data field then the size of the packet increases.

This attack comes to be more attractive if we see the results of next Figure. In distinction from Figure 35 where an attacker sends ICMP echo requests to all the hosts in a network, the graph below shows the attempts of different bots (most of them comes from China) to send ICMP echo requests packets to a specific targeted network 94.68.153.0/24 which is a part of our honeynet.

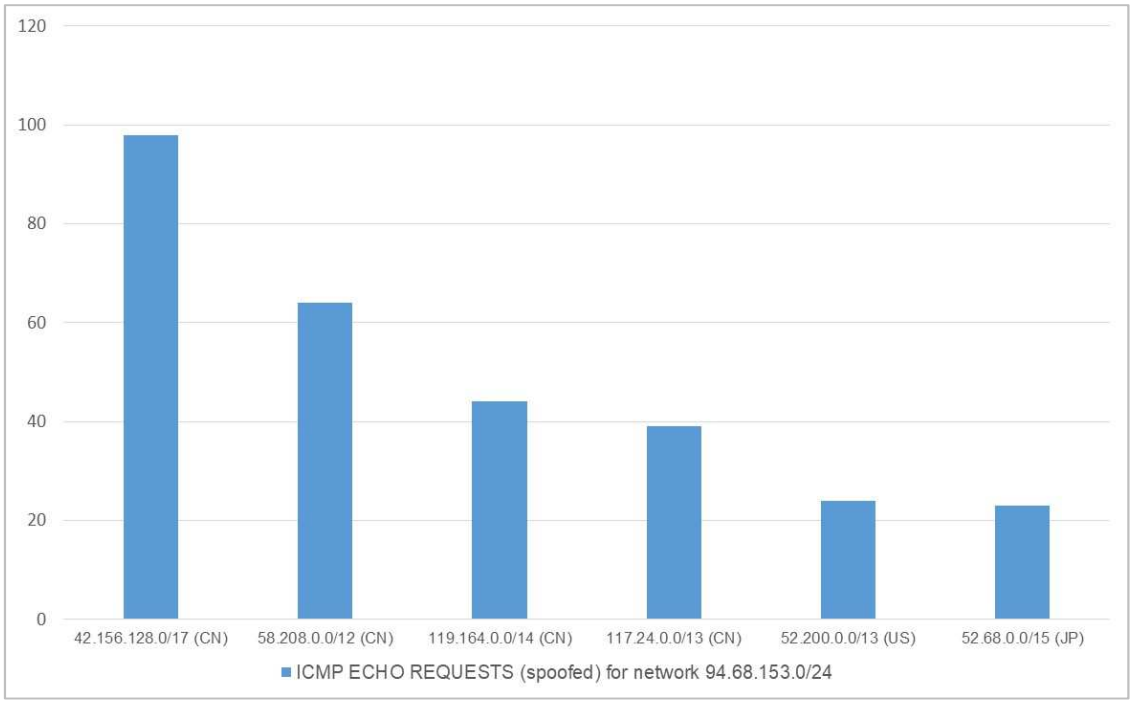


Figure 36: ICMP echo requests for the targeted network 94.68.153.0/24

In the second case, which is illustrated below, the victim (a range of IPs from the network 94.65.215.0/24) started receiving ICMP Echo Reply (reflector attacks) from the attacker with IP 136.243.78.27. In this scenario the attacker sends spoofed (with victim's IP range of addresses) ICMP Echo Replies from every machine.

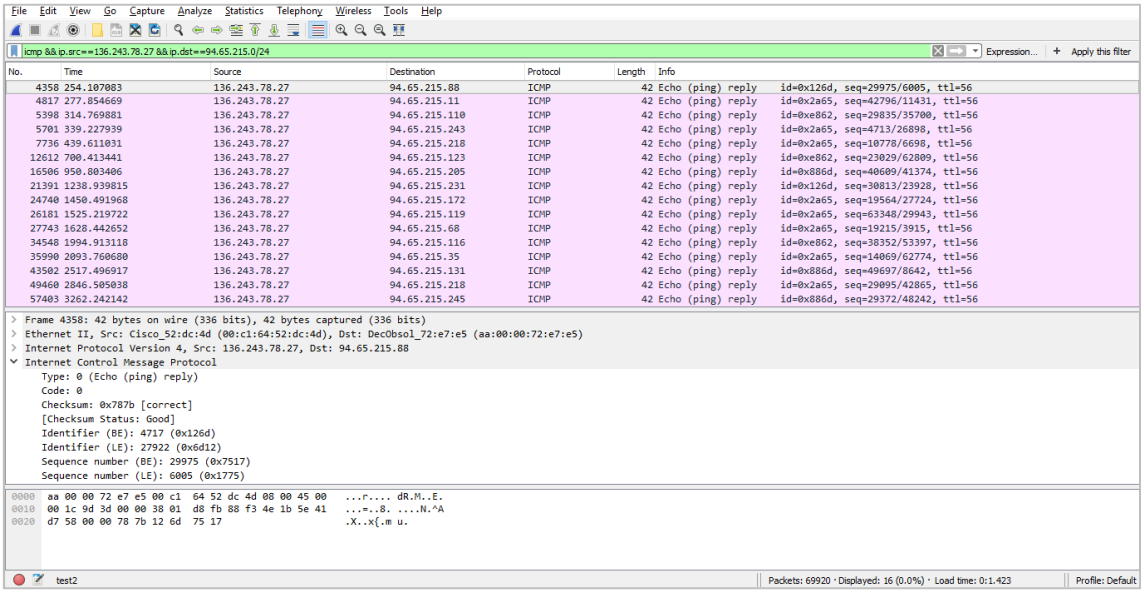


Figure 37: ICMP reply packets for the targeted network 94.65.215.0/24 (screenshot from Wireshark)

Disabling a ping flood is most easily accomplished by disabling the ICMP functionality of the targeted router, computer or other device. A network administrator or security expert can access the administrative interface of the device and disable its ability to send and receive any requests using the ICMP, effectively eliminating both the processing of the request and the Echo Reply. The consequence of this is that all network activities that involve ICMP are disabled, making the device unresponsive to ping requests, traceroute requests, and other network activities [42].

4.2.4 Exploit attempts for Netis Router Backdoor: case study

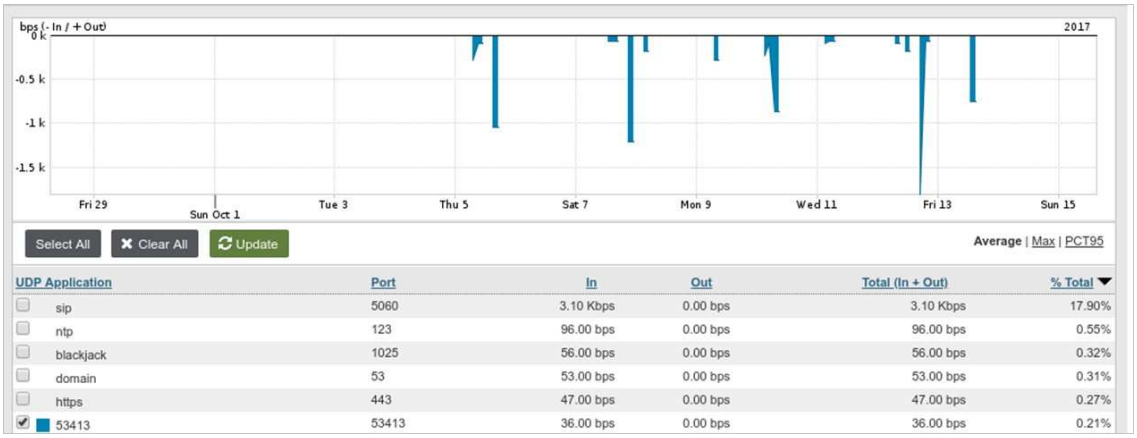
Netis routers, manufactured by Netcore[17], have a wide-open backdoor that can be easily exploited by cyber criminals. This backdoor was an always-open UDP port (listening on port 53413) that allowed an attacker to reach internal network via the router's WAN interface and run arbitrary code on these routers, rendering it vulnerable as a security device.

Trend Micro, the company that discovered the flaw, added specific rules in its security systems to detect exploitation attempts for this backdoor, back in August 2016. Most of these incidents are attempts to find vulnerable devices. These attacks are carried out by automated scanners that look for the vulnerable port and try to authenticate with the backdoor's credentials [59]. The attackers only needed to know the router's external IP address and then could gain access to it through the UDP port 53413, after which they could access the backdoor by entering a password hardcoded in the firmware. With full control over the affected devices, an attacker could modify settings to carry out man-in-the-middle attacks.

The following example highlights a scanning across the IPv4 spaces, looking for Internet-accessible routers that respond to the backdoor probe. From the pcap files we captured, we observed that during the period time from June to August, there wasn't generated traffic to the specific port 53413. This traffic was appeared on September from a unique IP source address 165.227.186.147 which sent UDP packets to port 53413 with a shell command to be executed. Although, it appears that this exploit string (Figure 38) might actually indicate an attempt by

[17] A popular brand for networking equipment in China https://netcore.in/company/

malware to compromise the device, there is no impact on the server side because there is no response from Dark IP spaces (no vulnerable port is open- Chapter 3).



UDP Application	Port	In	Out	Total (In + Out)	% Total
sip	5060	3.10 Kbps	0.00 bps	3.10 Kbps	17.90%
ntp	123	96.00 bps	0.00 bps	96.00 bps	0.55%
blackjack	1025	56.00 bps	0.00 bps	56.00 bps	0.32%
domain	53	53.00 bps	0.00 bps	53.00 bps	0.31%
https	443	47.00 bps	0.00 bps	47.00 bps	0.27%
53413	53413	36.00 bps	0.00 bps	36.00 bps	0.21%

Figure 38: Significant increase of traffic on UDP port 53413 (Source: Arbor Networks, Inc.)

Figure 39 shows the number of packets that detected on port 53413. We believe these are scans by attacker who seek to locate and compromise certain models of Netis router to build a botnet. This is a different criminal behavior, since the attacker is likely seeking to exploit a vulnerability of the host itself rather than detect a UDP service suitable for use in reflection attacks.

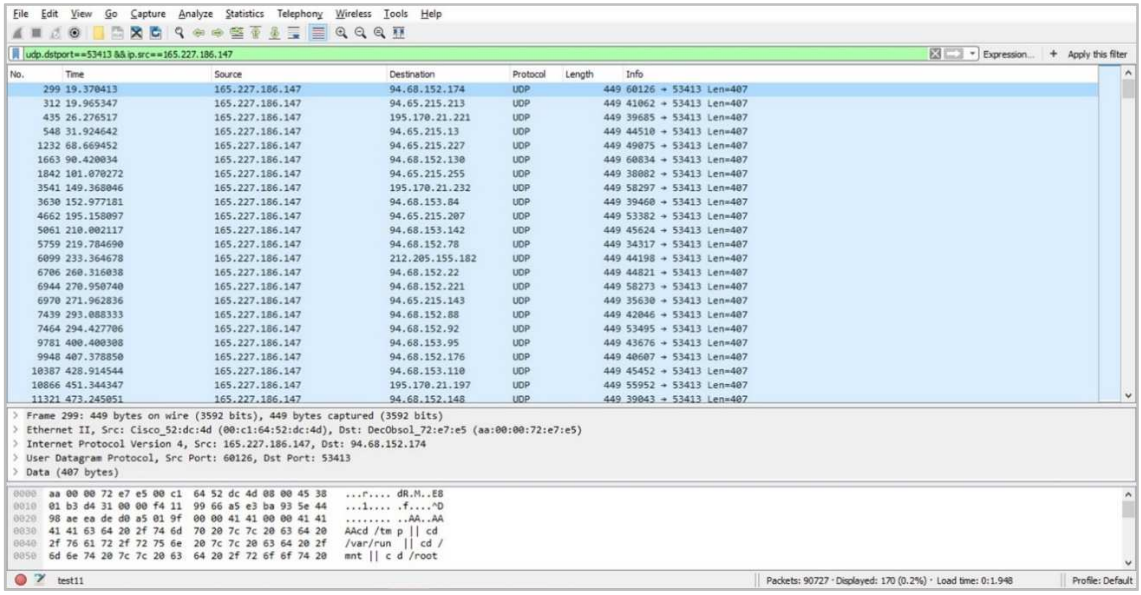


Figure 39: Exploit attempts for Netis Router Backdoor (screenshot from Wireshark)

While a packet payload such as this one resembles an exploitation string, it was distinct because it targeted UDP port 53413, which is not associated with any established service rather than TCP ports 80 or 8080 (the most commonly targeted ports).

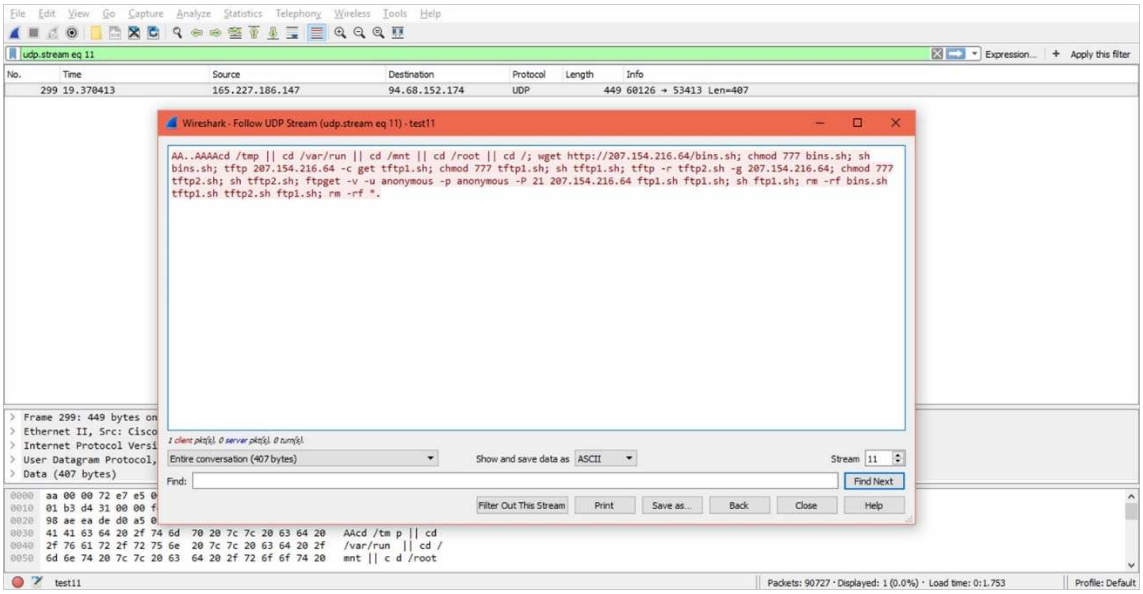


Figure 40: Raw shell exploitation attempt (UDP stream→screenshot from Wireshark)

The IP address of the web/ftp/tftp server above is different from the IP address the attack came from, so unlike other worms, the victim does not appear to be offering the files for download [60]. The below raw shell exploitation indicates an attempt to download some files which matches the affective Netis routers.

```
cd /tmp || cd /var/run || cd /mnt || cd /root || cd /;
wget http://207.154.216.64/bins.sh; chmod 777 bins.sh; sh bins.sh;
tftp 207.154.216.64 -c get tftp1.sh; chmod 777 tftp1.sh; sh tftp1.sh;
tftp -r tftp2.sh -g 207.154.216.64; chmod 777 tftp2.sh; sh tftp2.sh;
ftpget -v -u anonymous -p anonymous -P 21 207.154.216.64 ftp1.sh ftp1.sh; sh ftp1.sh;
rm -rf bins.sh tftp1.sh tftp2.sh ftp1.sh; rm -rf *.
```

First the attacker tries to move into a directory where he has read and write permissions. Thus he uses the double-pipe (||) to so that if the first command fails, the second command is executed and if the second fails, the third is executed etc. If he moves into any of those folders then he's ready to execute the more interesting commands.

Then the attacker is using the tftp command to get a remote file, in this case from the server at 207.154.216.64 and save it to the current directory (this could be /tmp, /var, or /mnt depending on the first command executed) in order to change the permissions with the command **chmod 777 sh;** and be able to execute the .sh script which includes his downloaded payload and the server be compromised.

The problem of the Netis router firmware backdoor hasn't been solved. Since we cannot stop using IoT devices, the easiest way to determine if a router is affected is to probe port 53413 with an online scanner.

Chapter 5: Defending mechanisms of DDoS attacks

5.1 Defense strategies and deployment location

Taking into account the effects of DDoS attacks requires a new approach that not only detects increasingly complex threats but also mitigates these effects in order to ensure business continuity and resource availability. A DDoS protection mechanism is built around four key issues:

1. Not only detect the attacks but mitigate them.
2. Distinguish good traffic from bad traffic to preserve business continuity, not just detect the overall presence of an attack.
3. Include performance and architecture to deploy upstream to protect all points of vulnerability.
4. Maintain reliable and cost-efficient scalability.

For an ISP, it is important the protection mechanisms to be deployed quickly and on a high number of ingress points. A DDoS defense mechanism must deliver the following protection attributes [9]:

- Enables immediate response to DDoS attacks through integrated detection and blocking mechanisms, even during spoofed attacks when attacker identities and profiles are changing constantly.
- Provides more complete verification capabilities than either static router filters or IDS signatures can provide today.
- Delivers behavior-based anomaly recognition to detect valid packets sent with malicious intents to flood a service.
- Identifies and blocks individual spoofed packets to protect legitimate business transactions.
- Offers mechanisms designed to handle the huge volume of DDoS attacks without suffering the same fate as protected resources.
- Enables on-demand deployment to protect the network during attacks without introducing a point of failure or imposing the scaling costs of an inline solution.
- Avoids reliance on network device resources or configuration changes.
- Uses standard protocols for all communications, helping ensure maximum interoperability and reliability.

A DoS attack can easily bring down any unprotected online service. The threat of DoS attacks is increasing dramatically. The reasons for that increased danger is that DDoS attacks get cheaper and easier to initiate every day. There are several reasons and common motives (e.g. hacktivism, vandalism, revenge and politics) that lead people to be DDoS attackers.

The strategies of various DoS defense mechanisms can be divided into four categories: prevention, detection, response, and tolerance. Prevention approaches attempt to eliminate the possibility of DoS attacks or prevent the attack from causing any significant damage. Detection can be further classified as attack detection and attack source identification. Attack detection monitors and analyzes events in a system to discover malicious attempts to cause denial of service. It is an important step before directing further actions to counter an attack. Attack source identification, on the other hand, aims to locate the attack sources regardless of whether the source address field of malicious requests contain erroneous information. Response mechanisms are usually initiated after the detection of an attack to eliminate or minimize the impact of the attack on the victim. Tolerance aims to minimize the damage caused by a DoS attack without being able to differentiate malicious actions from legitimate ones. It might be necessary to merely know the fact that a system is under attack, in order to initiate the tolerance mechanisms [32].

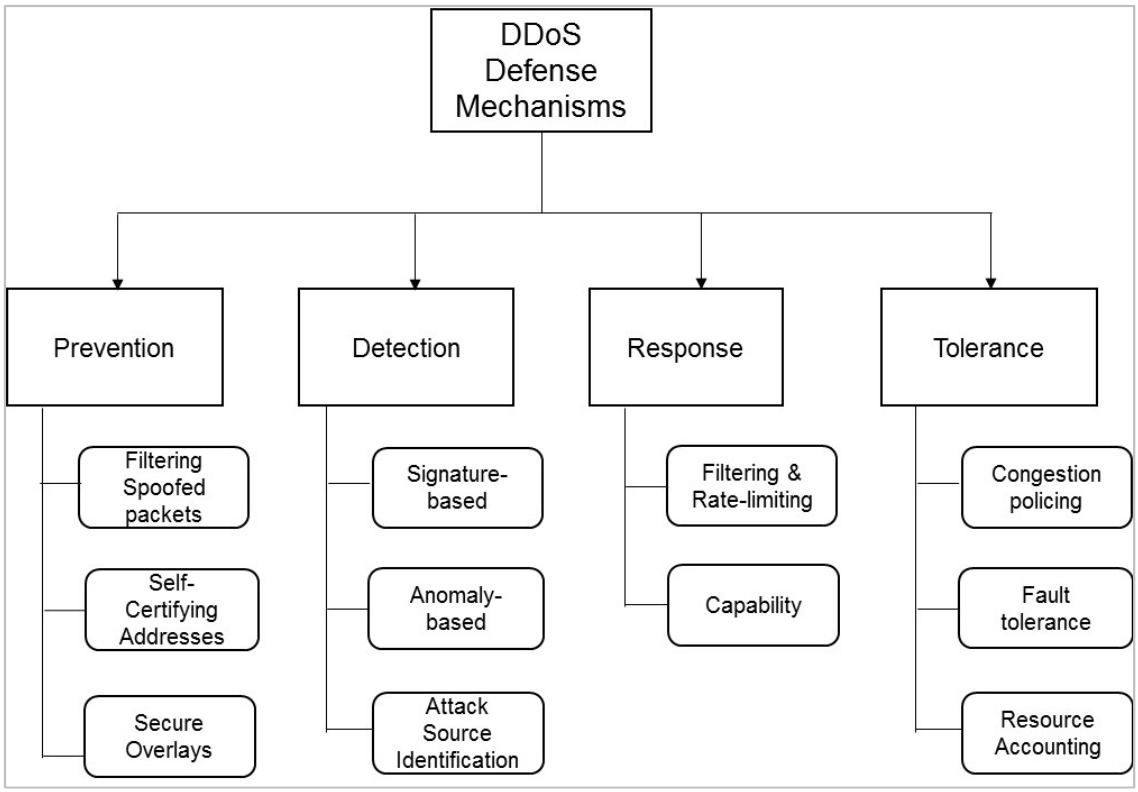


Figure 41: A practical taxonomy of DDoS defense mechanisms [32]

Since DDoS attacks can be defended at different locations in the network, defense mechanisms can also be divided into different categories based on the deployment location. The most common locations are near the target, intermediate network, near the attack source and multiple locations.

Detection is the most important step before directing further actions to counter a DoS attack. DoS response mechanisms depend on the attack related information discovered by detection mechanisms for countering the attack. Some response mechanisms rely on identification of the malicious actions, while others require identifying the entity that is performing the malicious actions. There are also few mitigation mechanisms that depend on discovering the fact that an attack is ongoing, in order to initiate the mitigation process.

Simply discovering the fact that an attack is taking place is usually easy, since the performance degradation of the attack target is easily noticeable. Except, it is rather difficult to differentiate between flash crowds and subtle DoS attacks. On the other hand, identifying the malicious actions is a very difficult task. Generally, there are two groups of detection techniques for identifying malicious actions: signature based detection and anomaly based detection. Detection of malicious actions need to be completed quickly, so that response mechanisms have more time to react before serious damages are made. Furthermore, to block malicious actions near their sources or to take legal actions against the attackers, the source of an attack needs to be identified using attack source identification.

In this section, we look at a vast variety of techniques and tools are used for DDoS attack mitigation. The techniques involve mitigation using Access Control List (ACL), Rate Limiting or combination of both.

5.2 Traditional DDoS mitigation techniques

Some techniques that service providers can use to strengthen the security in their networks are the below:

Blackholing is a common defense strategy used by ISPs to block packet traffic from a domain or IP address and redirecting it into a “null route”. DDoS blackhole routing/filtering, is a countermeasure to mitigate a DDoS attack in which network traffic is routed into a “black hole,” and is lost. When blackhole filtering is implemented without specific restriction criteria, both legitimate and malicious network traffic is routed to a null route or black hole and dropped from the network. At first glance this might seem like a viable mitigation strategy to ward off incoming cyber threats but is this the best way to define DDoS attacks? How does it work?

For organizations that have no other means of blocking an attack, this technique is completely capable of blocking all DDoS attacks in real-time. It has the ability to create an IP traffic route that virtually goes nowhere. The packets destined for the null route end up in the bit bucket. Null routing is essentially available on every commercial router and there is little performance impact to silently drop all traffic to a specific destination. An advantage of this method is that it lets the router do what it was designed to do: route packets.

It’s no secret in the world of DDoS attacks, that using null routing is a tool of choice for organizations to block an attack. For example, if an attacker selects a victim and launches a DDoS attack against them, the victim may not be the only entity impacted. Other users that share the same infrastructure as the victim may also experience the effects of the attack. With no DDoS defenses in place, victims normally call their ISP and ask for assistance with blocking the attack upstream. The ISP injects a null route with the IP address of the original victim into their routing infrastructure and begins blocking all DDoS traffic to the victim with the hopes of reducing the impact against the rest of their customers who are experiencing collateral impact as a result of the attack.

A key consequence of using blackhole routing when good traffic is also affected, is that the attacker has essentially accomplished their goal of disrupting traffic to the target network or service. Even though it can help a malicious actor accomplish their goal, blackhole routing can still be useful when the target of the attack is a small site that's part of a larger network. In that case, blackholing the traffic directed at the targeted site could protect the larger network from the effects of the attack [62].

BGP blackhole filtering is another routing technique used to drop unwanted traffic. Black holes are placed in the parts of a network where unwanted traffic is routed into a “black hole” and then is dropped. When blackhole filtering is implemented without specific restriction criteria, both legitimate and malicious traffic is routed to a null route.

With BGP, two kinds of blackholing can be achieved: blackholing depending on the source IP address of a packet (source-based Remote Triggered Black Hole, S/RTBH) and blackholing depending on the destination IP address of a packet (destination-based Remote Triggered Black Hole, D/RTBH) [61]. Service provides prefer destination-based BGP blackholing to mitigate the damaging effects of DDoS attacks. Specifically, the ISP sets a permanent static route utilizing the unused prefix pointing to the null interface on its PE routers.

It has become common practice to setup "Sinkholes" to capture traffic sent by infected hosts to command and control servers. These Sinkholes are usually established after a malicious domain name has been discovered and registrars agreed to redirect respective NS records to a specific name server configured by the entity operating the Sinkhole.

Once a sinkhole is established, it is possible for the operator of the sinkhole to collect IP addresses from hosts connecting to it. In many cases, a host is only considered "infected" if it transmits a request that indicates it is infected with a specific malware type.

ISP security sinkholes belong to a group of techniques that take advantage of routing protocols as a security tool. A sinkhole is a part of the network that advertises (usually via BGP) certain ranges of IP addresses and attracts traffic destined for those ranges so that it can be analyzed. The IP address ranges that are forwarded to the sinkhole may belong to unused address spaces - private IP ranges or unallocated parts of the provider's address space. Used

in this way, sinkholes facilitate the detection of worms and other attacks that randomly generate packets to unknown addresses.

Depending on the size of the IP blocks advertised by the sinkhole, the sinkhole can attract a lot of junk traffic. For this reason, it is important to carefully choose the size of the address block that the sinkhole will accept. In addition, the network infrastructure supporting the sinkhole must be capable of handling heavy traffic loads. By using anycast, a technique of advertising the same address space from multiple points in the network, it is possible to establish a kind of distributed sinkhole.

Sinkholes allow administrators to perform monitoring using resources that would otherwise need to be centrally deployed rather than distributed throughout the network or embedded into the network infrastructure.

Accordingly, the filtering of malicious traffic appears to be an effective countermeasure against DDoS attacks. It is also important to note that the closer the attacker applies the filtering, the more effective it is. This happens because when traffic is filtered by the victim, the victim continues to operate smoothly but at the same time the ISP's network is already flooded. Therefore, the best solution would be to filter the traffic to its source, which involves filtering the movement of zombie's nodes.

5.3 Mitigating DDoS using Access Control Lists (ACLs)

To protect infrastructure devices and minimize the risk, impact and effectiveness of direct infrastructure attacks, administrators are advised to deploy Access Control Lists (ACLs) to perform policy enforcement of traffic sent to infrastructure equipment. ACLs perform packet filtering to control which packets move through the network and where. One of the most important reasons to configure access lists is to provide a basic level of security for the network by controlling access to it. Such control provides security by helping to limit network traffic, restrict the access of users and devices to the network and prevent traffic from leaving a network. Access lists are defined on a per-protocol basis. Maintaining ACLs at the network level serves to strictly control access to specific ports and protocols. This may not work great at fixing DDOS attacks on port 80, but will prevent unnecessary ports from being open. For example, telnet ports can remain closed and this will protect you for some DDOS attacks.

ACLs are the set of rules that are applied to a machine (e.g. to the edge router of the network) in order to control permissions and stop specific set of IP packets. Such a list provides protection to a network as it controls the traffic moving in and out of that point.

For instance, when an ACL is applied on a router, the incoming IP packets are checked if they satisfy the ACL table before entering. When a packet conforms to an existing rule present in a router, various options like deny, accept, reject could be performed. Thus the ACL table grows in its size, degrading the network performance thus also making it harder to manage.

In an effort to protect routers from various risks—both accidental and malicious—infrastructure protection ACLs should be deployed at network ingress points. These IPv4 and IPv6 ACLs deny access from external sources to all infrastructure addresses, such as router interfaces. Data received by a router can be divided into two broad categories [11]:

1. traffic that passes through the router via the forwarding path
2. traffic destined for the router via the receive path for route processor handling

The anti-DDOS measures are of no use if we allow open access to multiple ports. An authorization system that allows specific network operations and restricts several operations is important. An ACL (access control list) is often used to restrict access. This will prevent easy access to the network for potential DDOS botnet attacks. An authentication system will allow access to parts of the system only to specific IPs/ or a set of IPs.

In order to inhibit scanning for vulnerable IoT devices, it is possible for broadband access network operators to implement ACLs at an appropriate point in the network topology to prohibit high-port TCP traffic destined for TCP/23, 2323,103 on their customer access networks. Such policies would typically be implemented as ingress ACLs on the core interfaces of broadband customer aggregation gateways.

To provide some security benefits of ACLs, we should face up Mirai botnet configuring ACL on the border, located at the edges of the networks, router interface so that inbound traffic or outbound traffic or both are filtered on this interface.

5.4 BGP Flow Specification; a step forward in DDoS mitigation

BGP flow specification (FlowSpec)[18] defines a protocol to rapidly deploy access control lists and forwarding policies (flow-specification filters and actions) amongst all participating routers via a BGP address family. It is considered a new trend and alternative method of blocking unwanted attack traffic from the network. That because it matches a specific flow based on source, destination, Layer 4 properties and packet specific items (length, fragment, e.g).

The BGP Flowspec feature allows us to rapidly deploy filtering and policing functionality among a large number of BGP peer routers to mitigate the effects of a DDoS attack over the network. Routers that support BGP FlowSpec can match packets based on certain characteristics, such as destination prefix, source prefix, IPv4 protocol (e.g. TCP or UDP), destination port, source port and more. Matched packets can be rate-limited, dropped (i.e. rate-limiting to 0), filtered, or redirected [61].

Using BGP FlowSpec provides several benefits in comparison to Access Control Lists (ACLs) on routers [61]:

- Since FlowSpec leverages the BGP control plane, an infrastructure between routers that is already in place, it requires less effort to deploy FlowSpec. Also, adding a new FlowSpec route requires less effort. There is no need to log in on all individual routers to add a new ACL, which results in less repetitive work.
- BGP FlowSpec is standardized while ACLs are configured in a vendor-specific manner. As such it is easier to deploy the same set of rules on routers from different vendors.

The Flowspec router advertises these flows to the other edge routers on the network in order to install the flows into their hardware. Once the flow is installed into the hardware, the transit routers are able to do a lookup to see if incoming traffic matches the defined flows and take suitable action. The action in this scenario is to 'drop' the DDoS traffic at the edge of the network itself and deliver only clean and legitimate traffic to the Customer Edge.

As an example, using this technique it is possible to block a Mirai botnet attack by specifying that all TCP traffic with destination port 23 and 2323 must be dropped.

[18] Defined in RFC 5575 and updated by the IETF https://tools.ietf.org/html/rfc5575

Chapter 6: Conclusions and future work

6.1 Summary

The present time is full of challenges in the world of network security. Everything seems to indicate that the number of threats that users have to deal with will continue to grow, so now more than ever protection is essential. The number of DDoS attacks are constantly increasing over the past year, so it is important for network engineers, designers and operators to create services and monitor networks in the context of defending against them.

DDoS attacks are still a major threat, especially due to the raise of the IoT. In this thesis, we implemented a honeynet at a high-speed network topology of an ISP and used this to gather and analyze data from real DDoS attacks. Honeynets are a powerful security mechanism to capture different types of attacks against the network and monitor efficiently the attackers' activities.

The selection of the appropriate software tools was determined by the variety of factors. Our primary goal has been to detect and record the malicious activities targeting honeynet in an attempt to obtain as much information as possible about the attackers, the services (applications) via they use to get into the network and the vulnerabilities of the network. The response of our honeynet to these attacks doesn't take apart to this thesis because we used unused IP addresses for the implementation so we only wrote down the behavior and reaction from outside systems.

In Chapter 2, it has become known that the most common types of attack against honeynet has to do with DDoS attacks. For this reason, we used as software tools, Wireshark one of the most powerful network protocol analyzers and tcpdump filters. Arbor network platform was our third choice. Arbor's advanced threat solutions deliver complete network visibility through a combination of packet capture and NetFlow technology, enabling the rapid detection and mitigation of malware and malicious insiders. It also delivers market leading analytics for dynamic incident response, historical analysis, visualization and forensics [37].

For our traffic analysis, we focused on the results of visualization tools. We set up a honeynet system to be accessible from outside for a four-month period time. The results have been quite impressive given the short duration of our experiment. Specifically, we were able to:

- Identify the most vulnerable network services and ports within the network infrastructure. The most targeted port proved to be port 23. Attackers attempted to gain access to our network through HTTP and SSH to a great extent, as ports 80 and 22, were also among the top targeted. As it was expected, attackers showed also interest in our emulated database services such as MS-SQL and MySQL.
- Capture and identify different kinds of attacks against our network. A variety of different types of attacks were recorded. Among other, our honeynet faced port scanning attacks, amplification and reflection attacks, some of them are described to case studies section.
- Record valuable information about the attackers. The honeynet gathered important information about the attackers' IP addresses, the countries of origin and other. The majority of those IP addresses were reported as blacklisted. It was not expected that France was among the top source of connection attempts. As expected, Russian Federation and People's Republic of China along with other countries like the USA were among at the top sources of connection attempts.

To that end, the different methods of mitigating DDoS attacks were compared.

6.2 Issues and future scope of work

In our days, service providers such as cloud providers, ISPs as well as large enterprises require an environment that is highly available and secure, as the Internet is the main way to channel an organization's services to its customers. Over the last few years, distributed denial of service (DDoS) attacks have grown dramatically in frequency, size and complexity. While organizations have existing security strategies in place that mitigate a range of security threats, they are not sufficient enough to address new breeds of DDoS attacks that leverage large distributed "botnet" networks of compromised "zombie" machines to simultaneously launch attacks using compliant protocols that are very difficult to detect and even harder to mitigate. It is clear that additional solutions are needed to complement existing security infrastructure in a layered defense model.

DDoS attacks are a complex and serious problem and consequently, numerous approaches have been proposed to counter them. The multitude of current attack and defense mechanisms obscures the global view of the DDoS problem. It is important to recognize and understand trends in attack technology in order to effectively and appropriately evolve defense and response strategies. Even though they have been around for more than a decade, the scale and frequency of these attacks are increasing faster than the capacity of most organizations to absorb them. The attack landscape is changing every day and attackers are deploying new techniques to increase the magnitude of attacks and make them more difficult to mitigate. Protection of high-speed networks and successful mitigation of DDoS attacks is one of the key challenges for internet service providers and backbone operators.

Some open issues that must take into consideration for future work in order to improve the way that service providers check the risk analysis for secure networks are the following:

- Improve the capture of real-time traffic. The real-time bandwidth monitoring allows the early detection of potential attacks and provides the ability to set threshold alarms and alerts for set bandwidth usage.
- It is observed that the number of anomaly packets generated by worms or DDoS attacks is dramatically increasing. These packets typically aim at DoS or attack service vulnerability. If network managers or security experts notice alert information, they should take an appropriate action, such as adding firewall rules or applying patches. Otherwise, their system will have a high risk of services stoppage or intrusion by malicious outsiders. Network managers are responsible for examining alert reports so that they do not overlook important security affairs.
- A Network Management System (NMS) lets network administrators to manage independent components (software and hardware) inside a bigger network management framework. It usually records data from a network's remote points to carry out central reporting to a system administrator. An interesting idea is to examine centralized NMS-based attack analysis from different networks in cooperation with the ISP.
- And of course if we had included more networks (ranges of unused IP addresses) to the honeynet, we could have taken improved results and statistics compared with the existing of Chapter 4.

References

[1] Haris Sinanovic and Sasa Mrdovic “IoT honeypot: a multi-component solution for handling manual and Mirai-based attacks,” 25th Telecommunications forum TELFOR 2017, Serbia, Belgrade, November 21-22, 2017.

[2] Marcin Nawrocki, Matthias Wahlisch, Thomas C. Schmidt, Christian Keil, Jochen Schonfelder “A Survey on Honeypot Software and Data Analysis” 2016.

[3] T. Luo, Z. Xu, X. Jin, Y. Jia, and X. Ouyang, “IoTCandyJar: Towards an intelligent-interaction honeypot for iot devices,” Black Hat, 2017.

[4] Saman Taghavi Zargar, James Joshi and David Tipper, A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks, IEEE Communications surveys & tutorials, p. 1-20, February 2013

[5] K. Angrishi, “Turning internet of things (IoT) into internet of vulnerabilities (Iov): IoT botnets,” CoRR, vol. abs/1702.03681, 2017.

[6] Mart Meijerink, Backbone Network Resources Consumed by DDoS Attacks, University of Twente, Faculty of Electrical Engineering, Mathematics and Computer Science, p.p. 1-3, January 2014

[7] Sigit Haryadi and Jordia Ibrahim, Security Requirements Planning To Anticipate The Traffic Flooding On The Backbone Network, p. 1-4, 2015

[8] Christoph Dietzel, Anja Feldmann and Thomas King, Blackholing at IXPs: On the Effectiveness of DDoS Mitigation in the Wild, International Conference on Passive and Active Network Measurement, p.p. 319-332, March 2016

[9] Defeating DDOS Attacks, Cisco Network Foundation Protection White Papers, updated January 2014, [online] http://www.cisco.com/c/en/us/products/collateral/security/traffic-anomaly-detector-xt-5600a/prod_white_paper0900aecd8011e927.html

[10] Service Provider Security, Cisco Network Foundation Protection White Papers, [online] http://www.cisco.com/c/en/us/about/security-center/service-provider-infrastructure-security.html#2

[11] Protecting Your Core: Infrastructure Protection Access Control Lists, Cisco Network Foundation Protection White Papers, updated 21 October 2008, [online] http://www.cisco.com/c/en/us/support/docs/ip/access-lists/43920-iacl.html#frag

[12] Madalina Baltatu, Antonio Lioy, Fabio Maino and Daniele Mazzocchi, Security Issues in Control, Management and Routing Protocols, TERENA Networking Conference, p. 1-12, May 22-25, 2000

[13] Rajkumar and Manisha Jitendra Nene, A Survey on Latest DoS Attacks: Classification and Defense Mechanisms, International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization), Vol. 1, Issue 8, October 2013

[14] Christian Rossow, Amplification Hell: Revisiting Network Protocols for DDoS Abuse, San Diego, CA, USA, February 2014

[15] Sumesh Shivdas, Automating Provisioning of NetFlow Analyzers, SANS Institute Reading Room site, 7 September 2016

[16] Pierce M Gibbs, Botnet Tracking Tools, SANS Institute Reading Room site, August 2014.

[17] Manish Gupta, Gayathri Gopalakrishnan, and Raj Sharman, Countermeasures against Distributed Denial of Service (A Literature Review), 11th annual symposium on information assurance (ASIA 2016)

[18] K. Munivara Prasad, A. Rama Mohan Reddy & K.Venugopal Rao, DoS and DDoS Attacks: Defense, Detection and Traceback Mechanisms -A Survey, Global Journal of Computer Science and Technology: E Network, Web & Security, Volume 14, Issue 7, Version 1.0, 2014

[19] Youssef Gahi, Junaid Israr and Mouhcine Guennoun, Wormhole Detection in Secured BGP Networks, IEEE 3rd International Conference on Cyber Security and Cloud Computing, 2016

[20] DDoS Quick Guide, National Cybersecurity and Communications Integration Center, 29 January 2014

[21] Bradley Mitchell, What Internet and Network Backbones Do, Updated October 07, 2016 [online] https://www.lifewire.com/definition-of-backbone-817777

[22] J. Mirkovic, J. Martin, P. Reiher, A taxonomy of DDoS attacks and DDoS defense mechanisms, UCLA CSD Technical Report no. 020018

[23] Saket Acharya, Namita Tiwari, Survey of DDoS Attacks Based On TCP/IP Protocol Vulnerabilities, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 18, Issue 3, Ver. IV (May-Jun. 2016), PP 68-76

[24] Valency Networks, Cyber Security Attacks Explained: DoS and DDoS [online] http://www.valencynetworks.com/articles/security-attacks-dos.html

[25] Constantin Oesterling, Denial of Service Attacks: Definition & Prevention, 8 October 2015, blog, https://javapipe.com/denial-of-service-attack#define-dos

[26] A Cisco Guide to Defending Against Distributed Denial of Service Attacks, Cisco Network Foundation Protection White Papers, [online] http://www.cisco.com/c/en/us/about/security-center/guide-ddos-defense.html#3

[27] Subramani Rao Sridhar Rao, Denial of Service attacks and mitigation techniques: Real time implementation with detailed analysis, SANS Institute Reading Room site, 2011.

[28] Lenny Zeltser, Information Security in Business, 9 Reasons for Denial-Of-Service (DoS) Attacks: Why Do They Happen? Updated August 31, 2016, [online] https://zeltser.com/reasons-for-denial-of-service-attacks/#

[29] Youksamay Chanthakoummane, Saiyan Saiyod, Nunnapus Benjamas and Nattawat Khamphakdee, Evaluation Snort-IDS Rules for Botnets Detection, National Conference on Information Technology: NCIT, 2015

[30] Rawal, B., Ramcharan, H., Tsetse, A. (2013). Emergence of DDoS resistant augmented Split architecture. 10th International Conference on High Capacity Optical Networks and Enabling Technologies. 37-43

[31] Tao Peng, Christopher Leckie and Kotagiri Ramamohanarao. Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems. ACM Computing Surveys, Vol. 39, No. 1, Article 3, Publication date: April 2007

[32] Mehmud Abliz, Internet Denial of Service Attacks and Defense Mechanisms. University of Pittsburgh Technical Report, No. TR-11-178, March 2011, Pages 1–50

[33] Tobias Knecht, Cyber Security is an ISP's Top Priority. The Abusix Blog, posted on 11/08/2016, [online] https://www.abusix.com/blog/cyber-security-is-an-isps-top-priority

[34] Stephen M. Specht, Ruby B. Lee, Distributed Denial of Service: Taxonomies of Attacks, Tools and Countermeasures. International Workshop on Security in Parallel and Distributed Systems, pp. 543-550, September 2004

[35] Packet Header Analysis, Insecure Lab India, 2014 blog [online] http://www.insecure.in/packet_header_analysis.asp

[36] Tim Shimeall, Traffic Analysis for Network Security: Two approaches for going beyond Network Flow Data, posted on September 16, 2016, SEI blog [online] https://insights.sei.cmu.edu/sei_blog/2016/09/traffic-analysis-for-network-security-two-approaches-for-going-beyond-network-flow-data.html

[37] Arbor Networks, Hellenic Telecommunication Organization joins Arbor Networks' cloud signaling coalition to stop DDoS attacks, Burlington, MA, March 24, 2014, [online] https://www.arbornetworks.com/hellenic-telecommunications-organization-ote-joins-arbor-networks-cloud-signalingsm-coalition-to-stop-ddos-attacks

[38] Roland Dobbins, Steinthor Bjarnason, Mirai IoT Botnet Description and DDoS Attack Mitigation. Published on October 2016 [online] https://www.arbornetworks.com/blog/asert/mirai-iot-botnet-description-ddos-attack-mitigation/

[39] Ivo van der Elzen, Jeroen van Heugten, Techniques for detecting compromised IoT devices, Research Project, February 12, 2017

[40] SSDP DDoS Attack Mitigation: Radware Emergency Response Team, Publication date: November 2014

[41] SSDP Attack Learning Objectives, Cloudflare Inc. [US] blog [online] https://www.cloudflare.com/learning/ddos/ssdp-ddos-attack/

[42] Ping (ICMP) Flood Learning Objectives, Cloudflare Inc. [US] blog [online] https://www.cloudflare.com/learning/ddos/ping-icmp-flood-ddos-attack/

[43] Harish Kapri, Network Traffic Data Analysis. Master thesis, Nagpur India, December, 2011

[44] Types of DDoS attacks, Verisign iDefense Threats & Trends Report, 2015 [online] https://www.verisign.com/en_IN/security-services/ddos-protection/types-of-ddos-attacks/index.xhtml?loc=en_IN

[45] Archana Kesavan, Three Types of DDoS Attacks, November 15th, 2016, blog [online] https://blog.thousandeyes.com/three-types-ddos-attacks/

[46] Eir's d1000 modem is wide open to being hacked, kenzo2017. Reverse engineering blog [online] https://devicereversing.wordpress.com/2016/11/07/eirsd1000-modem-is-wide-open-to-being-hacked/

[47] DDoS Quick Guide, National Cybersecurity and Communications Integration Center, 29 January 2014, white paper

[48] Alexandre Dulaunoy, Gérard Wagener, Sami Mokaddem, Cynthia Wagner, An extended analysis of an IoT malware from a blackhole network, research paper/case study.

[49] Tod Beardsley, Mirai FAQ: When IoT Attacks, Rapid7 Blog, Oct 24, 2016 [online] https://blog.rapid7.com/2016/10/24/mirai-faq-when-iot-attacks/

[50] Mirai botnet hammers college with two-day attack, 2017 Pindrop blog [online] https://www.pindrop.com/blog/mirai-botnet-hammers-college-with-two-day-attack/

[51] Mirai source code release leads to huge increase in botnet, 2017 Pindrop blog [online] https://www.pindrop.com/blog/mirai-source-code-release-leads-to-huge-increase-in-botnet/

[52] Waqas Amir, Meet the Leet DDoS Botnet, Just as Powerful as Mirai. Published on December 29 2016, HackRead [online] https://www.hackread.com/meet-the-leet-ddos-botnet-as-powerful-as-mirai/

[53] Catalin Cimpanu, 650Gbps DDoS Attack from Leet Botnet Rivals Mirai Attacks. Published on December 28 2016, BleepingComputer [online] https://www.bleepingcomputer.com/news/security/650gbps-ddos-attack-from-leet-botnet-rivals-mirai-attacks/

[54] Claud Xiao, Cong Zheng and Yanhui Jia, New IoT/Linux Malware Targets DVRs, Forms Botnet. Palo Alto Networks, published on April 6, 2017 [online] https://researchcenter.paloaltonetworks.com/2017/04/unit42-new-iotlinux-malware-targets-dvrs-forms-botnet/

[55] TCP SYN Flood (attack description), Imperva Incapsula blog [online] https://www.incapsula.com/ddos/attack-glossary/syn-flood.html

[56] Ping Flood (ICMP Flood) (attack description), Imperva Incapsula blog [online] https://www.incapsula.com/ddos/attack-glossary/ping-icmp-flood.html

[57] DDOS attacks, Imperva Incapsula blog [online] https://www.incapsula.com/ddos/ddos-attacks/

[58] Slowloris DDoS Attack Learning Objectives, Cloudflare Inc. [US] blog [online] https://www.cloudflare.com/learning/ddos/ddos-attack-tools/slowloris/

[59] Catalin Cimpanu, Security Firm Detects 57M Attempts to Exploit 2-Year-Old Router Firmware Backdoor, published on November 21, 2016 blog Bleepingcomputer [online] https://www.bleepingcomputer.com/news/security/security-firm-detects-57m-attempts-to-exploit-2-year-old-router-firmware-backdoor/

[60] Johannes Ulrich, Surge in exploit attempts for Netis router backdoor (UDP/53413), 2016 SANS ISC InfoSec Forums [online] https://isc.sans.edu/forums/diary/Surge+in+Exploit+Attempts+for+Netis+Router+Backdoor+UDP53413/21337/

[61] C.J.T.M. Schutijser “Comparing DDoS Mitigation Techniques” 2016 [online] http://referaat.cs.utwente.nl/conference/24/paper/7526/comparing-ddos-mitigation-techniques.pdf

[62] A DDoS mitigation strategy that works by eliminating all traffic from certain sources, Cloudflare, 2017 [online] https://www.cloudflare.com/learning/ddos/glossary/ddos-blackhole-routing/