Investigation of Open Resolvers in DNS Reflection DDoS Attacks

Mémoire

Saeed Abbasi

Maîtrise en informatique
Maître ès sciences (M.Sc.)

Québec, Canada

Résumé

Les serveurs du système de noms de domaine (DNS) représentent des éléments clés des réseaux Internet. Récemment, les attaquants ont profité de ce service pour lancer des attaques massives de déni de service distribué (DDoS) contre de nombreuses organisations [1, 2, 3]. Ceci est rendu possible grâce aux différentes vulnérabilités liées à la conception, implantation ou une mauvaise configuration du protocole DNS. Les attaques DDoS amplifiées par DNS sont des menaces dangereuses pour les utilisateurs d'Internet. L'objectif de cette étude est d'acquérir une meilleure compréhension des attaques DDoS amplifiées par DNS par l'investigation des résolveurs DNS ouverts à travers le monde. Dans ce contexte, il est nécessaire d'adopter une approche en phase précoce pour détecter les résolveurs DNS ouverts. Cela devient cruciale dans le processus d'enquête. Dans cette thèse, nous nous intéresserons à l'utilisation de résolveurs DNS ouverts dans les attaques DDoS amplifiées par DNS. Plus précisément, la principale contribution de notre recherche est la suivante : (i) Nous profilons les résolveurs DNS ouverts, ce qui implique : détecter les résolveurs ouverts, les localiser, détecter leur système d'exploitation et le type de leur connectivité, et étudier le but de leur vivacité. (ii) Nous effectuons une évaluation de la sécurité des résolveurs DNS ouverts et leurs vulnérabilités. De plus, nous discutons les fonctions de sécurité des résolveurs DNS, qui fournissent, par inadvertence, les attaquants par la capacité d'effectuer des attaques DDoS amplifiées par DNS. (iii) Nous présentons une analyse pour démontrer l'association des résolveurs DNS ouverts avec les menaces de logiciels malveillants.

Abstract

Domain Name System (DNS) servers represent key components of Internet networks. Recently, attackers have taken advantage of this service to launch massive Distributed Denial of Service (DDoS) attacks against numerous organizations [1, 2, 3]. This is made possible due to the various vulnerabilities linked to the design, implementation or misconfiguration of the DNS protocol. DNS reflection DDoS attacks are harmful threats for internet users. The goal of this study is to gain a better understanding of DNS reflection DDoS attacks through the investigation of DNS open resolvers around the world. In this context, there is a need for an early phase approach to detect and fingerprint DNS open resolvers. This becomes crucial in the process of investigation. In this thesis, we elaborate on the usage of DNS open resolvers in DNS reflection DDoS attacks. More precisely, the main contribution of our research is as follows : (i) We profile DNS open resolvers, which involves : detecting open resolvers, locating them, fingerprinting their operating system, fingerprinting the type of their connectivity, studying the purpose of their liveness. (ii) We conduct an assessment with respect to DNS open resolvers security and their vulnerabilities. Moreover, we discuss the security features that DNS open resolvers are equipped with, which inadvertently provide the capability to the attackers in order to carry out DNS reflection DDoS attacks. (iii) We present an analysis to demonstrate the association of DNS open resolvers with malware threats.

Table des matières

Liste des tableaux

Liste des figures

If you don't build your dream,
someone else will hire you to help
them build theirs.

Dhirubhai Ambani

Acknowledgements

I would like to hereby express my deepest gratitude to my supervisors, Professor Mourad Debbabi and Professor Mohamed Mejri for providing me with the special opportunity to work with them in the fast-growing area of network security. I really appreciate all the time, support and effort, which they have invested in order to help me with my thesis. Especially, I would like to acknowledge their guidance and support throughout the period of my Master studies both in academic and non-academic situations.

With all my sincere feelings, I would like to extend my appreciations to my parents, Hajar and Askar, my two lovely sisters, Sepideh and Sussan, and my best brother, Soroush and his wife Mahtab for their unconditional support in my life. I owe them all that I am. I dedicate this thesis to all of them. As well, I thank to all my friends and family members.

I am also thankful to my colleagues of the laboratory namely, Sujoy Ray, Andrei Soeanu and Mina Sheikh Alishahi for their valuable academic advice and personal support. Special thanks goes to my best friend Hossein Ghafari for his sustained help and support.

Acronyms and Abbreviations

API	Application Programming Interface
ARIN	American Registry for the Internet
AS	Autonomous Systems
BAF	Bandwidth Amplification Factor
BIND	Berkeley Internet Name Domain
C&C	Command and Control
CH	Chaos
DDAA	Detecting DNS Amplification Attacks
DDoS	Distributed Denial of Service
DIY	Do-it-Yourself
DNS	Domain Name service
DNSSEC	Domain Name System Security Extensions
DRDoS	Distributed Reflection Denial of Service
DS	Delegation Signer
FQDN	Fully Qualified Domain Name
FTP	File Transfer Protocol
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
IPS	Intrusion Prevention Systems
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IRC	Internet Relay Chat
ISP	Internet Service Provider
NSEC3	Next Secure 3
NTP	Network Time Protocol

PUP	Potentially Unwanted Program
RFC	Request for Comments
RR	Resource Record
RRL	Response Rate Limiting
RRSIG	Resource Record Signature
SNMP	Simple Network Management Protocol
SOA	Start of Authority
TCP	Transmission Control Protocol
TLD	Top Level Domain
UAC	User Account Control prompts
UDP	User Data-gram Protocol
URL	Uniform Resource Locator
VOIP	Voice Over IP
WAP	Wireless Application Protocol

Chapitre 1

Introduction

DNS represents a core interaction of most Internet activities, and it is a key component of network infrastructure. DNS is an Internet service that translates the domain names into numerical IP address. Therefore, Internet users do not need to remember the IP addresses of each website, and they need to know only the domain names. DNS is a fundamental piece of any action in the Internet, and therefore it fascinates attackers given its potential to facilitate the perpetration of malicious actions. The UDP based protocols such as DNS and NTP are frequently abused by attackers, and the current incidents show that such UDP based protocols are relatively easy to exploit [1, 2, 3, 4]. For example, Spamhaus [1] is a non-profit international organization whose the main mission is to track email spammers and spam-related activities and to prepare dependable real-time anti-spam protection. This organization was a target of DNS reflection DDoS attacks. It was the biggest DDoS (distributed denial of service) attacks based on DNS reflection in the Internet history, and it broke the Spamhaus infrastructure [2]. DDoS occurs when many machines (usually bot clients, which are machines that previously got compromised by attackers with malware, which is a type software that is acting via coordinated command and control and under control of attackers) try to submerge an entire network, a machine or a particular service by initiating a tremendous amount of requests involving a huge quantity of data in order to slow down or make access impossible to legitimate users. A DNS open resolver represents a name server that provides a recursive name resolution for non-local users. Thus, it accepts recursive queries from users located all around the world without restriction to the local users who are trusted.

When combined with DNS reflection, these attacks become difficult to stop even by organizations/companies with heavily armed security infrastructures. Sending DNS requests that generate a much larger response can produce the amplification effect. Reflection, on the other side, happens when an attacker can spoof the IP source address so that the answers will be returned to a specific target instead of the originator of the query.

1. http ://www.spamhaus.org/

DNS amplification [5] can be mixed with source IP address spoofing to provide more impact whereby the attacker can access a large number of DNS open resolvers with a good amplification factor. DNS amplification attacks represent a real threat for Internet security, and efforts such as DNSSEC[6] (DNSSEC is a group of extensions that adds several security factors to the DNS protocol. The main aspect is to add to DNS data integrity, authenticated denial of existence and origin authority. DNSSEC is designed to cryptographically sign DNS zone and DNS records.) do not offer solutions against them.

DNS open resolvers are often used in reflected denial of service attacks[7, 8]. In these types of cyber attacks, an attacker orders the bots under his control to send DNS queries to a list of DNS open resolvers ; bots will spoof IP addresses with the victim address, and all the DNS responses will be sent to the victim. In this research study, we investigate DNS open resolvers with respect to DNS reflection DDoS attacks and we provide our assessment thereof. Our analysis includes their platforms, geolocation information, device types and purpose, distinguished individual or corporate name servers, their security measurements as well as their vulnerabilities. Moreover, by providing several analyses with respect to Malware blacklist databases we show that a portion of DNS open resolvers are malicious and had been correlated with cyber crime activities.

1.1 Motivation

Domain Name Service (DNS) servers represent key components of the Internet networks. Recently, attackers have taken advantage of this service to launch massive distributed denial of service (DDoS) attacks against public and private organizations and business services. The goal of this thesis is to draw up a profiling of existing DNS open resolvers around the world and examine their security with respect to reflection DDoS attacks.

In this context, DDoS attacks target systems, networks and individual services, with disruptive traffic, which can either crash software systems or make services and operations unavailable. As a result, such attacks effectively make the services unavailable (denied) for trusted and legitimate users. These types of attacks can result from the DNS server's nature and the availability of DNS servers in all networks. Thus, DNS servers are a desirable resource for attackers to take advantage of their vulnerabilities and security problems. Moreover, by exploiting these security issues, attackers can launch attacks based on this critical service. This is due to various vulnerabilities [1, 2, 3] linked to the DNS protocol, its implementations or its misconfigurations. DNS reflection is a method for an attacker to intensify DDoS attacks by the amount of traffic they can target to a selected victim.

Nowadays, one of the network security community concerns is shutting down the DNS open resolvers as a main source of DNS reflection DDoS attacks [9]. There is a protocol feature behind the DNS reflection DDoS attacks which allows for a small DNS query to generate a

much larger response. These types of attacks can be combined with source IP address spoofing. The compromised machines (bot clients) are equipped with a piece of malware. The latter triggers (just after receiving the command for DNS reflection DDoS attacks) the spoofing of their IP address with the victim IP address and as the result, an attacker can transmit a large volume of DNS traffic to a target system by driving such DNS queries. In this pursuit, attackers need to find a sufficient number of DNS servers with security vulnerabilities in order to reach their goal ; these security vulnerabilities allow the possibility to answer queries unconditionally, regardless of the source network that the query is originating. An attacker who is armed with a botnet network, a sufficiently large list of DNS open resolvers and who is able to use IP source spoofing, could launch a harmful distributed reflection denial of service attack. Our research aims at providing detailed insights about DNS open resolvers, including their platforms, geolocations, distinguished individual or corporate name servers, device types and purpose (the main purpose of DNS resolvers is to provide name resolution for their clients ; they are mostly hosted by ISPs and are based on operating systems such as Linux and Windows. This is the regular purpose for DNS resolvers. Moreover, another category exists which does not have the domain naming service as their main purpose ; these include firewalls, WAPs, Storage-MISCs, VOIPs, broadband routers, etc.). In addition, we explore DNS open resolvers security assessment, their legitimate use and their vulnerabilities. We cross-correlate the identified open resolvers with a malware database in order to discover, which malware samples they have been associated with during their period of operation.

1.2 Problem Statement

In this dissertation, we study a specific profiling and detection approach for security assessment of DNS open resolvers, with respect to DNS reflection DDoS attacks. Additionally, we investigate the association of DNS open resolvers with malware families, which can exploit DNS open resolvers as platforms for malicious activities.

The main research questions considered in this thesis are :

1. *Question 1 :* Is it possible to elaborate an approach to detect and profile the DNS open resolvers ? If so, can we characterize them based on geolocation, platforms, device type and purpose, their ownership (individual or corporate) ?
2. *Question 2 :* What DNS security features inadvertently provide capabilities for attackers to carry out DNS reflection DDoS attacks ?
3. *Question 3 :* What types of abuse and malicious activities (based on malware samples) DNS open resolvers are subjected to and/or linked with ?

1.3 Objectives

Our research objectives in the context of detecting and profiling of DNS open resolvers involve :

- Investigating the amount of available DNS open resolvers ;
- Geo-locating the DNS open resolvers ;
- Finding their ownership and hosting information ;
- Elaborating on their platform fingerprinting ;
- Discovering the device types and purposes ;
- Exploring the connection speed to identify whether they are individual or corporate name servers.

With respect to different aspects of the DNS open resolvers security assessment, we aim at investigating :

- Root zone attacks (upward referrals) ;
- BIND Authors vulnerability ;
- DNS server software version distribution vulnerabilities.

Concerning DNS open resolvers association with malware activities :

- Assessing DNS open resolvers and malicious traffic ;
- Discovering malware families that abused DNS open resolvers ;
- Performing protocol analysis using malware sample traffic associated with DNS open resolvers.

1.4 Contributions

In this dissertation, we provide core insights related to DNS reflection DDoS attacks. These insights are instrumental in DNS open resolvers profiling and investigation. In addition, we elaborate on DNS open resolvers security assessments and association of DNS open resolvers with malware samples. More precisely, our contribution is mainly threefold :

- Investigating DNS Open Resolvers ;
- Security Assessments for DNS Open Resolvers
- DNS Open Resolvers corroboration with malware samples.

1.4.1 Investigating DNS Open Resolvers

The contribution of this part is the profiling and investigation DNS open resolvers. We propose an approach to detect DNS open resolvers around the world. We explore DNS security threats, and construct a profile including the details on geo-locating of DNS open resolvers, IP allocation for DNS open resolvers, connection speed of DNS open resolvers, device type and purpose of DNS open resolvers as well as their platforms.

1.4.2 Security Assessments for DNS Open Resolvers

The contribution of this part consists in evaluating the security features that open DNS resolver servers are equipped with. We probe DNS features that can inadvertently provide the capability for attackers to carry out DNS reflection DDoS attacks. In Section 4.1, 4.2 we also highlight potential misconfiguration issues, security problems and features in DNS servers which could allow attackers to perform the reflection part of DNS attacks. Furthermore, what is more, such security problems are not only encountered in open DNS resolver servers and we show that there are vulnerabilities even in authoritative name servers that allow attackers to abuse them in DNS reflection DDoS attacks. More precisely, we analyzed the following :

- Root zone attack (upward referral), which is made possible by certain type of misconfiguration whereby a feature that should normally be disabled in authoritative name servers allows these servers to answer queries asking for root server addresses.
- BIND Authors feature, which is employed by BIND[10] name servers, irrespective of their intended role (authoritative or resolver) in order to reply to regular queries for the list of BIND authors. This can be viewed as a vulnerability since it allows attackers to use regular BIND Authors queries with spoofed IP address in order to drive malicious traffic to their victim. To the best of our knowledge, this security issue has not been previously investigated in the literature. This type of request is rather small (the average request size is 38 bytes), but the response is quite large (average responses size is 443 bytes). This allow to generate DNS reflection DDoS attacks. From an attacker's point of view, this security vulnerability can provide an alternative replacement for low bandwidth DNS open resolvers.

1.4.3 DNS Open Resolvers Corroboration with Malware Samples

The contribution of this part consists in analyzing DNS open resolvers by corroborating their information to malware database. Our analysis provides relevant insights with respect to the type of malicious actions that the DNS open resolvers are linked to during the time that they are active. Moreover, we show that a portion of DNS open resolvers are exhibiting malicious behavior and they are contributing to malicious activities.

1.5 Thesis Structure

The remainder of thesis is organized as follows. Chapter 2 presents a comprehensive review of the main areas of research that play an important role in DNS reflection DDoS attacks. In this context, we provide a comparative study of the relevant contributions in this area with respect to related work. Chapter 3, illustrates the proposed approach for DNS open resolvers investigation and profiling. We discuss about the threats to/from DNS. Then, we describe our approach to detecting and profiling DNS open resolvers. This includes geo-location, platforms,

device types and purposes, connection speed and their liveness. Chapter 4 explores the security assessment and association of the investigated DNS open resolvers with malware samples available in malware databases. Chapter 5 briefly summarizes our achievements and draws the conclusion. In addition, it comments on possible future extensions.

Chapitre 2

Literature Review

In this chapter, we review the main research areas related to our study. Many techniques have been deployed during the last years to mitigate DNS reflection DDoS attacks. Disabling open recursion on name servers and only accepting recursive DNS from trusted sources or local clients could help to reduce the DNS reflection attacks. In this chapter, first, we explain the Domain Name System (DNS) in more detail. Second, we expound DNS reflection DDoS attacks thoroughly. Finally, we provide some information about the Botnet networks. Moreover, for each topic we provide the relevant related work.

2.1 Domain Name System

As mentioned in the previous chapter, the main component in this research is focused on DNS. The idea behind the DNS reflection DDoS attacks is coming from security issues related to this service, which requires a better understanding.
DNS is a globally distributed, extensible, hierarchic, and dynamic database that provides a mapping between names and IP addresses (both IPv4 and IPv6). More precisely, it translates domain names (meaningful and user-friendly names) to numerical IP addresses needed for the purpose of locating computer devices around the world. A DNS server can be queried using both the User Datagram Protocol (UDP) and Transmission Control Protocol (TCP). The UDP is one of the main representatives of Internet protocols that is used to transmit short messages named datagrams. It is unreliable and does not utilize handshaking dialogs for reliability, data integrity. The UDP is a connectionless protocol. The TCP is one of the key protocols in TCP/IP networks which allow hosts to establish connections and exchange information over the network. TCP is a connection oriented protocol and guarantees that packets will be delivered. It is worthy to mention that both protocols use port 53. For performance reasons, most queries use the UDP protocol with a block-size limited to 512 bytes. Since DNS is a key component of the Internet network, this service is used by almost all network based applications. When some DNS servers are out of service, some website(s), email server(s) and

other network applications are not available by simply using their names. However, if all the root DNS servers are out of service (by being victims of DDoS attacks), eventually the entire Internet will be out of DNS service. While not very frequently heard about, effective DDoS attacks against root servers and DNS servers are often used by hackers to make their attacks more effective against other services and in particular, against websites.

2.1.1 Service versus Server

The DNS System is basically designed to provide a service. Each of the individual name servers represents a portion in the full chain of the DNS hierarchy. In some cases, there are name servers which play a more demanding role rather than other name servers, as long as they are at a higher level in the DNS hierarchy.

2.1.2 DNS Functional Types

DNS functions play an important role in this type of attacks and each type of DNS functions has its own security problems. Interestingly, most of the problems are coming from DNS open resolvers. There are several security problems with authoritative name servers which we explain in the next chapter. To this end, we need to know what exactly are DNS functional types. With respect to the DNS purpose, DNS servers are divided into two groups :

- Authoritative Name Servers : Particularly configured, they maintain entries to answer queries according to specific domains ;
- Recursive DNS Servers : They place queries on behalf of the clients when they do not already have the required mapping stored in the cache from other name servers.

2.1.3 Authoritative Name Servers

Authoritative name servers maintain and serve a fragment of the global DNS databases for companies and organizations and even individuals with respect to their 'authoritative' zones, for instance “ulaval.ca". These types of name servers provide any name resolution within their fragment of name space such as “*.ulaval.ca". In the case where they are unable to provide any results to the clients placing queries, an error message is sent regarding the query, i.e., “NXDOMAIN". An authoritative name server should return answers only to queries about domain names that have been specifically configured. In Section 4.1, some serious issues are presented regarding the authoritative name servers which are not well configured. Consequently, they could be part of DNS amplification DDoS attacks by answering to particular queries and these issues are names as Upward Referrals.

2.1.4 Recursive DNS Servers

A recursive DNS server executes DNS resolution on behalf of its clients and when it obtains the answer to the queries, it stores the responses in its cache for subsequent queries potentially from other users. When an Internet client enters a URL (Uniform Resource Locator) in his browser, a query is sent to translate that URL into the correct IP address of the website, in order to establish a connection to the website. To do so, the browser has to send the query to recursive DNS servers. This type of DNS servers provides the necessary information for web users.

2.1.5 DNS Role Types

DNS servers can play a variety of roles. A name server could be a master for some zones, a slave server for others, and maybe configured to provide caching or forwarding services for others as well. There are several security misconfigurations about DNS roles that could bring significant security problems for DNS administrators. To assess these misconfigurations, we need to know what are the specific features of the aforementioned roles. In Section 4.5 we provide various recommendations to make DNS configurations secure.

2.1.6 Primary and Secondary Name Servers

A primary DNS server (Master), also known as a zone master, contains zone files for the DNS authoritative name server, and it stores all databases in the local system. The term master comes from the location of the zone file databases. It is probable that the master tries to perform zone transfer to secondary or slave DNS servers. All the change should happen in Master DNS server. Slave servers get new updates by a procedure mechanism, namely automatic update. All the slave servers have one identical copy of the zone file records. Secondary name servers have a zone database information from a primary name server. The two main advantages for having a slave name server is sharing the traffic load to improve the DNS availability and server backup in case the primary name server fails. The slave name server must use the refresh and expiry values from the SOA (Start of Authority) RR (Resource Record) to ask for the zone data.

2.2 Domain Name System Security Extensions

In this section, we explain one aspect of DNS security which unfortunately does not prevent DNS reflection DDoS attacks, but it could amplify these types of attacks because of the Extension Mechanisms for DNS (EDNS0) feature, which is described next. As the result of this feature, the block size of DNS could be increased by this feature to 4096 (4K) bytes while the standard DNS block size is under 512 bytes. DNSSEC is an extended version of DNS aiming to provide some of the necessary security features for these sensitive services. DNSSEC pro-

vides extra security for data integrity and data origin authentication in the DNS protocol. By signing data before sending and using public key cryptography, it makes the receiver assured that the data comes from the trusted source and was not altered during transmission. The data itself is not encrypted and could be read by anyone sniffing the message. A DNS server which is equipped with DNSSEC has public and private key pairs for each record. It strongly prevents responses to be tampered. From experimental results, signatures are almost impossible to fake without access to the private keys. Unfortunately, the recent security bug in OpenSSL protocol, called “HeartBleed" found by Google security researchers, let the attackers exploit the heartbeat extension in OpenSSL protocol. Attackers could access some parts of the memory in a server which hosts OpenSSL and as a result, this security flaw allows attackers to access the private keys and other sensitive information. Internet security researchers announced that this security flaw affected most of the websites, because more than 66% of the websites use this open-source encryption technology [11].
The Domain Name Security Extensions (DNSSEC) standard is specified in several IETF RFCs : [12, 13, 14, 15]. However, unfortunately DNSSEC does not protect DNS to many security problems. In fact, it is likely to be affected by IP address spoofing (no client authentication). The confidentiality of data is not preserved and this aspect can play a heavily involved role in DNS reflection DDoS attacks. By signing messages, the size of the messages increases and this gives the attackers a new and highly-potent way to amplify their flooding traffic. Thus, DNS servers need to implement effective and efficient countermeasures. In [16], the authors send search requests to the whole IPv4 address space and collect information related to DNS servers, DNS types classification and DNS software versions. It has been shown in their result that about 30 million DNS servers have been found. As well, they demonstrate that DNSSEC could be so harmful because some types of DNS queries could respond back with very large data packets.

2.3 DNS open resolvers

DNS open resolvers represent a necessary element in DNS amplification attacks and attackers have mostly taken advantage of that for amplifying the attacks. An open DNS resolver server is a DNS server that resolves recursive queries for both local and non-local users. Usually, DNS servers should answer queries coming from their trusted network and reject those that come from other networks. In the case of DNS open resolvers, they answer all the queries that are coming to the server. However, some situations may force some companies to make their DNS server as open DNS resolver in order to serve their employees and clients that are travelling around the world and need trusted DNS servers. It should be noted that most of the known public DNS servers have some security features and very strong DNS policy to avoid abuses. There are also many public DNS open resolvers such as OPENDNS (208.67.222.222 or 208.67.220.220) and Google’s DNS (8.8.8.8). It is worth noting that most of the known

public DNS servers are equipped with a rate limiting system to avoid abusing their system [17]. Notice that according to the Open Resolver Project [18] there are 32 million DNS open resolvers in the world.
In DNS reflection DDoS attacks, DNS open resolvers receive spoofed DNS queries from botnets and return much larger DNS responses to victims (the spoofed IP address). However, statistics [19, 20, 21] show that a huge percentage of DNS servers around the world operate as open DNS recursive servers, unprotected or misconfigured. An attacker can get a high amplification factor when a DNS server answers “ANY" requests. “ANY" request returns all records of entire types recognized to the name server. A firewall can be configured to block all “ANY" requests, but this will probably block legitimate traffic as well. The attackers can easily switch to other DNS queries that cause large amplifications like “RRSIG", “DNSKEY" and “TXT" (more details related to these requests are available in [22]). In Section 3.2 we present the detailed results about this type of DNS servers.

2.4 IP Source Spoofing

In fact, IP spoofing plays a significant role in DNS reflection attacks, and without IP spoofing these types of attacks could never happen. The problem with IP source spoofing in this type of attacks comes from the nature of UDP (User Data-gram Protocol) which DNS uses. In UDP there is no handshaking process and this is the reason that attackers could fool the DNS to respond back to an address different than the original source address of packets.
IP spoofing [23] involves modifying the packet header with a forged (spoofed) IP source address. It involves also the modification of the checksum, and eventually other values like the checksum of TCP header. There are many tools allowing to easily construct crafted packets. Hackers use IP spoofing to illicitly impersonate other machines and launching different attacks. IP spoofing is considered as a serious security problem since a number of notorious attacks, including Smurf attacks, SYN flooding and DNS amplification, are based on this technique. ISP providers can play and an important role to limit IP source spoofing by using ingress filtering [24]. An ingress filtering is a specific technique that used to be sure incoming network packets are absolutely from the networks that they claim to be from. On the other hand, spoofing the source address is very known for Intrusion Prevention Systems (IPS) and can be detected by using guards that block any traffic coming from non-assigned addresses. IP source guards can be also implemented inside local networks by inspecting DHCP traffic and eliminating all packets coming from non-assigned addresses [25]. It can also configure firewalls to stop all incoming traffics that come to our IP space blocks by ingress filtering techniques [24]. Some measurements from MIT Spoofer project [26] show that the ingress filtering has been deployed on 76.2% Autonomous Systems (AS) DNS servers. Other similar statistics coming from ARBOR Networks show that the ingress filtering has been deployed on drawing near 60% ISP networks [27].

2.5 Distributed Reflection Denial of Service

A DRDoS attack happens when reflected responses bombard a victim. It is not easy to distinguish between illegitimate network traffic and legitimate traffics. It is a sophisticated attack that aims to consume either bandwidth or the CPU of the target machine or networks. Once it happens, it is almost unstoppable and no one could stop , the user has to wait until it passes and then evaluate its damage. Hosts used to reflect traffic are called “reflectors". The more reflectors/amplifiers are involved at the same time, the more harmful the attack is. A DRDoS attack can easily exploit UDP based services such as DNS, SNMP (Simple Network Management Protocol) and NTP(Network Time Protocol) to produce good traffic amplification.

2.6 Botnets

In addition to DNS open resolvers and IP source spoofing, another main component, which is more than necessary for attackers is to have access to Botnet networks to run DNS reflection attacks. The attackers who are heavily armed with bot clients use this opportunity to launch bigger attacks. In today’s Internet security threats, most attacks are caused by botnets and their compromised machines. In recent years, there is a number of research works taking place for detecting and mitigating botnets based on DNS traffic [28, 29, 30, 31]. Attackers remotely control the zombies (bot clients). Malicious botnets are networks of compromised machines called “Bots" under the remote control of a human operator called “Botmaster". The term bot comes from the word Robot, in the sense that most of these bots are designed to perform some functions in an automated way [32]. Botnets are collections of bots with malicious software (malware) installed on them that runs autonomously and automatically on a compromised machine without being noticed by the victim users. Those bot clients could be located in schools, homes, companies, and governments around the world [33].

In recent years, the huge amount of research in computer security on botnets has resulted from the fact that botnets are one of the biggest threats to the Internet security. In addition, botnets are one of the main root causes of the large number of powerful threats to Internet security [34]. Botnets are a significant threat to cyber-security because they provide a powerful platform for different cyber-crimes such as DDoS attacks, spams, malware propagation, phishing, click fraud [35, 36, 37]. Another use of these zombie armies for attackers is to anonymize attacker’s identities by using each zombie machine as proxy. The botnets turn into robust weapons for launching distinct cyber attacks that need a large number of bots.

In [38], it has been shown that about 40% of all computers connected to the Internet in the world are infected by botnets and controlled by attackers to do malicious activity. Botnets are classified based on their structure. In case of the impact of botnets, FBI revealed that over 20 million dollar in financial losses had been experience in the USA [39]. The main motivation behind the design and development of botnets by attackers (botmasters) is usually financial

gains. One of the biggest challenges for attackers who owned a botnet is how to turn the internet user's machine to a bot. For instance, in case of Slammer worm [40] the attackers use particular vulnerabilities to infect hosts. There are more botnets which use very modern techniques for joining bots to their botnet, for example, SDBot (aka rBot) uses a number of different mechanisms consisting of P2P networks, open file's shares, back doors that are left by other worms, and exploits of Windows vulnerabilities.

In most of the botnets, there are four main participants, which build a botnet network :

- Developers : They are people or groups that implement and design the botnet. It should be noted that the developer is not necessarily the exact person as the botmaster, and the design and implementation could be subcontracted. Also there are existing malware kits, which provide all the tools for building and managing botnets. They are named Do-it-Yourself (DIY) malware creation. For instance, Zeus DIY and Twitter DIY [41].
- Clients : There are two types of clients for botnets. The first group that rents botnets or botnet services from a botmaster for different purposes such as DDoS attacks or spam distribution. The second group comprises those who aim to be a botmaster for themselves by taking control of botnets, and subsequently they use the botnet network for their own purposes or to conduct illegal activities.
- Victim : An individual, network or system that is target of attack and the attack will be executed to them by the attackers. There are a variety of victims based on the purpose of attacks and botnet.
- Passive Player : This is the owner of the bot clients which got compromised. They have been infected by malware, turn into a bot and part of botnet network. These types of participation, in most situations without machine owner consent, could bring huge legal consequences, as the case of Matthew Bandy, who could have received a 90-year prison sentence for disseminating child pornography [41].

2.7 DNS reflection DDoS Attacks

A DNS reflection DDoS attack is a type of distributed denial of service (DDoS) attack that takes advantage of the fact that a small DNS query can generate a much larger response. When it is joined with IP source spoofing, an attacker can transmit a large volume of DNS traffic to a target system by driving small DNS queries. The amplification factor depends on the type of DNS queries and whether a DNS server supports sending large UDP packets in a response or not. If a DNS server does not support a block size larger than 512 bytes, UDP packets in a response can revert to TCP for that particular request. This reduces the effectiveness of an amplification attack because TCP is much less vulnerable to source address spoofing due to the TCP three way handshake protocol. The relationship between the size

request and its corresponding response is known as the bandwidth amplification factor(BAF) and is computed with the following formula :

$$Amplification\ Factor = response\ size\ /\ request\ size \tag{2.1}$$

The DNS reflection DDoS attack gets its name from how it works. It can turn a few kilobytes of DNS traffic into hundreds of kilobytes and even megabytes. This makes it very effective as a technique for launching distributed denial of service (DDoS) attacks. Usually the attacker, first needs to compromise a set of machines to build his own botnet and then finds and prepares a list of DNS open resolvers, which has a good amplification factor. Attackers could use DNS servers because they could be compromised or simply misconfigured. In case of compromised name servers, an attacker can insert a large “TXT" RR into a zone database hosted on them. Moreover, in most cases, attackers only use “A" records or even design query, which leads to the NXDOMAIN (non existing domain name error) error message and reply code 3.
Finally, the attacker executes the attack by sending spoofed queries from his bots to get a list of DNS open resolvers requesting “TXT" record, “A" record or all record (ANY). Then, for each query sent by the attacker’s botnet, the victim will get one packet, which gives a ratio 1 :1. The key point here is that the responses are much larger than the queries. This is made possible by EDNS0 (Extension mechanisms for DNS) [42], which allows to increase the UDP buffer sizes. Naturally, the name servers that support EDNS0 [42] could participate effectively in this type of attacks. In these cases, a DNS request with a size of approximately 60B can generate a response larger than 4kB. Now consider the scenario where the attacker has a botnet containing thousands/millions of machines while each could simply use its ISP’s name servers to execute the attack.

To sum up, here are some features of the DNS reflection DDoS attacks :

- DNS reflection DDoS attacks must use DNS protocol, port 53 and UDP protocol.
- DNS reflection DDoS attacks cause a large volume of UDP packets in a short period of time.
- Victims receive DNS packets without previously sending out a packet.

In DNS reflection DDoS attacks, the targeted servers get responses without having sent out their corresponding requests. One possible solution to detect this kind of orphan pairs is a one to one mapping [43, 26, 44] for DNS requests and responses. These methods are called DNS guard. Detecting DNS Amplification Attacks (DDAA), usually implement it on both hardware and software equipment. This process is the same as firewalls that monitor all outgoing DNS packets and by filters to detect orphan DNS packets. These detection schemes have no false-positive result but may cause false negatives as mentioned in [45, 46]. By using DNS guards or DDAA, it is also possible to monitor the number of DNS responses, in particular, period of time. Once it reaches a threshold that could be managed by network administrators based

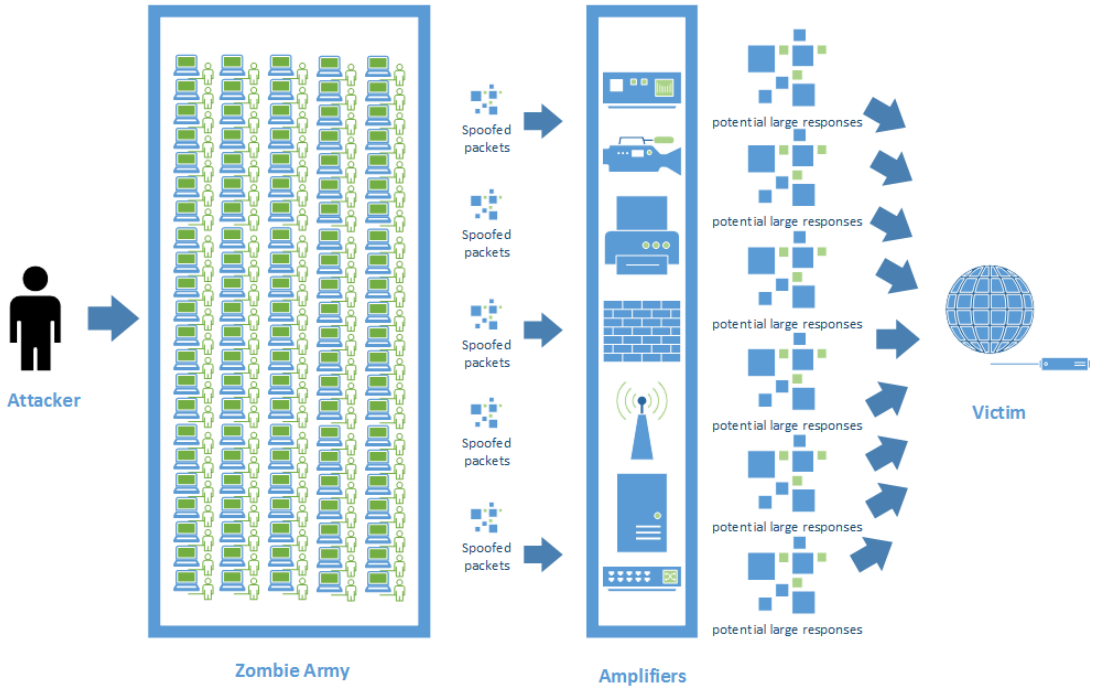

FIGURE 2.1 – A Sample DDoS Attack by Using reflection Methods.

on their organization policies, an alert will be sent dynamically to the system for decision making [47]. One of the most promising methods for slowing down the DNS reflection DDoS attacks is Response Rate Limiting (RRL) which drops responses that exceed a previously configured rate limit, and it is a successful mitigation method for the basic attacks as shown in [48, 49, 50].

In Fig. 2.1, There is a demonstrates of a DNS reflection DDoS attack based on amplification methods. First of all, attackers need to have botnet or network of compromised computers under their control while they can use existing botnet network or build their network of zombies. Attackers then send commands through C&C server which is connected to bot clients. The bot clients based on instruction commands (that could be special set of DNS request which previously designed by attacker) received from attackers and at the same time performing spoofing the source address, send the packets to the amplifiers. As the result, based on this fact that the source address was spoofed, all the potential responses will be send to victims. Any device such as Firewalls, Routers and other devices that act as DNS open resolvers could be considered as amplifiers. In Section 3.2.7, we will explain this specific issue in details.

2.8 Summary

DNS is one of the main, essential and highly used protocols in networks and the Internet. At the same time, DNS is highly attractive to attackers to use this service to achieve their malicious and illegal purposes. DNS reflection DDoS attacks are powerful weapons for attackers and the idea behind these type of attacks comes from the security problems with DNS services, and it works based on this fact that small query could lead to much larger responses. Attackers largely use DNS open resolvers for launching these types of attacks because DNS open resolvers resolves recursive queries for both local and non-local users. The main components for DNS reflection DDoS attacks are a list of DNS open resolvers, and Botnet network which it needed to equip the botnet clients with the malwares to execute attacks to victims while the malware spoofed the IP address. As a consequence all the DNS responses will go to that particular victim. DNS amplification attack could utilize the targeted server or network and could cause serious damage to the victims. In addition, one of our future research is considering research in other UDP based services like NTP. NTP protocol has huge amplification factor, and we have recently seen the biggest attack [4] in the Internet happened based on this protocol, with amplification factor of 600 and 400Gbps traffic. In aforementioned chapter, we have illustrated the main areas of research with high implications for our study. We have also covered major previous research towards this direction.

Chapitre 3

Profiling DNS Open Resolvers

Internet researchers have identified a wide variety of security threats linked to DNS protocol. In this section, we provide an overview of threats with respect to DNS. These threats have been divided into two categories : the attacks that threaten DNS itself, and the attacks that are originated from DNS and DNS infrastructure. For instance, open DNS servers are used to attack other resources. We also present investigation of DNS open resolvers profiling and its upcoming results, such as DNS open resolvers GeoLocation, IP allocations, connection speed, device type, and their purpose.

3.1 DNS Security Threats

Domain Name System (DNS) is a significant element of the Internet infrastructure. Without using DNS servers, we would find it demanding to use the Internet for web browsing or sending email, etc. In recent years, computer and network security researchers identified a vast diversity of security threats linked to DNS. Figure 3.1 summarizes the most common types of threats where DNS is involved. The purpose of this section is to provide information about threat attacks, either threats to the DNS itself, or attacks that take advantage of some elements in the DNS infrastructure in order to launch attacks, and in some cases attack other information technology assets in the infrastructure of the victim.

3.1.1 Threats to the DNS

There is a number of threats to DNS, that are enough powerful to make interruptions in the network operation. Threats to DNS could be classified into three main categories [51] :

- Distributed Denial of Service
- Data Corruption
- Privacy and Information Exposure

In the following subsections, we explain each category in detail.

Distributed Denial of Service

In today's world, distributed denial of service is one of the most significant and powerful threats to the Internet. Unfortunately, this type of attack is very hard to mitigate. As well, with this type of threats, attackers try to flood all available resources which make the DNS server unable to respond. As a result, the server is not able to answer legitimate requests and this makes the services very slow or in some cases unavailable. These problems could happen due to malicious activities or infrastructure failures. DDoS could apply to all the involved elements such as physical, network and server infrastructure. It should be noted that in most of the cases, as soon as the DDOS attacks stop, the system is able to return back to regular operation. Providing multiple servers on separate networks and making servers isolated can help to minimize the effectiveness of this type of attack. The best techniques which are known to network security communities and Internet Service Providers (ISP) for mitigation against this type of attack are BCP38 [52] and BCP84 [53] which are required to be deployed widely. The Network Ingress Filtering (BCP38) is a technique based on restricting traffic that originates from networks, that the source address does not belong to an assigned network range for that particular network. Ultimately that packet should be dropped.

Data Corruption

Data corruption could happen under conditions where changed (or altered) data do not match correct records any more. Attackers try to do unauthorized modification to the sensitive DNS information. Among the possible scenarios for data corruptions we emphasize incorrect data inserted into the cache (aka cache poisoning [54]). Another scenario could happen when an attacker sends an answer to a query before the legitimate DNS server does. Domain Hijacking happens when attackers try to take control of a domain name from the right name holder. These types of problems are more prevalent during the registration process or by falsifying the registrant's account or the transfer authorization. Typosquatting or URL hijacking represent attacks based on a user's mistake while entering accidentally the wrong URL or just a misspelled URL, and thus transfer the user to another IP address which is owned by attackers or malicious groups. Any type of data corruption could significantly affect the ability of domain name service. DNSSEC [12] could help DNS to refuse corrupted response by providing data integrity and data origin authentication. These types of threats do not provide any problem for infrastructures and mostly provide misleading data for users.

Privacy and Information Exposure

DNS servers keep very sensitive information regarding network infrastructure while this information can be extremely useful for attackers to get a better understanding about networks.

Attackers can use it to build the network map and DNS servers represent the best place to provide this sort of information. The fact is that traditional DNS does not use any type of encryption and could be observed at different points in networks. This could be seen as one of the privacy issues in DNS. DNS cache snooping [55] is another privacy issue in DNS, which is a process that determines which 'RR' record is present in DNS cache and gives information regarding queries that DNS resolvers handled in the past.

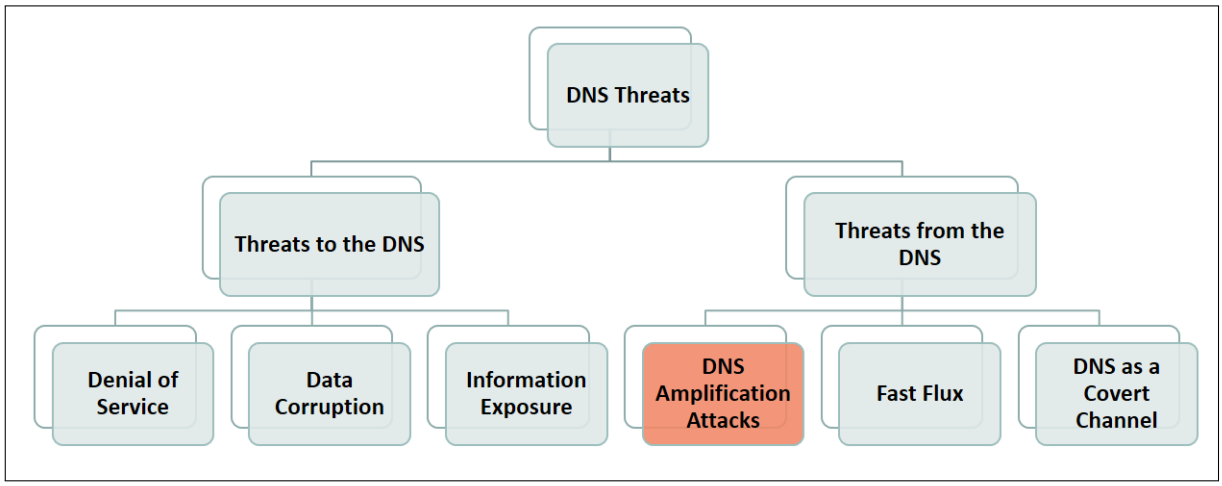

FIGURE 3.1 – DNS Security Threats

3.1.2 Threats from the DNS

There is a number of threats where DNS can play an essential role. Attackers do their best to take advantage of the nature of DNS to achieve their malicious goals. These security threats are :

- DNS Amplification attacks
- Fast Flux DNS
- DNS as covert channel

We will present precise definitions of each part in the following :

DNS Amplification DDoS Attacks

A DNS amplification is a type of distributed denial of service (DDoS) attacks that takes advantage of the fact that a small DNS query can generate a much larger response. When joined with source IP address spoofing, an attacker can transmit a large volume of DNS traffic to a target system by driving small DNS queries. The amplification factor is depends on the type of DNS queries and whether a DNS server supports sending large UDP packets in response (EDNS0) which is a predestinated feature to optimize DNS communications. If a

DNS server does not support large (512 bytes) UDP packets in a response, it can be reverted to TCP. This reduces the effectiveness of an amplification attack because TCP is much less vulnerable to source address spoofing due to the TCP three way handshake. The relationship between the size request and its corresponding response is known as the amplification factor. The DNS amplification attack gets its name from how it works. It can turn a few kilobytes of DNS traffic into hundreds of kilobytes and even megabytes. This makes it a very effective technique for launching distributed denial of service (DDoS) attacks. Usually, the attacker first needs to compromise a set of machines to build his own botnet and then find a list of DNS open resolvers, which have a good amplification factor. These DNS servers can be completely compromised or simply misconfigured. In case of compromised name servers, an attacker can insert a large TXT RR into a zone database.
Moreover, in most of the cases, attackers only use A records or even design a query, which leads to the NXDOMAIN (non existing domain name error) or reply code 3. Finally, the hacker executes the attack by sending spoofed queries from his bot to his list of open resolvers requesting TXT record, A record or all record (ANY). The spoofed queries have as their source address the victims' ones. Then, for each query sent by the attacker, the victim will get one packet, and this gives a ratio 1 :1. The key point here is that the replies are much larger than the queries. This is made possible by RFC2671 [42] (Extension mechanisms for DNS) which allows to increase the UDP buffer sizes. Naturally, the name servers will need to support EDNS0 [42] in order to participate in harmful the attacks. In this case, a request with average size 60 bytes may approach responses up to 4096 bytes. Now consider the scenario where the attacker has a botnet containing thousands of machines. Each of these machines could simply use its ISP's name servers to execute the attack.

Fast Flux DNS

The idea behind the DNS fast-flux is changing the DNS address very quickly. A fast-flux domain returns the small number of records from the large pool of compromised machines. It returns DNS records as soon as low TTL (Time to live is a value in an Internet Protocol packet that present the lifetime of data in network, as well it dictates how long after that value your computer will refresh its DNS information.) is expired, considering the fact that they use their bot client as a proxy. An attacker is able to create a robust one-hop overlay network [56]. Interestingly, DNS fast-flux hosts a massive percentage of online scams.

DNS as Covert Channel

DNS request and reply can be used as a channel to hide the communications. One of the ways to bypass networks firewall is to create a tunnel connection based on existing protocols. Botmasters prefer tunnelling to keep their communications hidden to network security devices. DNS is used as a carrier for other services by transferring inbound and outbound traffic into

request and response queries. These types of tunnels could be established by using DNS. Moreover, attackers can use public DNS servers to achieve their goals without any cost.

3.2 Profiling DNS open resolvers

The main part of our study will be presented in details in this section. We explain our data set, then we extract a list containing DNS servers for our study. Thereafter, we present our data set preparation, and our approach for finding the DNS open resolvers and profiling. The profiling section will be divided into several parts such as, geolocation analysis on which we will demonstrate our profiling based on DNS open resolvers, top continent, countries, cities with the high number of DNS open resolvers with a special analysis for United States and Canada, distribution of DNS open resolvers based on population and their connection speed. Our analysis for DNS open resolvers is based on operating systems, DNS software versions, device type, brands, their purposes and tools which are used for profiling.

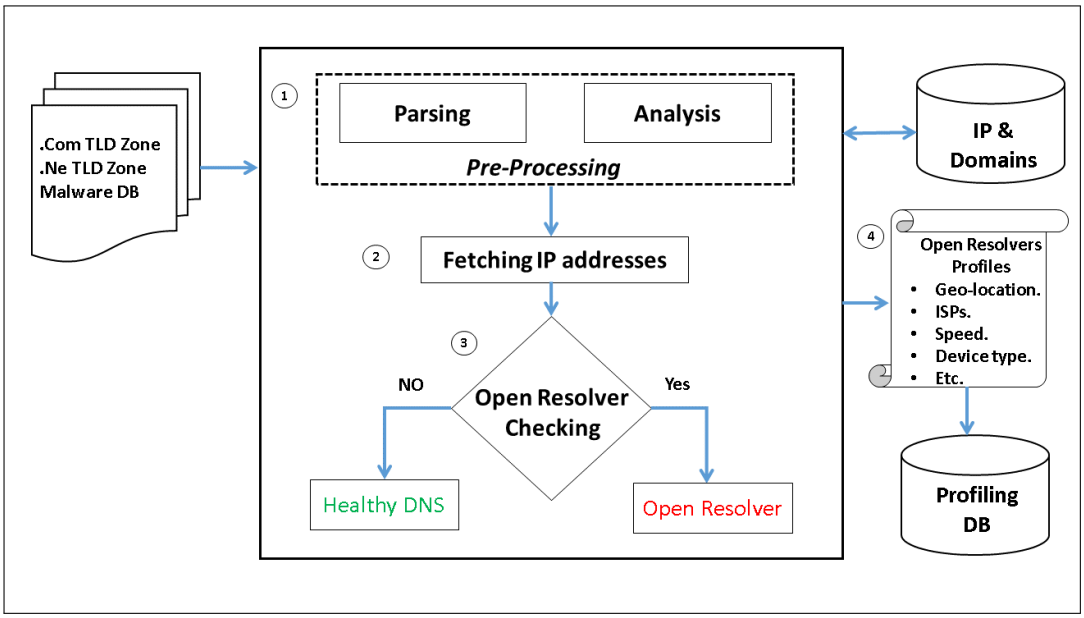

FIGURE 3.2 – Approach for Finding the DNS Open Resolvers and Profiling

3.2.1 Data Set

In this research study, we use two main data sets, TLD zone files, and malware blacklisting database. Our data sets are provided by a third-party. First of all, we give an overview of the information that could be found in TLD zone files. The TLD zone files are updated daily. We can find very useful information about new domains (domains that register every day). Furthermore, useful information is provided by domains registered in the past, both active and inactive ones, name servers associated with each domain, DNSSEC record such

as RRISG (Resource Record Signature), DS (Delegation Signer), DNSSEC keys and NSEC3 (Next Secure 3) [1]. It is important to notice also that the data is stored as flat files. TLD zone files which are maintained by VeriSign are also available in other sources [57] as well. In particular, for Verisign TLD Zone File, Verisign provide FTP access for daily updated zone files. They provide separate files for each TLD such as .Com and .Net TLDs. One of the main causes for selecting these two TLDs, is that the domains hosted in .Com and .Net TLDs are mostly used for general purpose and they provide large portion of domains in the Internet, and also their corresponding name servers. It should be noted that, there are more than 250 million domains which have been registered and interestingly, more than half of them are in .Com and .Net TLDs [58]. we use public malware blacklisting databases.

3.2.2 TLD Zone Files Structure

The TLD files represent flat files that begin with the same structure as DNS zone files. As shown in Fig. 3.3, they have serials, refresh, retry and expiry time. In addition, DNSKEY records related to .Com or .Net TLD can also be seen. In the majority of the files as shown in Fig. 3.4, there are three columns in each TLD zone file : Domain name, type of records, and name servers. One of the steps consists of a preprocessing phase for this study. This is needed in order to extract and prepare valuable information such as name servers (corresponding to domains for our case), and add it to our databases. Consequently, this requires a good understanding of the format of these files. In the first column we have the domain name. It should be noted that based on each zone file, the first column in the domain name is altered and it is not finished by a domain suffix. For instance, as shown in Fig. 3.4, at line 3314, “NORVAL" it means "NORVAL.COM". As well, on the same line, in the third column, the name server corresponds to that particular domain, "NS51.1AND1". Because this information is in .Com TLD zone file, the full name ("NS51.1AND1.COM") is replaced by "NS51.1AND1". The name servers in the TLD zone files are divided into two categories : FQDN (Fully Qualified Domain Name) and relative names (name that did not finish with dot and should be completed by TLDs). It should be highlighted that as shown in Fig. 3.4, there are lines such as 3310, there are FQDN for name servers corresponding to domains. In case of line 3310 in Fig. 3.4, the name server in column three is ended with ".Org", and it means "TOOMANYTHINGS.COM" hosted in that specific name server. In these types of circumstances where the corresponding, name server is not ended with ".COM" they put the full name server name in TLD zone files.

1. RRSIG is a record that contains digital signature of the RR set which is being signed. DS used to create the chain of trust from sign parent zone to a signed chile zone. NSEC3 is a resource record which provides the authenticated denial of existence for DNS RR sets. DNSSEC keys is a record that provides information regarding the public key.

```
; The use of the Data contained in Verisign Inc.'s aggregated
; .com, and .net top-level domain zone files (including the checksum
; files) is subject to the restrictions described in the access Agreement
; with Verisign Inc.

$ORIGIN COM.
$TTL 900
@ IN    SOA     a.gtld-servers.net.               grs.com. (
                                     583723 ;serial
                                    1800 ;refresh every 30 min
                                    900 ;retry every 15 min
                                    604800 ;expire after a week
                                    86400 ;minimum of 15 min
                                    )
$TTL 172800
 NS A.GTLD-SERVERS.NET.
 NS G.GTLD-SERVERS.NET.
 NS H.GTLD-SERVERS.NET.
 NS C.GTLD-SERVERS.NET.
 NS I.GTLD-SERVERS.NET.
 NS B.GTLD-SERVERS.NET.
 NS D.GTLD-SERVERS.NET.
 NS L.GTLD-SERVERS.NET.
 NS F.GTLD-SERVERS.NET.
 NS J.GTLD-SERVERS.NET.
 NS K.GTLD-SERVERS.NET.
 NS E.GTLD-SERVERS.NET.
 NS M.GTLD-SERVERS.NET.
COM. 86400 DNSKEY 256 3 8 AQOxJEkf9BHVH8S3YLvvoEsfwFOHmueyGy7+sw1mC8PB6iziExk
QyKwZXTwtvXJquju0xpHDLJOfW1Eh5eb3UlnHCZvN+UZ3qDBKQ==
COM. 86400 DNSKEY 257 3 8 AQPDzldNmMvZFX4NcNJ0uEnKDg7tmv/F3MyQR0lpBmVcNcsIszx
oiEfGNyvNPaSI7FOIroDsnw/taggzHRX1Z7SOiOiPWPNIwSUyWOZ79VmcQ1GLkC6NlYvG3HwYmynQ
EGU/Qh2K/BgUe8Hs0XVcdPKrtyYnoQHd2ynKPcMMlTEih2/2HDHjRPJ2aywIpKNnv4oPo/
COM. 86400 NSEC3PARAM 1 0 0 -
COM. 86400 RRSIG DNSKEY 8 1 86400 20130821182533 20130814182033 30909 COM. K/
gbn3gU3TuYzzpUVY8Zps5MGMzW85FQR+B9ZHKP2u7ELBI2LgrsrDdzVxC9FE07t3o6moD4T1IyoF2
pMoYzC7VNHZuYzg0AIr7/84EeFTup4JYbxbpwkNhYIXTiwAL8iQ8B9wO67N++1YEPrFGy0PwQjppB
COM. 900 RRSIG SOA 8 1 900 20130822162203 20130815151203 8795 COM. eALJpkz5cS
M9SHfYKe0lOC3/bjZdpVE5GMCPzjqNM2G2SLuPvvG+vZjmQoKmv8X9eUm05qWWXqNBi6H6zk+KIIG
COM. 86400 RRSIG NSEC3PARAM 8 1 86400 20130822041510 20130815030510 8795 COM.
q+80azgx4koRRng+u5BN2T+qdDNkUyO643Sz17EjaV1jRHGp1BdVneZlaFIwjw/x2t0rPPJoq1M0/
COM. RRSIG NS 8 1 172800 20130822041510 20130815030510 8795 COM. CquoPe/uLGmy
3Nk0wCTZkQRYb/eLVdpbSK2/pVju57oSZjLJ9rrNkX3ldUJQsl5e+KAETSRRi965pdC71TJMAC/Al
```

FIGURE 3.3 – TLD Zone Files Structure

```
  3308 WISCONSINRAPIDSTRIBUNE NS NS1.INFI.NET.
  3309 WISCONSINRAPIDSTRIBUNE NS NS2.INFI.NET.
⇨3310 TOOMANYTHINGS NS NS2.MYDYNDNS.ORG.
  3311 TOOMANYTHINGS NS NS1.MYDYNDNS.ORG.
  3312 FLAGWHARFINC NS NS2.CNCHOST
  3313 FLAGWHARFINC NS NS1.CNCHOST
⇨3314 NORVAL NS NS51.1AND1
  3315 NORVAL NS NS52.1AND1
  3316 EARRINGSHOP NS BUY.INTERNETTRAFFIC
  3317 EARRINGSHOP NS SELL.INTERNETTRAFFIC
  3318 TOMMOSER NS NS1.BLUEHOST
  3319 TOMMOSER NS NS2.BLUEHOST
  3320 DONCKERS NS NS49.WORLDNIC
  3321 DONCKERS NS NS50.WORLDNIC
  3322 VISIONLOGISTICS NS NS1.SEDOPARKING
  3323 VISIONLOGISTICS NS NS2.SEDOPARKING
  3324 LOCAL-TRANSPORT NS DNS1.NAME-SERVICES
  3325 LOCAL-TRANSPORT NS DNS2.NAME-SERVICES
  3326 LOCAL-TRANSPORT NS DNS3.NAME-SERVICES
  3327 LOCAL-TRANSPORT NS DNS4.NAME-SERVICES
  3328 LOCAL-TRANSPORT NS DNS5.NAME-SERVICES
  3329 ALLETC NS NS1.ISTI.NET.
  3330 ALLETC NS NS2.ISTI.NET.
  3331 MORENCICANDLES NS NS1.BIGCOMMERCE
  3332 MORENCICANDLES NS NS2.BIGCOMMERCE
  3333 ANANDANDANAND NS NS63.WORLDNIC
  3334 ANANDANDANAND NS NS64.WORLDNIC
```

FIGURE 3.4 – A Portion of TLD Zone Files

```
Whois Server Version 2.0

Domain names in the .com and .net domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.

Aborting search 50 records found .....

   Server Name: NS1.DREAMHOST.COM.AVENIDANET.COM
   IP Address: 66.33.206.206
   Registrar: GODADDY.COM, LLC
   Whois Server: whois.godaddy.com
   Referral URL: http://registrar.godaddy.com

   Server Name: NS1.DREAMHOST.COM.AUTOMAGICIANS.COM
   IP Address: 66.33.206.206
   IP Address: 208.96.10.221
   IP Address: 66.33.216.216
   Registrar: GODADDY.COM, LLC
   Whois Server: whois.godaddy.com
   Referral URL: http://registrar.godaddy.com

   Server Name: NS1.DREAMHOST.COM.AU
   Registrar: MELBOURNE IT, LTD. D/B/A INTERNET NAMES WORLDWIDE
   Whois Server: whois.melbourneit.com
   Referral URL: http://www.melbourneit.com

   Server Name: NS1.DREAMHOST.COM.ATTRACTIONGEEK.COM
   IP Address: 66.33.206.206
```

FIGURE 3.5 – A Sample WHOIS Result

Our analysis setup is shown in table 3.1.

Machine specifications	Operating system	Number of machine
CPU i7-2600 3.4 GHz RAM 12 GB	Debian 7 (Wheezy) PHP and MySQL	5

TABLE 3.1 – Research Project Analysis Setup

3.2.3 Approach to Finding and Profiling DNS open resolvers

In this section, we present our approach for finding and profiling DNS open resolvers worldwide. After analyzing more than 4.9 million unique DNS servers that are available in TLD zone files, we found 330,000 DNS open resolvers all around the world. According to Fig. 3.2, there are four steps to find DNS open resolvers. First, as explained in the data set section, we need to extract valuable information from our data sets. The main aspect in this study is the large number of valid DNS name server. For this reason, we extract DNS server IP addresses or FQDN from our existing databases from TLD zone files. Toward this end, several steps are needed. In case of TLD zone files, we need to extract FQDN and relative names and for those relative names we have to add the corresponding suffix. Also, an important step after extracting the list from TLD zone files is to remove the duplicates since there are a lot of duplicate DNS server names and IP addresses. The duplicated DNS servers mean that at some point one DNS server hosts more than one domain or websites.
According to RFC2182 [59], the number of name servers related to a domain should be at least two or more but preferably no more than seven. Due to this fact, we need to fetch IP addresses associated with our records (domain names). By using the intelligent 'whois' client provided by Network Solutions company, we collect all the IP addresses for each record in our database. Fig. 3.5 shows a sample result for our queries to 'whois' server. As it can be seen, for one domain, we got several IP addresses. We note that in each step we eliminate the duplications to have unique records. In this phase, our goal is also to test our DNS servers records to distinguish DNS open resolvers among the others. It should be pointed out that the test for this part had been taken place with several conditions such as, testing our records in different periods of time, placing different number of queries per time unit (second) and different query types in order to achieve good quality results for the rest of our analysis.
To run this test on our data set, we place a query by using one of DNSUTILS tools called "dig". DNSUTILS is a package that provides various client programs related to DNS. If our query gets an answer, meaning that the DNS server replies backs, then that DNS server is an open DNS resolver. For these tests, we tried to place various queries for some popular domains like Google or Twitter. In order to perform this step with efficiency, we use PHP code to make this process automatic and fast as it can be seen in APPINDEX A. Interestingly, based on our result in this section, we find that there is a large number of name servers which are not DNS open resolvers but even so, they answer some specific queries. However, most of them are

authoritative name servers but, due to some careless misconfiguration they answer particular queries such as 'Upward Referrals' or 'BIND Authors' names list which we will explain in detail later in next chapter.
'Upward Referrals' provides the answer from an authoritative name server which brings back list of root hints, which will be discussed in detail in Section 4.1. 'Bind Authors' is a query that could be placed to any type of name server and could bring back the list of authors for that particular version of BIND, which we will discuss in detail in Section 4.2. In case of Upward Referrals, 13% of the records (which amount to 637,000 name servers out of 4.9 million DNS servers) that we analyzed correspond to answered queries even though the corresponding DNS resolvers were not open. These servers can be part of DNS amplification DDoS attack because they can send back a relatively large answer when compared to its corresponding query size. The size of the answer has an average of 230 bytes, with a related amplification factor close to 4. Moreover, there is a huge list of available name servers that have this problem.
In case of the 'BIND Authors' vulnerability analysis, which is one of our achievements in this research study, we have taken the top one million websites in the world (based on Alexa's [2] website daily top one million ranking list) and discovered that they are served by 180,918 unique DNS servers where 94,908 of them exhibit this vulnerability. This means that 52.45% of these DNS servers answer this query. Also, it should be noted that the average size of responses is 443 bytes (using UDP protocol), whereas the average requests is 38 bytes (using UDP) meaning that the amplification factor is 11.6. This amplification factor is quite large and it could be very dangerous since it can support launching massive attacks. A sample answer resulting from one DNS resolver is provided in Fig. 3.6.

2. http ://www.alexa.com/

```
; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> @[redacted]  google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY. status: NOERROR, id: 11113
;; flags: qr rd ra; QUERY: 1, ANSWER: 11  AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;google.com.                    IN      A

;; ANSWER SECTION:
google.com.             47      IN      A       74.125.239.6
google.com.             47      IN      A       74.125.239.4
google.com.             47      IN      A       74.125.239.2
google.com.             47      IN      A       74.125.239.5
google.com.             47      IN      A       74.125.239.8
google.com.             47      IN      A       74.125.239.14
google.com.             47      IN      A       74.125.239.9
google.com.             47      IN      A       74.125.239.1
google.com.             47      IN      A       74.125.239.3
google.com.             47      IN      A       74.125.239.0
google.com.             47      IN      A       74.125.239.7

;; Query time: 166 msec
;; SERVER: [redacted]#53([redacted])
;; WHEN: Wed Jul 24 08:33:30 2013
;; MSG SIZE  rcvd: 204
```

FIGURE 3.6 – A Sample Result by DNS Open Resolver.

Finally, we proceed to the next level and build our proposed DNS open resolvers profiling. We present a general methodology for building comprehensive profiling of DNS open resolvers in terms of geolocation, their population distribution and connection speed, operating systems, DNS software versions, device type, brands and their purposes. Toward this end, we used several tools and databases such as MaxMind database for geolocation, connection speed, Nmap to OS fingerprinting, device type and brand. We also used Microsoft MapPoint and Google map API (Application Programming Interface) to present our result in visual maps.

3.2.4 Geo-Locating DNS Open Resolvers

This section describes the location of DNS open resolver. The aspect of having knowledge about the location of each DNS open resolvers can help to discover the local DNS open resolvers in specific parts of the world. These DNS open resolvers, which are found in different locations, need to be shut down or fixed. Moreover, for companies and organizations this information could be used to build their policies for receiving DNS related traffic. Another direction is to develop a web-based tool to show the list of DNS open resolvers in a Geographic Information System environment and indicate the distribution of DNS open resolvers based on continent, countries, and cities. In this research study we developed PHP based tools

to bring DNS open resolvers information such as location on the map. This could help to better understand the potential challenges posed by them. Our research will act as a catalyst that can trigger a detailed discussion intended to help the Internet security communities and Internet Service Providers to reduce these types of threats and improve the Internet experience for people around the world. Moreover, it should be mentioned that countries with high bandwidth availability and less secure infrastructure are primary locations of interest for attackers that build and deploy botnets. Furthermore, from our results, we could provide specific information, analysis and measurements on the top originating countries with respect to these types of attacks which could target computers and networks all around the world. In October 2013, with collaboration of Google and Arbor Networks, a real time visualization threat monitoring system was developed. It can show current DDoS attacks around the world and can also provide historical trends in DDoS attacks [60].
In APPINDEX A, we explain the PHP based tools used for presenting our record on the map. In Fig. 3.7, we see the distributions of DNS open resolvers around the world.

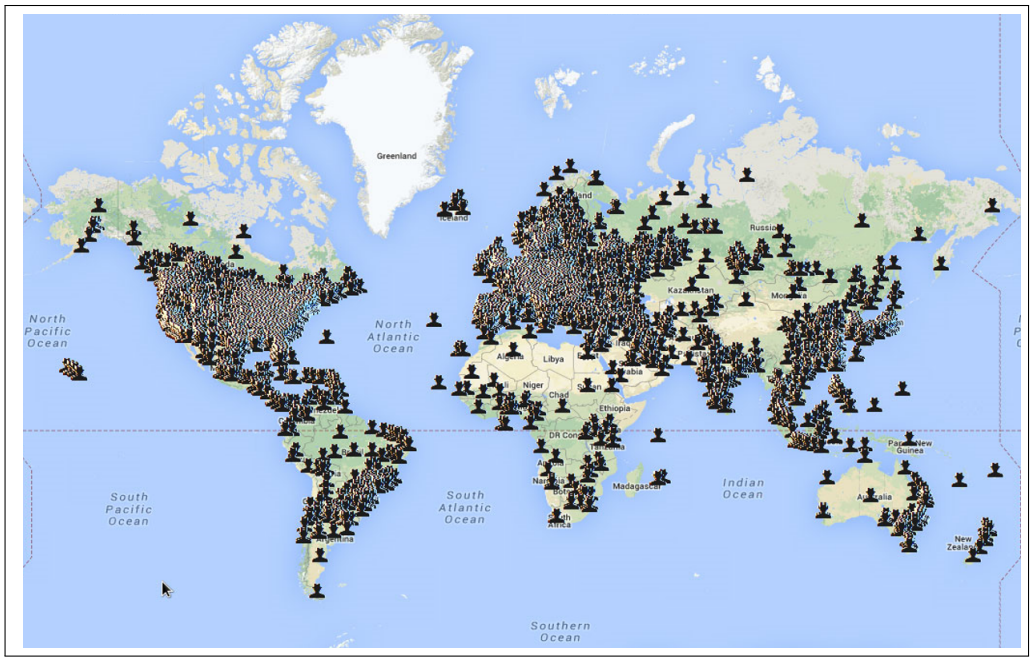

FIGURE 3.7 – Distribution of DNS Open Resolvers in the World.

In Table 3.2, we present the list of top 10 countries with a high number of DNS open resolvers. It should be noted that based on our records, these 10 countries have almost half of the DNS open resolvers that we investigated. In table 3.3, we provide the cities with the most DNS open resolvers around the world. Moreover, we present top cities in each continent in Tables 3.4, 3.5, 3.6, 3.7, 3.8, 3.9.

Countries	Number of DNS open resolvers
United States	77668
Korea	11986
Japan	10029
China	8901
Canada	8258
Russia	7945
United Kingdom	6937
Taiwan	6694
Germany	5981
Turkey	4873

Table 3.2 – Distribution of DNS Open Resolvers Based on Countries

City	Country	Number of Open DNS resolver
Houston	United States	8494
Taipei	Taiwan	3949
Seoul	Korea	3531
Dallas	United States	3185
Tokyo	Japan	2873
Phoenix	United States	2469
Scottsdale	United States	2154
Montreal	Canada	2036
Toronto	Canada	1686
Provo	United States	1562
Atlanta	United States	1553
Chicago	United States	1514
New York	United States	1349
Los Angeles	United States	1345
Beijing	China	1313
Gloucester	United Kingdom	1269
Central District	Hong Kong	1259
Moscow	Russian Federation	1106

Table 3.3 – Top Cities with a High Number of DNS Open Resolvers

City	State	County	Number of Open DNS resolvers
Houston	Texas	United States	8494
Dallas	Texas	United States	3185
Phoenix	Arizona	United States	2469
Scottsdale	Arizona	United States	2154
Montreal	Quebec	Canada	2036
Toronto	Ontario	Canada	1686
Provo	Utah	United States	1562
Atlanta	Georgia	United States	1553
Chicago	Illinois	United States	1514
New York City	New York	United States	1349
Los Angeles	California	United States	1345

TABLE 3.4 – Cities in North America with a High Number of DNS Open Resolvers

City	Country	Number of DNS Open resolvers
Gloucester	United Kingdom	1268
Moscow	Russia	1100
Sanayi	Turkey	861
Istanbul	Turkey	433
London	United Kingdom	427
Saint Petersburg	Russia	338
Berlin	Germany	291
Madrid	Spain	267
Host	Germany	260
Amsterdam	Netherlands	243

TABLE 3.5 – Cities in Europe with the High Number of Open DNS Resolvers

City	Country	Number of DNS Open Resolvers
Taipei	Taiwan	3949
Seoul	South Korea	3531
Tokyo	Japan	2873
Beijing	China	1313
Central District	North Korea	1259
Bangkok	Thailand	848
Guangzhou	China	691
Osaka	Japan	634
Shanghai	China	597
Taichung	Taiwan	541

Table 3.6 – Cities in Asia with the High Number of DNS Open Resolvers

City	Country	Number of Open DNS Resolvers
Bogota	Colombia	878
Buenos Aires	Argentina	540
Santiago	Chile	273
Sao Paulo	Brazil	166
Caracas	Venezuela	62
Montevideo	Uruguay	50
Medellín	Colombia	50
Lima	Peru	40
Quito	Ecuador	37
La Paz	Bolivia	35

Table 3.7 – Cities in South America with the High Number of DNS Open Resolvers

City	Country	Number of Open DNS Resolvers
Sydney	Australia	233
Melbourne	Australia	147
Brisbane	Australia	146
Perth	Australia	119
Auckland	New Zealand	91
Canberra	Australia	56
Adelaide	Australia	55
Surry Hills	Australia	42
Mulgrave	Australia	24
Ivanhoe	Australia	17

TABLE 3.8 – Cities in Oceania with the High Number of DNS Open resolvers

cities	Country	Number of DNS Open resolvers
Cairo	Egypt	50
Johannesburg	South Africa	39
Cape Town	South Africa	31
Lagos	Nigeria	30
Pretoria	South Africa	17
Parow	South Africa	15
Casablanca	Morocco	15
Windhoek	Namibia	12
Cheraga	Algeria	9
Nairobi	Kenya	9

TABLE 3.9 – Cities in Africa with the High Number of DNS Open Resolvers

In this section, we provide insights about Canadian DNS open resolvers. In Table 3.10, we list top cities in Canada based on the high number of DNS open resolvers. Also in Fig. 3.8 we provide the DNS open resolvers in Canadian provinces with the highest rate of DNS open resolvers.

City	Province	Country	Number of DNS Open Resolvers
Montreal	Quebec	Canada	2036
Toronto	Ontario	Canada	1686
Calgary	Alberta	Canada	704
Vancouver	British Columbia	Canada	568
Burnaby	British Columbia	Canada	524
Kelowna	British Columbia	Canada	516
Laval	Quebec	Canada	422
Ottawa	Ontario	Canada	244
Hamilton	Ontario	Canada	215
Mississauga	Ontario	Canada	159

TABLE 3.10 – Top Cities in Canada Based on High Number of DNS Open Resolvers

FIGURE 3.8 – Distribution of DNS Open Resolvers in Canadian Provinces

3.2.5 IP Allocation for DNS Open Resolvers

Another essential point is to know about the type of person, company or organization behind the DNS open resolvers. To do so, for our Canadian DNS open resolvers we performed an analysis in order to get a better understanding. Also, we looked into the type of IP allocation

for each of the analyzed DNS open resolvers. As a result, we found out that the major problem regarding the DNS open resolvers are related to “Direct Allocation" category, which provides the most potent element for attackers in their aim to amplify their attacks. Our statistics show that, in the case of Canadian open DNS servers, the most encountered type of allocation is “Direct Allocation". This means that ARIN (American Registry for the Internet) has given those IP address spaces directly to the companies and organizations. For instance, an ISP has a large IP address space assigned to their customers, and each ISP client is using a portion of that 'Direct Allocation' which the ISP received from ARIN. Sometimes, a large ISP and companies allocated some small IP address space to other companies or organizations having the type of IP allocation as “Reassignment". There is another type of IP space allocation such as “Direct Assignment". In these cases ARIN directly assigns the IP address space to an organization for its own exclusive use. As well, “Reallocation" is performed when an organization (the upstream) assigns part of its IP address space to a downstream customer. In Table 3.11, it could be seen that 72% of Canadian DNS open resolvers are “Direct Allocations".

Type of IP Allocation	Percentage
Reallocated	1%
Direct Assignment	6%
Reassigned	21%
Direct Allocation	72%

Table 3.11 – Canadian DNS Open Resolvers Type of IP Space Allocation

3.2.6 Connection Speed of DNS Open Resolvers

We emphasize that regarding our main goal (which is to provide rich insights about DNS open resolvers), one of the significant pieces of information that we need to know about each DNS open resolver is who is behind them ? Is that a company/corporate or individual(their ownerships). We could classify the stakeholders into two groups : corporate (company) and individual. Our results show that based on the connection speed, approximately two third of our DNS open resolvers are individual. By using PHP based application and Maxmind databases, we show DNS open resolvers in visual maps. Information regarding this application is provided in APPINDEX A. In Fig. 3.9, we show these two groups based on our records.

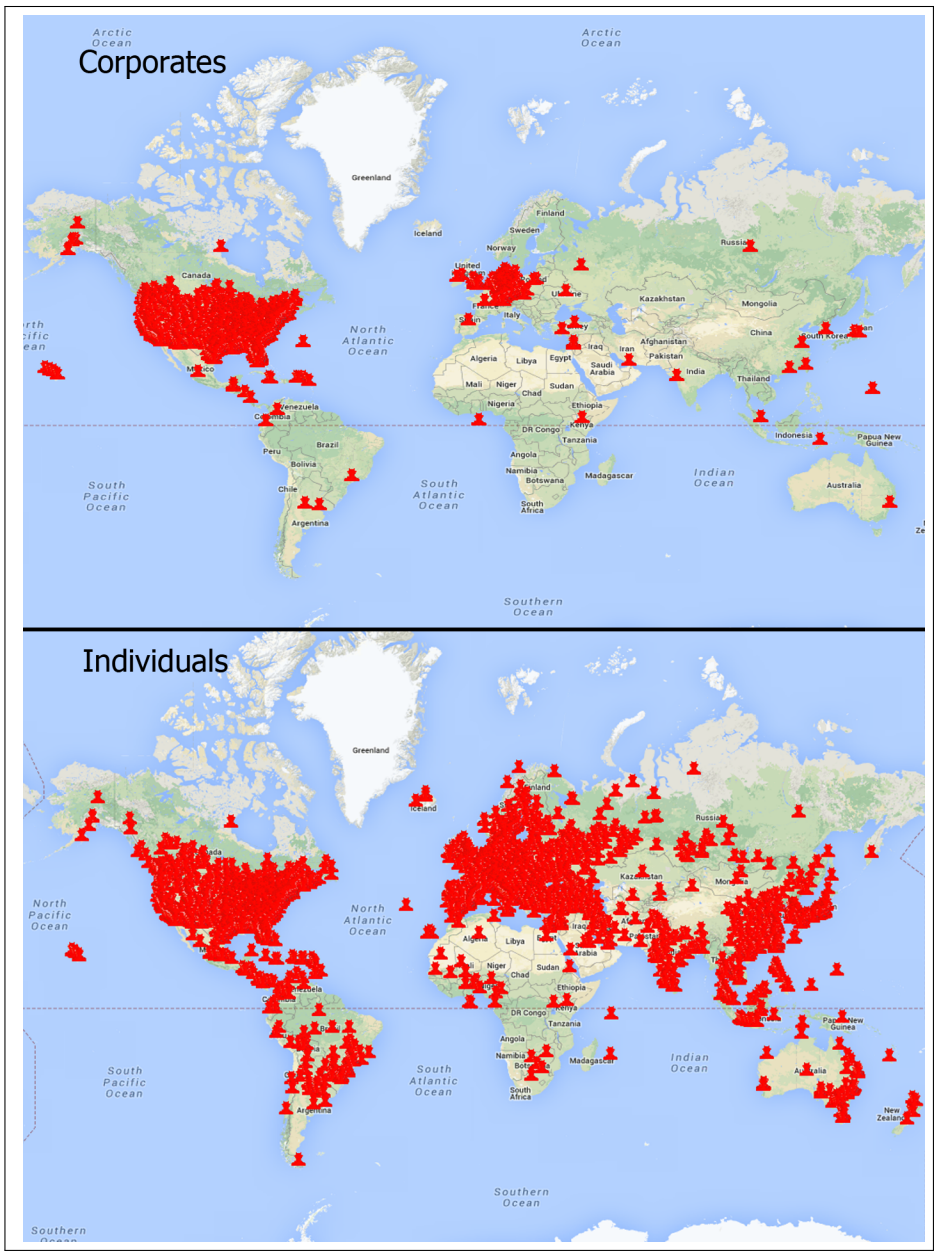

FIGURE 3.9 – Distribution of DNS Open Resolvers Based on Connection Speed.

3.2.7 Device Type and DNS Open Resolvers

During the analysis, we found surprising results regarding the amplifiers in DNS reflection DDoS attacks. We faced different categories of hosting equipments acting as DNS open resolvers. The first category involves DNS open resolvers running on a regular purpose. This category contains the majority of the DNS open resolvers (91% of all open DNS resolvers of our DB), and most of them are running on Windows or Linux operating system. On the other hand, another category exists which involves devices where domain naming service is not the

main purpose.

Moreover, most of the equipments in this latter category should not even act as DNS open resolvers. Around 9% of DNS open resolvers that we investigated have this characteristic. This may have happened because of the misconfiguration, or in some specific cases, the nature of the device. There is a possibility that the design team made some mistakes during coding and embedding the operating system of these devices. Interestingly, in our analysis we found more than 67 different brands and a large portion of these brands are well-known. It is remarkable that we found more than 150 different device models, and unfortunately because of sensitivity of this information, we do not provide the full details.

We divided this group of DNS open resolvers into 11 categories, including specialized [3], firewalls, WAPs, Storage-MISCs, VOIPs, broadband routers, Load balancers, switches, printers, phones and media devices. In Table 3.2.7, we show these categories of DNS open resolvers. It should be noted that this list is sorted based on the maximum number of DNS open resolvers in each category. To produce these results, we used Nmap Security Scanner 6.4, which was released in August 2013 and contains an important improvement related to OS detection algorithms and devices fingerprinting [61]. Concerning the analysis of different platforms for DNS open resolvers, Microsoft Windows has the top position and it was used by 46.62% of DNS open resolvers. In the second position, was Linux with 36.62% and BSD in the third position with 14.31%. The detailed information is presented in Table 3.2.7.

3. Any open DNS resolver, which does not fall into one of these groups ; we named them as specialized.

Device Type	Percentage
Specialized	30.37%
Firewall	27.82%
Wap	15.20%
Storage-misc	8.36%
Voip	3.65%
Broadband	3.54%
Router	2.58%
Load balancer	2.17%
Switch	1.79%
Printer	1.57%
Phone	1.51%
Media device	1.43%

TABLE 3.12 – Repartition of Devices

Operating system	Percentage
Microsoft Windows	46.62%
Linux	36.62%
BSD	14.31%
Apple Mac OS	1.23%
VMware	0.67%
Sun Solaris	0.47%
Novell	0.10%

TABLE 3.13 – Repartition of DNS Open Resolver's Operating System

We also provide detailed information for the repartition of DNS open resolvers using Microsoft Windows and Linux/Unix in Fig. 3.10 and Fig. 3.11.

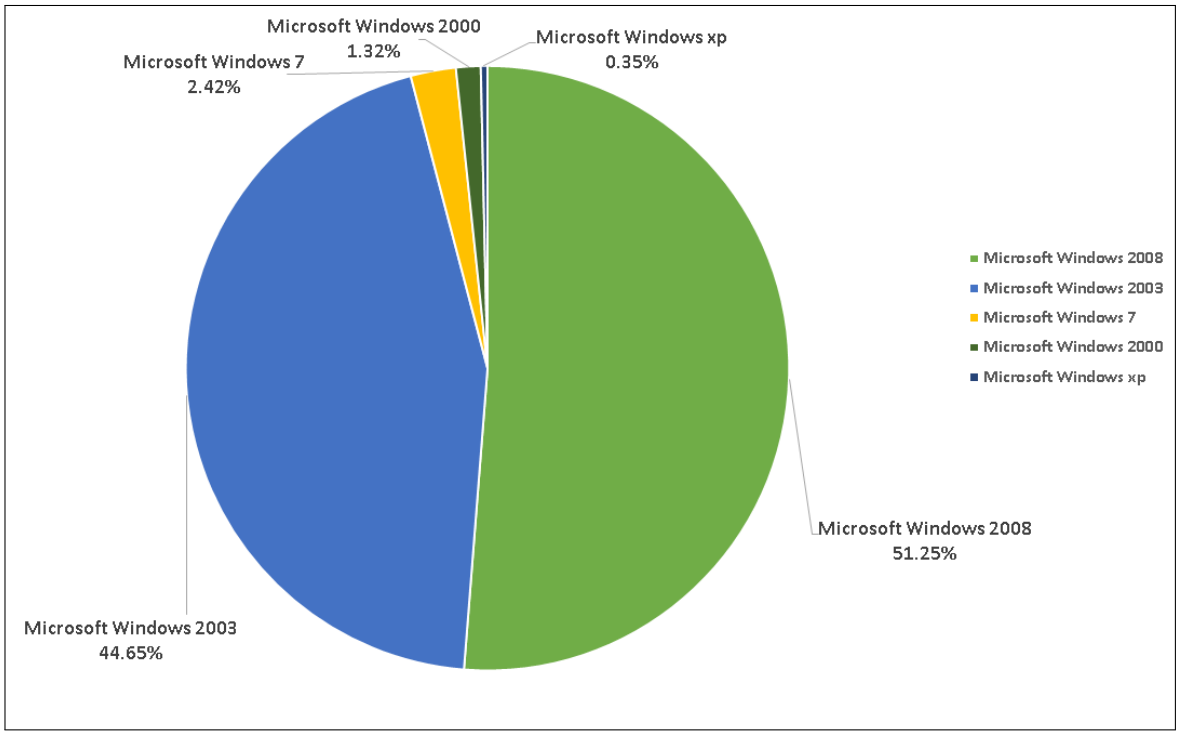

FIGURE 3.10 – Repartition of DNS Open Resolvers Based on Microsoft Windows.

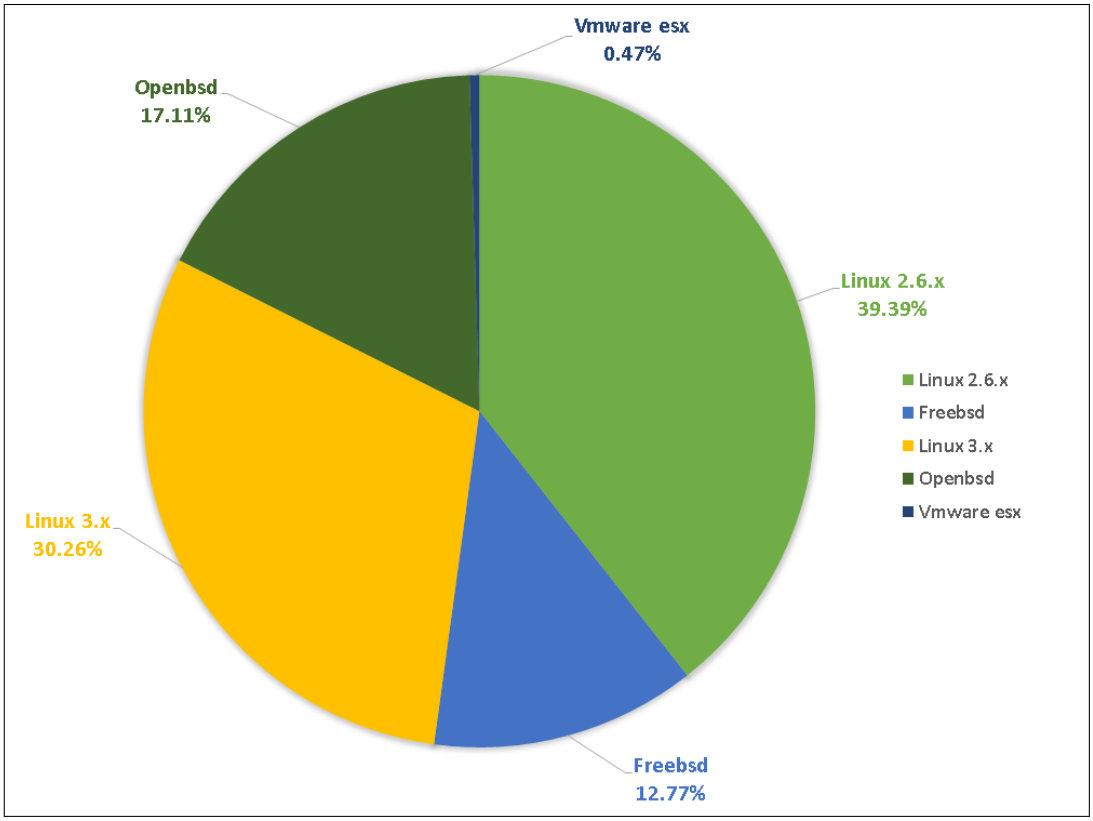

FIGURE 3.11 – Repartition of DNS Open Resolvers Based on Linux/Unix.

During our experiments, we identified different research directions that could be helpful and be considered as an extension of this research study. For instance, based on our data set which is quite large (4.9 million DNS name server), we analyzed Top Level Domains that host most of the DNS open resolvers. The result (shown in Fig. 3.12) was predictable to some extent as the first and second rank for top TLDs (which host the most DNS open resolvers) are ".COM" and ".NET". Interestingly, the third one is the domains ending in ".RU", the fourth is represented by the domains ending with ".ORG" while the fifth is represented by the domains ending with ".BR". This information could be used for future work since it is valuable for situations where we need to build intelligent systems for detecting malicious traffic. In this context we need reasonable factors to assign weight or cost. Based on such factors and the incoming traffic, one can profile malicious communication. Moreover, we analyzed the length of domains as another factor. We discovered that more than 91,000 DNS open resolvers have domain names with lengths from 11 to 17 characters (details are given in Fig. 3.13).

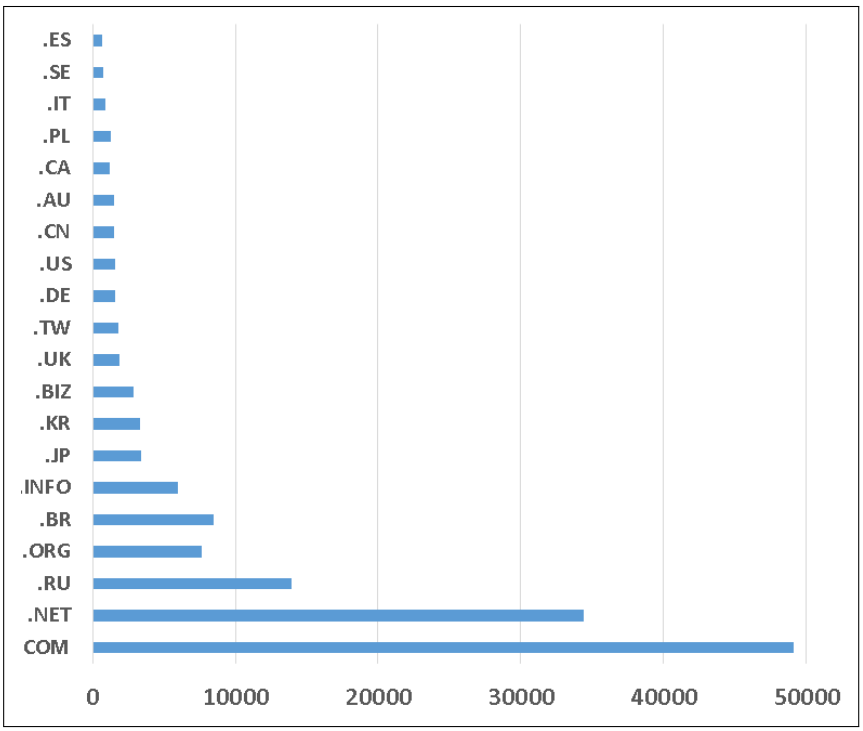

FIGURE 3.12 – Top Level Domains Host the Most DNS Open Resolvers.

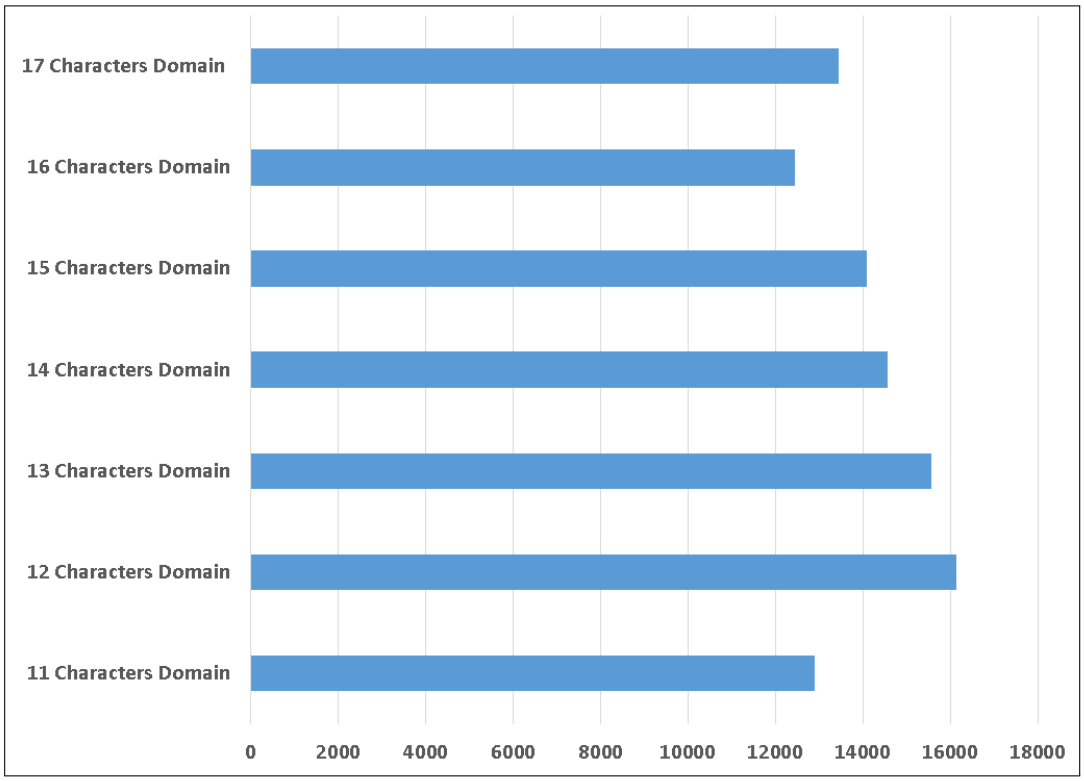

FIGURE 3.13 – DNS Open Resolver Based on Domains Length.

3.3 Summary

In this chapter, we addressed a variety of security threats linked to the DNS protocol. We also addressed the topic of DNS open resolvers profiling. More precisely, we explained the related structure of our data set, the approach to finding and profiling DNS open resolvers and then we elaborated on our results for different profiling aspects. These include DNS open resolvers GeoLocation, IP Allocations, connection speed, device type, and purpose.

Chapitre 4

Security Assessment and Abuse Analysis of DNS Open Resolvers

In this chapter, we look into the security assessment and analyze the collaboration of our DNS open resolvers with malware from malware blacklist databases. This offers a deep awareness about the security problems, which might be faced and an appropriate security assessment of DNS open resolvers. After analyzing DNS open resolvers, we found that there exist serious problems with DNS vulnerabilities, secure configuration and misconfiguration of DNS software. Also, we encountered a number of DNS open resolvers that are malicious because they are associated with different malware families. We discuss the corresponding information about our DNS open resolvers security assessment, abuse analysis, collaboration of DNS resolvers with malware and recommendations for secure configuration of 'BIND' name server, which is the most used name server in the Internet.

4.1 Root Zone Attacks

Name server functionality can be divided into two main categories, authoritative service and recursive service, both of which are detailed in Section 2.1.2. An authoritative name server is the one that is configured to answer queries for a specified set of zones and satisfies queries from its own data without the need of references from another source.
Unfortunately, if an authoritative name server is not configured correctly, it can be part of DNS reflection DDoS attacks. In fact, authoritative name servers should REFUSE those DNS queries that are not related to their hosted zone, and they should respond to these types of queries by a short message containing a REFUSING code. However, we found a huge number of authoritative name servers which are not configured correctly. Furthermore, while they do not answer queries unrelated to their hosted zones, they send however responses containing Upward Referrals (list of thirteen root DNS servers). Upward referrals represent a relatively big answer, and it is useless since iterative resolvers already know the information.

We analyzed 4.9 million domain name services and we found that 13% (637,000) of them have this problem while not being open resolvers.
These servers can be part of DNS reflection DDoS attacks, because they can send back a relatively big answer when compared to its corresponding query size. The size of the answer is not that big (an average of 230 bytes and amplification factor close to 4) but there is a huge list of available name servers with this problem. However, the following information, when added to the BIND configuration file, forbids them to send upward referrals.

Options {additional-from-cache no ;} ; (4.1)

In Fig. 4.1, we demonstrate a sample result for Upward Referrals.

```
; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> @ns.[redacted] yahoo.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 22547
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL: 13
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;yahoo.com.                     IN      A

;; AUTHORITY SECTION:
.                       342981  IN      NS      m.root-servers.net.
.                       342981  IN      NS      a.root-servers.net.
.                       342981  IN      NS      b.root-servers.net.
.                       342981  IN      NS      c.root-servers.net.
.                       342981  IN      NS      d.root-servers.net.
.                       342981  IN      NS      e.root-servers.net.
.                       342981  IN      NS      f.root-servers.net.
.                       342981  IN      NS      g.root-servers.net.
.                       342981  IN      NS      h.root-servers.net.
.                       342981  IN      NS      i.root-servers.net.
.                       342981  IN      NS      j.root-servers.net.
.                       342981  IN      NS      k.root-servers.net.
.                       342981  IN      NS      l.root-servers.net.

;; ADDITIONAL SECTION:
a.root-servers.net.     429381  IN      A       198.41.0.4
a.root-servers.net.     429381  IN      AAAA    2001:503:ba3e::2:30
b.root-servers.net.     429381  IN      A       192.228.79.201
c.root-servers.net.     429381  IN      A       192.33.4.12
d.root-servers.net.     429381  IN      A       199.7.91.13
d.root-servers.net.     429381  IN      AAAA    2001:500:2d::d
e.root-servers.net.     429381  IN      A       192.203.230.10
f.root-servers.net.     429381  IN      A       192.5.5.241
f.root-servers.net.     429381  IN      AAAA    2001:500:2f::f
g.root-servers.net.     429381  IN      A       192.112.36.4
h.root-servers.net.     429381  IN      A       128.63.2.53
h.root-servers.net.     429381  IN      AAAA    2001:500:1::803f:235
i.root-servers.net.     429381  IN      A       192.36.148.17

;; Query time: 33 msec
;; SERVER: [redacted]
;; WHEN: Fri Mar 14 12:49:26 2014
;; MSG SIZE  rcvd: 494
```

FIGURE 4.1 – A sample result for Upward Referrals

4.2 BIND Authors : A Replacement for DNS Open Resolvers

One of our results in this research study is that we found an interesting feature in BIND name servers that can be viewed as vulnerability. If this feature is available, attackers do not need to look for the list of DNS open resolvers to amplify their traffic in DNS amplification

DDoS based attacks, and this security issue could completely replace the need for DNS open resolvers. This feature allows attackers to use regular queries, even to authoritative BIND name servers in order to ask for the list of BIND authors. The request query is small, but the response is big enough to make DNS amplification attacks. The most important aspect of this vulnerability is the wide range of availability of this powerful feature. What makes the situation worst is that this feature is enabled by default in BIND name servers. This means that a huge number of websites have their corresponding DNS server with this problem (the type of DNS server does not matter, either authoritative name server or resolvers).
We analyzed the top one million websites in the world (based on Alexa's website daily top one million ranking list) and discovered that they are served by 180,918 unique DNS servers, where 94,908 of them have this vulnerability. It means that 52.45% of these servers answer this query. Also, it should be noted that the average size of responses is 443 bytes (using UDP), whereas the average size of a request is 38 bytes (using UDP) meaning that the amplification factor is 11.6. This amplification factor is quite big, and it could be very dangerous for launching massive attacks. Also the average size of responses related to this feature is very close to 512 bytes, which is close to the DNS protocol block-size limit. Fig. 4.2 shows a screen shot of this BIND problem.

```
root@debian-srv02:~/bind# dig CH @[redacted] authors.bind TXT
;; Truncated, retrying in TCP mode.

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> CH @NS1.[redacted].COM authors.bind TXT
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43450
;; flags: qr aa rd; QUERY: 1, ANSWER: 19, AUTHORITY: 1, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;authors.bind.                  CH      TXT

;; ANSWER SECTION:
authors.bind.           0       CH      TXT     "David Lawrence"
authors.bind.           0       CH      TXT     "Curtis Blackburn"
authors.bind.           0       CH      TXT     "Danny Mayer"
authors.bind.           0       CH      TXT     "Brian Wellington"
authors.bind.           0       CH      TXT     "Scott Mann"
authors.bind.           0       CH      TXT     "Bob Halley"
authors.bind.           0       CH      TXT     "Jeremy C. Reed"
authors.bind.           0       CH      TXT     "James Brister"
authors.bind.           0       CH      TXT     "JINMEI Tatuya"
authors.bind.           0       CH      TXT     "Francis Dupont"
authors.bind.           0       CH      TXT     "Matt Nelson"
authors.bind.           0       CH      TXT     "Ben Cottrell"
authors.bind.           0       CH      TXT     "Michael Graff"
authors.bind.           0       CH      TXT     "Mark Andrews"
authors.bind.           0       CH      TXT     "Evan Hunt"
authors.bind.           0       CH      TXT     "Damien Neil"
authors.bind.           0       CH      TXT     "Michael Sawyer"
authors.bind.           0       CH      TXT     "Andreas Gustafsson"
authors.bind.           0       CH      TXT     "John H. DuBois III"

;; AUTHORITY SECTION:
authors.bind.           0       CH      NS      authors.bind.

;; Query time: 85 msec
;; SERVER: [redacted]53([redacted])
;; WHEN: Mon Apr  7 13:45:56 2014
;; MSG SIZE  rcvd: 540

root@debian-srv02:~/bind#
```

FIGURE 4.2 – BIND authors sample screen shot.

DNS amplification DDoS attacks which use DNS open resolvers could be compared to NTP based DDoS attacks which use[1] feature of time synchronization service. Attacks via BIND author's vulnerability are most likely to happen when the attacker may have a huge number of amplifiers (DNS server's corresponding for websites), which are readily available and responsive. In this case, an attacker does not even need to have access to DNS open resolvers since any DNS type itself could be used to participate in these attacks. In the conditions where the

1. Monlist is a monitoring feature which allows administrators to query some statuses regarding online clients

attackers are not equipped with a large list of acceptable amplifiers, they send their requests to the bunch of IP space with this hope that, some of them will respond back to targeted victims. These types of attacks could be easily seen in Darknet traffic [62]. Our statistics show that about 52.45% of name servers exhibit the 'BIND Authors' exploit. Attackers do not even need to look for DNS servers with this problem. Instead, they can blindly select a bunch of name servers and order their bots to send a BIND author requests to them. On average, half of BIND name servers will answer the query and the attack will be effective. It is concerning that a large number of enterprise websites have this problem, and they can be part of DNS reflection DDoS attacks. Because of the high end infrastructure of many companies attackers can take advantage of this infrastructure and launch their attacks effectively even with a few numbers of DNS servers. Since the authors' information is stored in 'TXT RR' inside the BIND servers, this problem can be easily avoided by disabling the version feature. In the following, we show how the malicious query could take place ; the query type must be 'TXT' resource record, and the query class must be Chaos ('CH'), and placed for instance, with 'dig' software which is commonly available in Linux distributions :

$$\#dig\ CH\ @nameserver.com\ authors.bind\ TXT. \tag{4.2}$$

4.3 DNS Server Software Version Distribution and Vulnerabilities

Another important information is the relation of DNS open resolvers among, the DNS Server Software Version and the corresponding vulnerabilities. Fig. 4.3 shows the number of vulnerabilities for different versions of the BIND DNS servers. With a number of 15 vulnerabilities, BIND 9.3 takes the top position [63] as the most exposed. Our results indicate more than 62,000 name servers having their DNS Server Software Version as BIND 9.x. Table 4.1, shows a list of the top BIND versions having this problem together with the number of name servers found by our analysis regarding this BIND versions. The version 9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1 is in the top and there are 33,406 DNS name servers in our records with this DNS Server Software Version.

Bind version name	Number of DNS servers
9.8.2rc1-RedHat-9.8.2-0.23.rc1.el6_5.1	33406
9.3.6-P1-RedHat-9.3.6-20.P1.el5_8.6	23708
9.8.2rc1-RedHat-9.8.2-0.17.rc1.el6_4.6	4109
9.7.3	3187
9.8.2rc1-RedHat-9.8.2-0.17.rc1.el6_4.4	2484
9.8.1-P1	1914
9.8.4-rpz2+rl005.12-P1	1428
9.2.4	1338
9.3.4-P1	1146
9.8.2rc1-RedHat-9.8.2-0.10.rc1.el6_3.6	1143
9.8.2rc1-RedHat-9.8.2-0.17.rc1.el6_4.5	919
9.3.6-P1-RedHat-9.3.6-20.P1.el5	908
9.3.6-P1-RedHat-9.3.6-16.P1.el5	877
9.3.6-P1-RedHat-9.3.6-4.P1.el5_4.2	703
9.3.6-P1-RedHat-9.3.6-20.P1.el5_8.5	637
9.3.6-P1-RedHat-9.3.6-16.P1.el5_7.1	633
9.7.0-P1	565
9.9.3-P2	563
9.8.2rc1-RedHat-9.8.2-0.17.rc1.el6.3	538
9.9.4-P1	505
9.6-ESV-R4	503

TABLE 4.1 – BIND Versions, which are Open to Author's Problem with more than 500 Servers.

To find information related to the version, we send a query to the servers to look for a feature in BIND that returns the DNS server software version. The query asks for "version.bind" string. The query resource record type must be 'TXT', and the query class must be Chaos ('CH'). The query is as follows :

$$\#dig\ CH\ @nameserver.com\ version.bind\ TXT. \tag{4.3}$$

One of the most important name server features which should be disabled in order to prevent attackers from discovering weaknesses is the DNS server software version. Here is an example for disabling BIND version in the BIND configuration file (/etc/named.conf) :

Options {Version none ;} ; (4.4)

It should be noted that disabling BIND version, could help to prevent the DNS name server from providing authors name list.

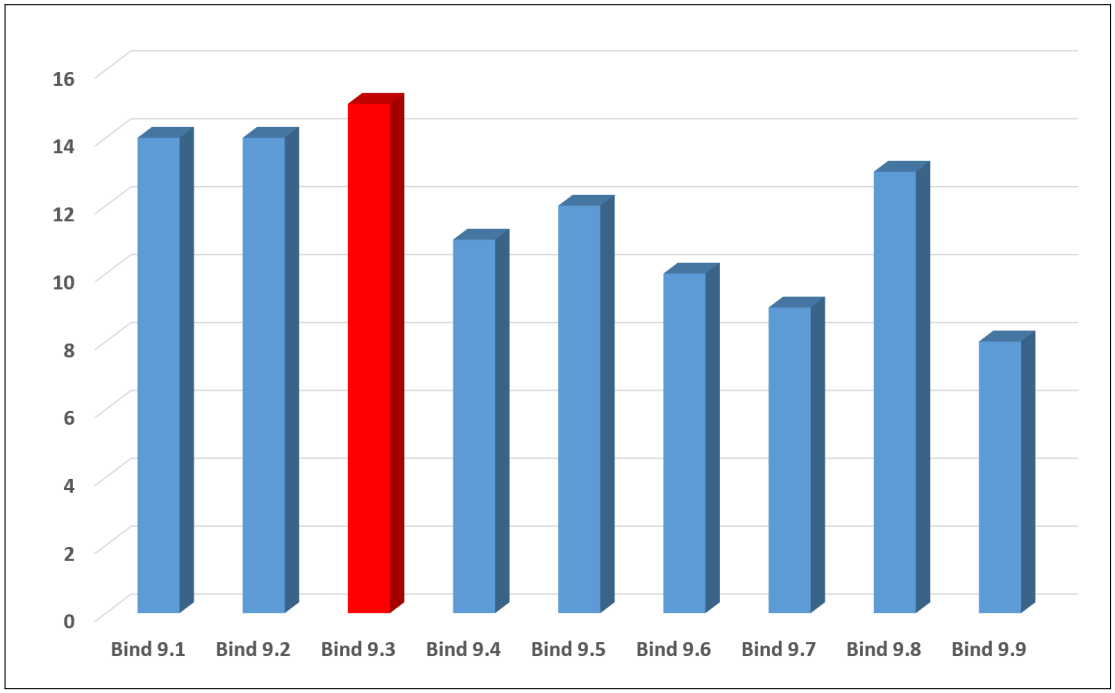

FIGURE 4.3 – Number of vulnerabilities for BIND Name Servers.

4.4 Association of DNS Open Resolvers and Malware

In this section, using the malware databases, we provide insights regarding the investigated DNS open resolvers which have been associated with malware. This analysis shows that these DNS open resolvers are exhibiting malicious behaviour and they are contributing to malicious activities. The phrase malware is a brevity for malicious software and it refers to software or malicious code. The malware was created for disruptive activities and that could act maliciously without the awareness of the user or the administrator. There are several free online services that provide malware blacklist databases which can be used to analyze files, websites and IP addresses and enable recognition of malicious codes, viruses, trojans, worms and plenty of other malicious content detected by several anti-virus engines. These databases provide unique hash values for each malware or piece of malicious code. The main mission for these databases is to help network/cyber security community and industry to make the Internet a much safer place. In this study, we aim at finding malicious activities related to our list of DNS open resolvers that have been associated with the malware. For that reason, we jointly analyze DNS open resolvers along with malware database records with respect to time

which is 4 weeks that we identify the DNS open resolvers. We analyze the significant amount of traffic which is generated by the malware that has been associated with the DNS open resolvers under investigation. The general statistics related to DNS open resolvers associated with malwares show that, in those particulate 4 weeks (July 2013 - August 2013), 104 DNS open resolvers have been involved in malicious activities. The number of malicious requests for those 104 DNS open resolvers is 28453. They connected with 2,495 malwares and employed 6 different protocols such as HTTP (Hypertext Transfer Protocol), DNS, IRC (Internet Relay Chat), FTP (File Transfer Protocol), as per the statistics shown in Table 4.2.

General Statistics about DNS open resolvers associated with Malware	Numbers
Number of involved open DNS resolvers	104
Number of Request associated with Malware	28453
Number of Unique Malware Family	2495
Number of protocols used associated with Malware	6

TABLE 4.2 – General Statistics of DNS Open Resolvers Associated with Malware.

Furthermore, we analyzed the behaviour of DNS open resolvers associated to malware based on their reputation, which means the number of malicious requests that have been associated with our DNS open resolvers. We also considered the abuse level which provides information regarding DNS open resolvers and the number of the involved malware families during the malicious activities. Moreover, we performed protocol analysis. For instance, while HTTP is the most used protocol in the analyzed traffic, IRC is the most abused one, which means that the percentage of malicious traffic over the IRC protocol is around 50%. This means that 1 out of 2 requests based on the IRC protocol is malicious. In Fig. 4.4, we represent the top 10 open DNS resolvers with the highest reputation for those 4 particular weeks. In Fig. 4.4 the arrow shows that there are some DNS open resolvers that have been available only for one week with very high reputation (approximately 3,600).

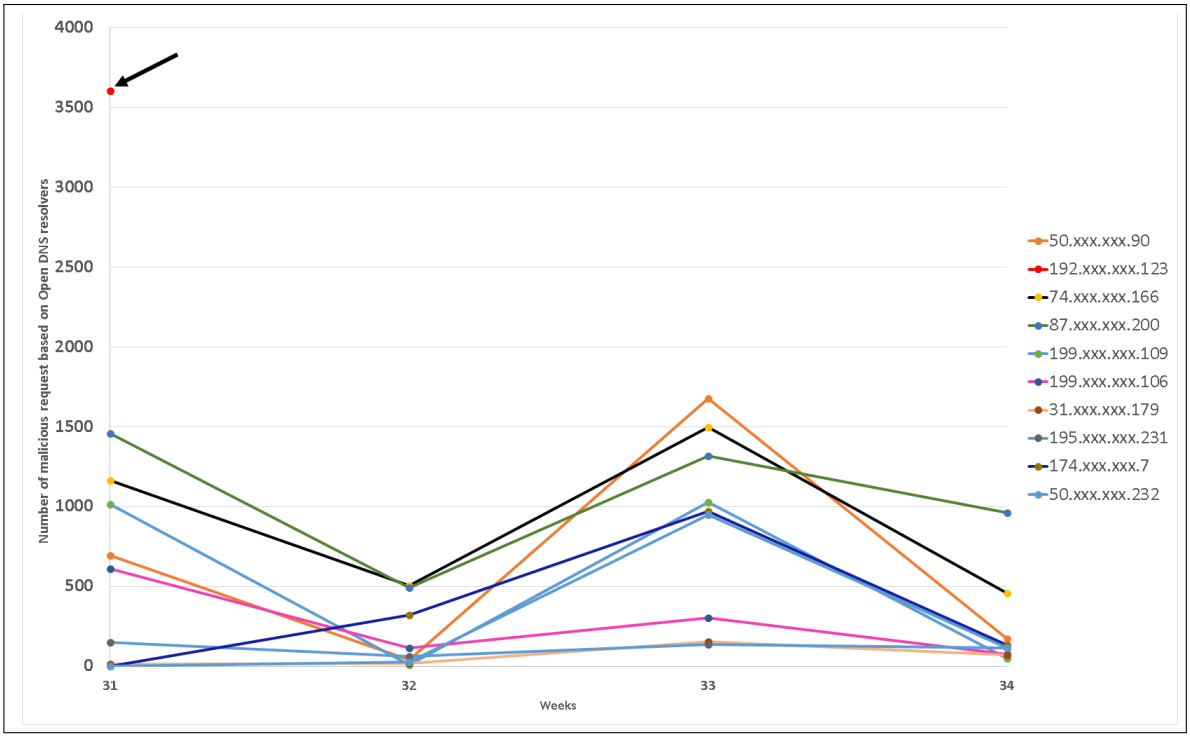

FIGURE 4.4 – Reputation of the top 10 open DNS resolvers that are associated with malware.

In Fig. 4.5, we present the top 10 open DNS resolvers trends with the highest abuse level which means the 10 DNS open resolvers that are contributing with the largest number of malware families in each week for different malicious activities. Also, in Fig. 4.5, the arrow shows that there are DNS open resolvers that, in week 33 of the year, contributed with 87 different malware families.

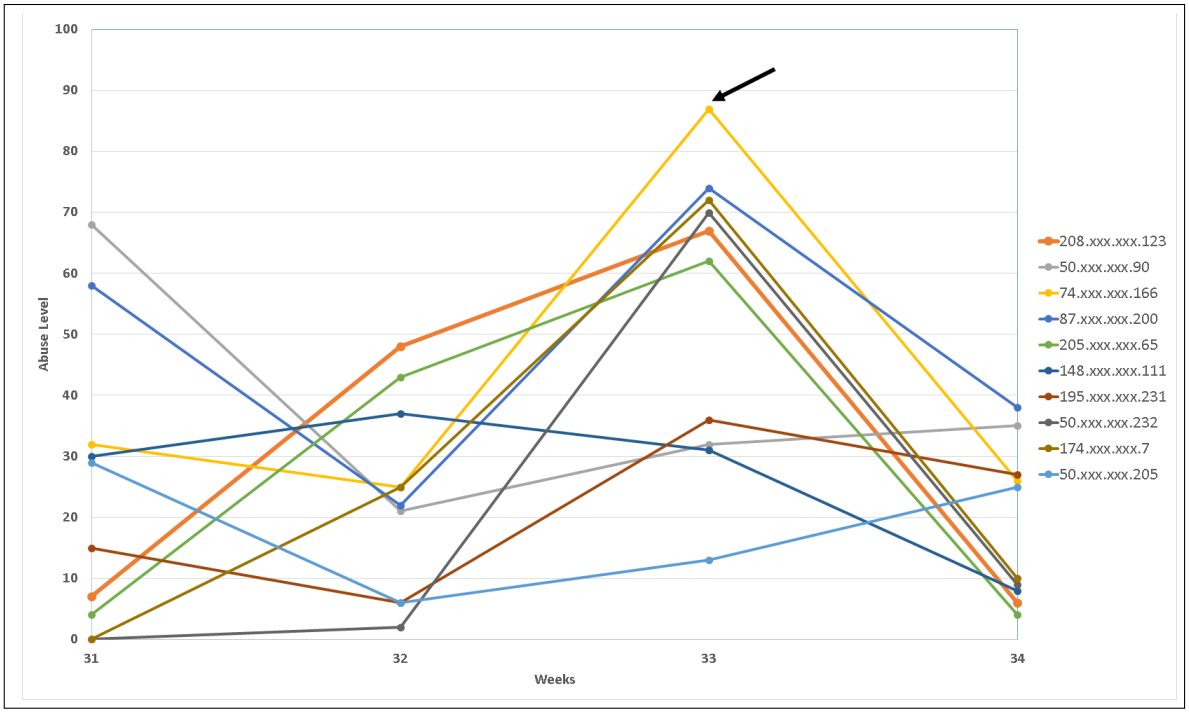

FIGURE 4.5 – Abuse level of top 10 open DNS resolvers associated with unique malware families.

In order to understand more about malware families that had interaction with DNS open resolvers, we used a free online database “VirusTotal" [64] in order to lookup the malware families based on the Hash values related to each record. The result is interesting since it indicates 175 unique malware families. In Fig. 4.6, we provide the top 20 malware families. As shown in Fig. 4.6, the malware family “Sality" was seen for more than 9,000 times in our records, the Malware family “Symmi" for more than 4,200 times and the “Loadmoney" family for more than 1,000 times.

The malware family ’Sality’ is a malicious software that mostly infects executable files in Microsoft Windows operating system as well as remote shared drives and removable devices. It has been discovered for the first time in 2003 [65]. Over the years, ’Sality’ core functionalities remained as in the beginning but it has been getting more sophisticated and improved to become a dynamic malware with a large number of harmful features. Computers which are infected by this malware communicate by using peer to peer networks and receive web addresses to download additional files for different objectives such as spreading spam, stealing highly sensitive information, compromising different types of servers such as web services or different distributed computing employed for malicious activities such as password cracking. This malware tries to disable all the security mechanisms and attempts to modify the security configurations on victims machines including the anti-virus software. A compromised machine

holds different HTTP addresses which point to malicious resources to be downloaded, and after receiving them in encrypted format, bot clients decrypts with RC4 [66], algorithms and execute the commands to the victims machines.

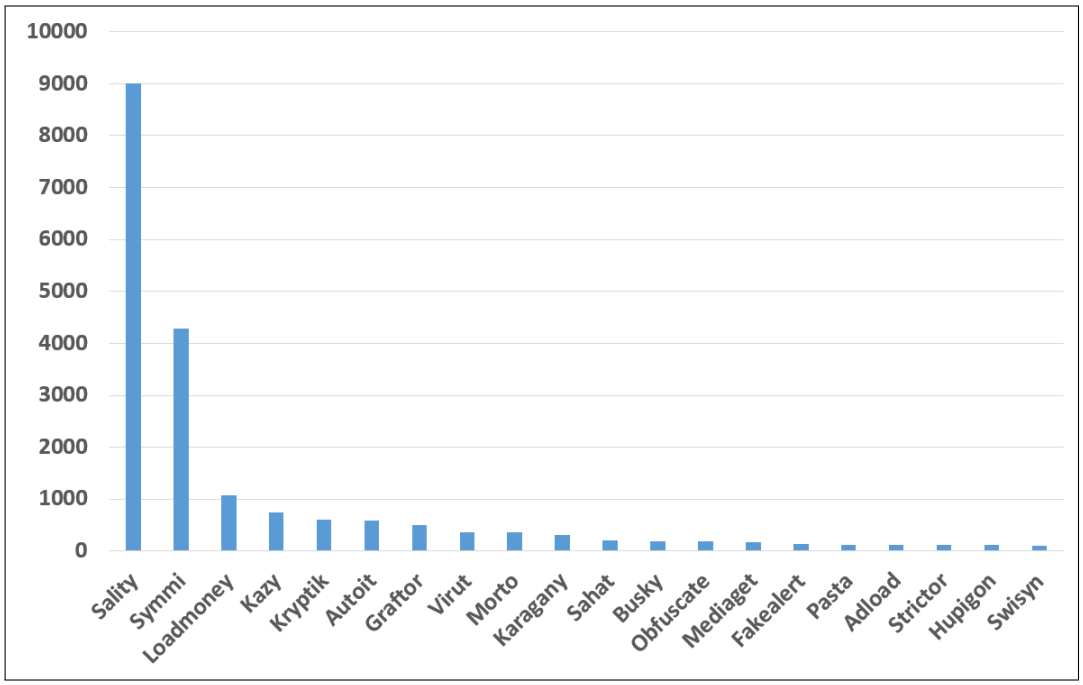

FIGURE 4.6 – Top 20 Malware Families with Highest Interaction to DNS Open Resolvers.

The malware family 'Symmi' belong to Trojans which do not self-replicate. Trojans try to pretend that they have legitimate use. They also hide within viruses and mostly spread manually. Trojans installation methods involve manually executing malicious software, email attachments or malicious Web sites. This malware runs on Microsoft Widows operating systems and aims to collect sensitive system information from user's machine and send it to the remote server that is under attackers control. This trojan carries a list of executable files and scans different folders in victims machine in order to find them. Once it finds those particular executable files, it tries to append the "s" character to end of the file name and , at the same time, it generates a copy of malicious code along with the initial executable file and drops it in different locations in the victims machine. Moreover, by adding some registry keys, it tries to disable UAC (User Account Control prompts) and, on scheduled times, run the executables files and tries to gather sensitive system information as well as Desktop screen-shots which are saved in PNG format and sent to remote servers. It is worth mentioning that it uses Base64 encoding format for transferring the information. Both DNS and HTTP protocols are used to connect to remote servers.

The malware family 'Loadmoney' is a malicious piece of code which is served by fraudulent web pages and could also be injected on some vulnerable legitimate web pages by drive-by-

download methods. It should be noted that Loadmoney is classified as a Potentially Unwanted Program (PUP). The main mission of these malicious codes is to exploit vulnerabilities on programs that are already installed on the victims computers in order to install malicious and unwanted programs which end up with compromising the security of that particular machine. As soon as the victim connects to the Internet, this malware connects to a remote server which is under attackers control in order to get additional malicious codes into the infected machine. This malware affects several Microsoft Windows system files including executable files.

Additionally, we can see that when the reputation increases, the abuse level also increase as depicted in Fig. 4.4 and Fig. 4.5. In week 33 of the year, most of the DNS open resolvers had a high rate in these trends. In Fig. 4.7, we present the 6 different protocols associated to DNS open resolvers and having collaboration with malware. We provide information related to reputation, abuse level and percentage of abuse for each protocol. As a result, we can see that the most used protocol is HTTP with a reputation of 21,045. Moreover, we have large traffic requests for binary files and also DNS, IRC and FTP protocols. In the investigated traffic data, we encountered requests that are not matched with any known protocol (we label them Unknown). One of the interesting parts in protocol analysis is represented by the abuse percentage for each protocol based on the percentage of traffic that acts maliciously. As a result, IRC exhibits the most abuse with 47.80%. In Fig. 4.7, we can see each protocol and its malware association level. For example, HTTP protocol is associated with 1,887 malware. We elaborate HTTP and DNS protocols in more details. We match the malware family using the Hash for each malware family from the malware Databases.

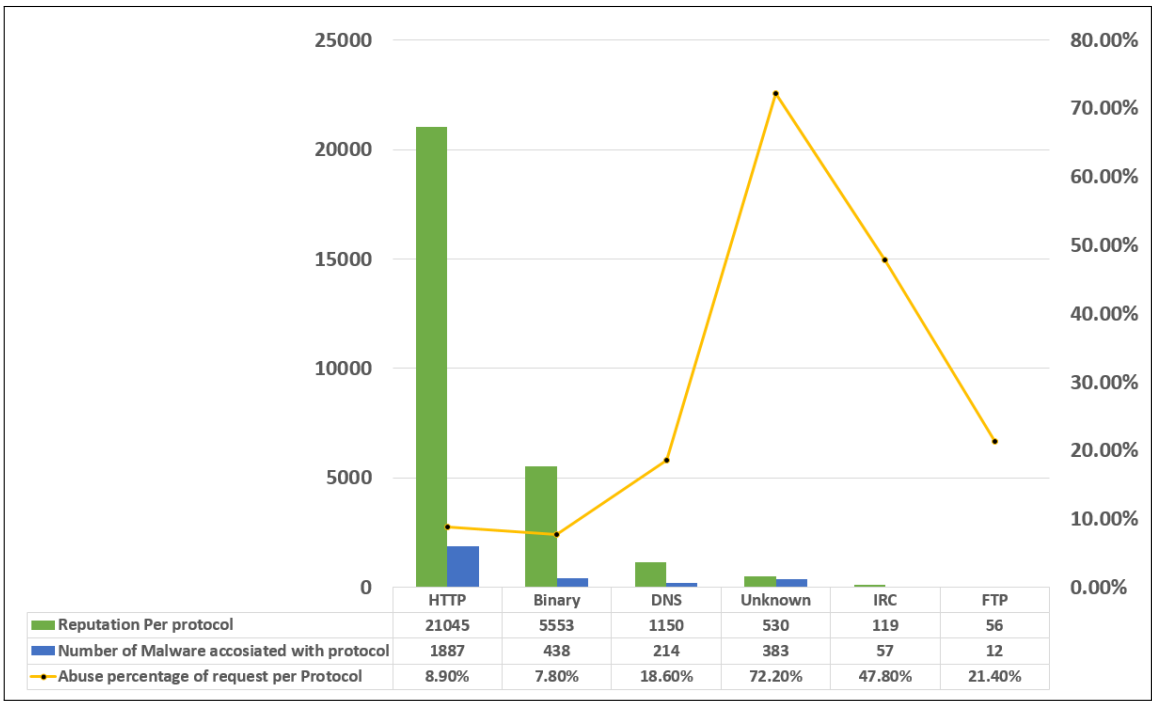

FIGURE 4.7 – Collaboration between Protocols and Malware.

At this stage, a more in-depth understanding was needed with respect to the highly used protocols such as HTTP, DNS and IRC. We performed the corresponding analysis for different HTTP request methods, remote ports and the files extension for HTTP request. HTTP is a request/response protocol which allows a user to place requests to web servers and the servers respond back to the user request. HTTP runs on top of TCP protocol. The web clients can use different request types when accessing resources on the web. These methods are supported by most browsers and the web servers could be configured to accept or refuse specific methods. In our analysis, we found only the GET, POST and HEAD methods. Moreover, GET and POST are the two most common HTTP request methods used on the web. The main difference between GET and POST is that GET requests use Request-URI to pass their parameters to a particular server while POST requests directly use HTTP payload for transferring the parameters. GET request method is very straightforward and can be cached and retained in the browser history. It should not be bookmarked if the URL contains sensitive data. When using HTTP POST for sending payloads, it is difficult to fool the parameters. One important aspect for using POST is that the parameters will not be logged in the system and this can protect against critical information getting written and logged in proxy servers and web servers. Moreover, POST requests are never cached and cannot remain in the browser history. Furthermore, there is no restriction on POST data length while GET hast some restrictions for its length. The HEAD request method is the same as GET with the difference that it instructs the web server to only respond with HTTP header and ignore the asked payload

and thus the server must not return a message-body. This allows an attacker to test different requests without any need to wait for the payload of a particular request. In our data we encountered a large number of GET methods with a total of 11,310 requests. In Fig. 4.8, we provide the trend for each HTTP request method per week. In the case of remote ports in HTTP traffic, the largest number of requests (11,502) is directed toward port 80 which is the default port for web servers access. Also, we encountered remote port 3306 with 315 requests and remote port 8080 with 47 requests in our record data.

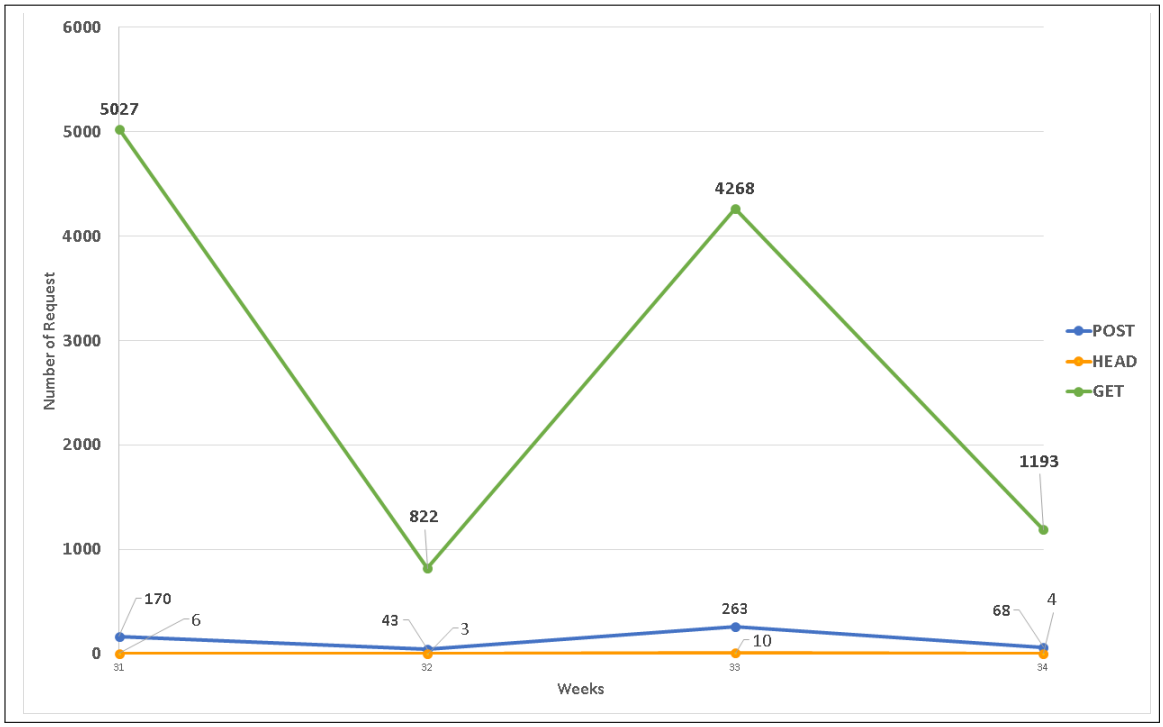

FIGURE 4.8 – Number of HTTP Request per Methods : GET, POST and HEAD.

In addition, in HTTP traffic we see several HTTP requests that ask for particular files. In Fig. 4.9 we show the top file extensions based on our HTTP traffic. We note that most of requested files are executables (with ".EXE" extension). It is obvious that these malware need to download malicious code and these executable files could help attackers to run the malware in the victim's machine. Also we encountered a large number of requests for ".FLV" files (Flash Video). Adobe Flash uses scripting languages and Flash has become one of the major concerns for the Internet security community over the years. Flash players suffers a lot from different common attacks such as buffer and heap overflows. Also they can carry exploits that could have a massive impact.

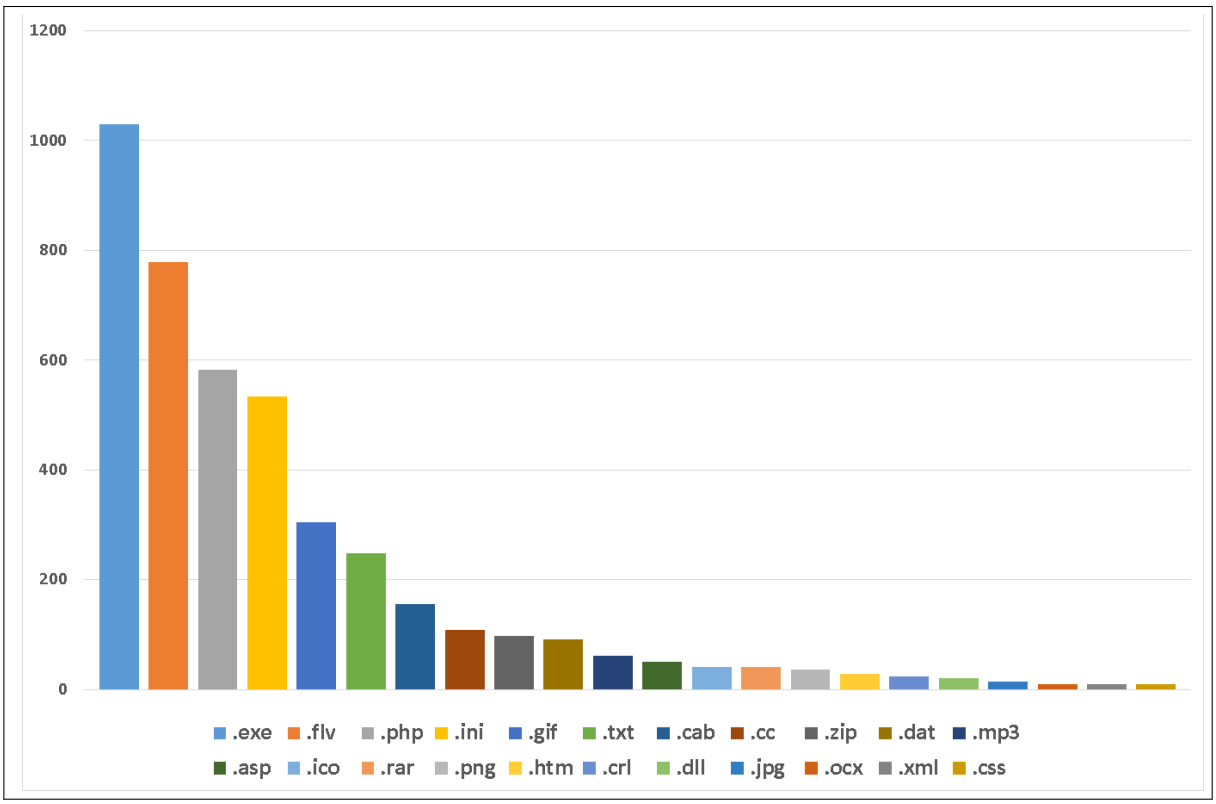

FIGURE 4.9 – The Top File Extensions Based on HTTP Traffic.

Furthermore, other protocols have been encountered in our data regarding DNS open resolvers in collaboration with malware databases : DNS, IRC and FTP. As a result of DNS requests, we observed seven different DNS query types in our data. The most used are 'SOA' type queries with 281 requests out of 1,150, as shown in Fig. 4.10.

In case of IRC traffic, we observed 52 times the channel name "&virtu" joining with remote port 65520 which matches with the malware family Virtu. The latter is a virus written in assembly language that infects SRC and EXE files in windows machines. In addition, it drops the malicious program in windows system folder and modifies system registry. Thus, it runs every time Windows starts and repeatedly tries to connect to remote IRC server on port 65520. After connecting to that remote server, the virus starts receiving commands for downloading other malicious files.

In the case of FTP, we did not get too much data and we only found 52 FTP requests we noticed that 4 requests out of 52 were trying to store some malicious files in a particular server by using the FTP command "STOR".

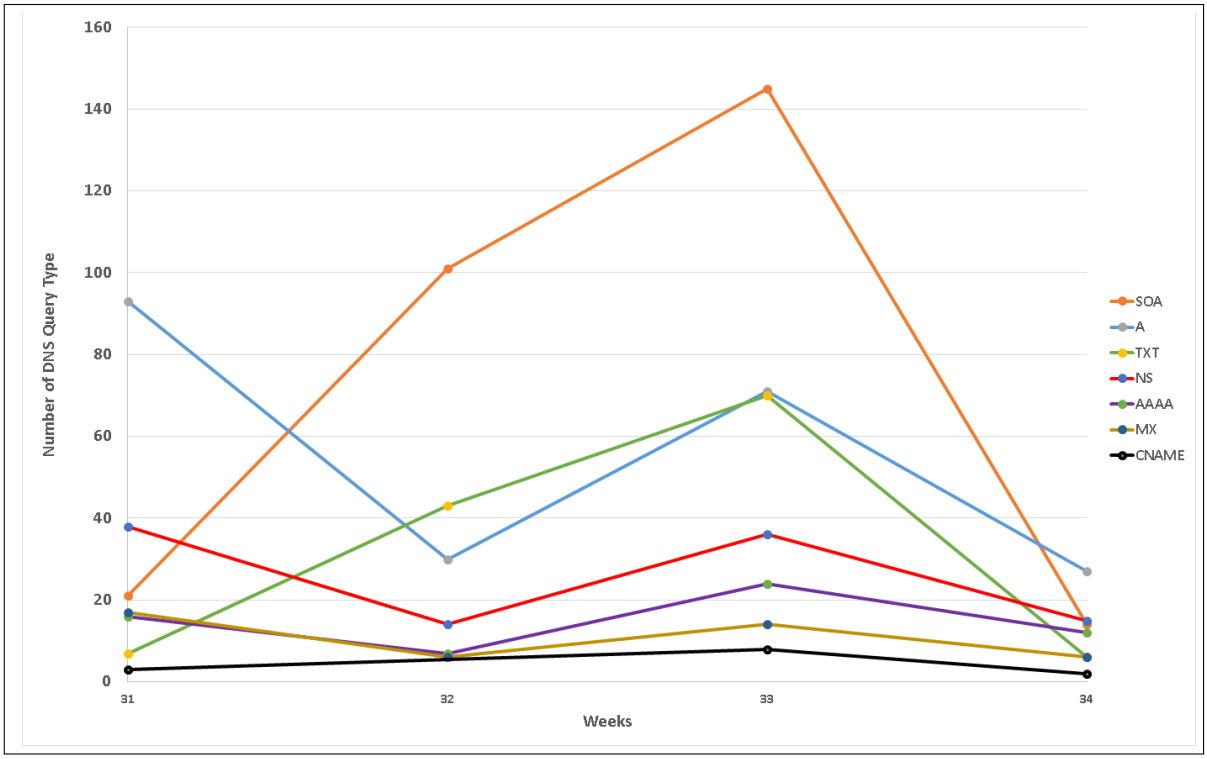

FIGURE 4.10 – The DNS Query Types Associated with Malware.

In Fig. 4.11, we investigate with respect to the top countries that we retrieved from our previous profiling database. The malicious DNS open resolvers are hosted by specific countries as follows : United States (with 19,212 reputation) has collaboration with 2403 malware during the four weeks analysis of the malicious traffic. We noticed that Poland, has a very high percentage of abuse with 58.31% which means that almost 2/3 of all analyzed traffic that is coming from Poland is malicious.

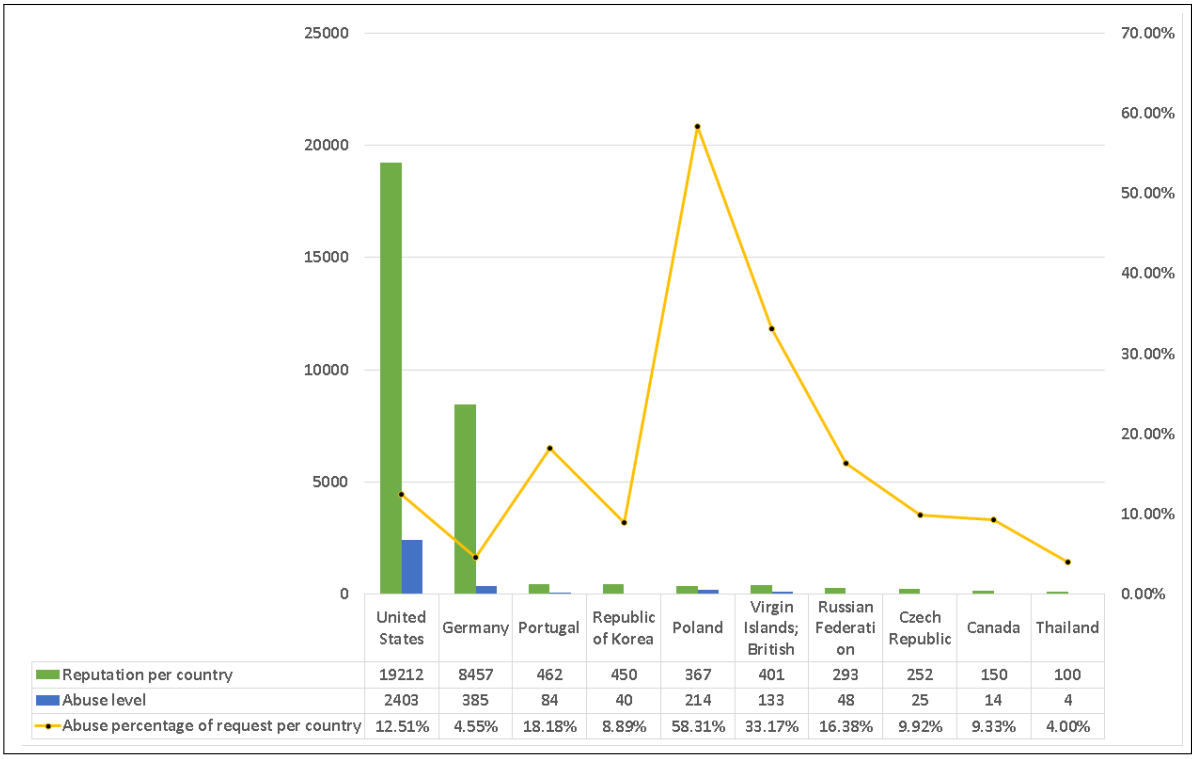

FIGURE 4.11 – Top Country Reputation, Abuse Level and Abuse Percentage.

Moreover, we performed an analysis of our records based on our profiling database in oder to see what type of operating systems are using these malicious DNS open resolvers. We accomplished this by joining the profiling and malware databases with respect to the malicious DNS open resolvers. The result is shown in Fig. 4.12. Thus, “Linux 2.6.9" has a high reputation of 17,602 and an abuse level of 1,753. As shown in the same figure, the highest percentages of abuse are as follows : “Freebsd 7.0" with 93.17%, “Linux 3.0" with 48.72% and “Microsoft Windows server 2003 SP1" with 46.95%.

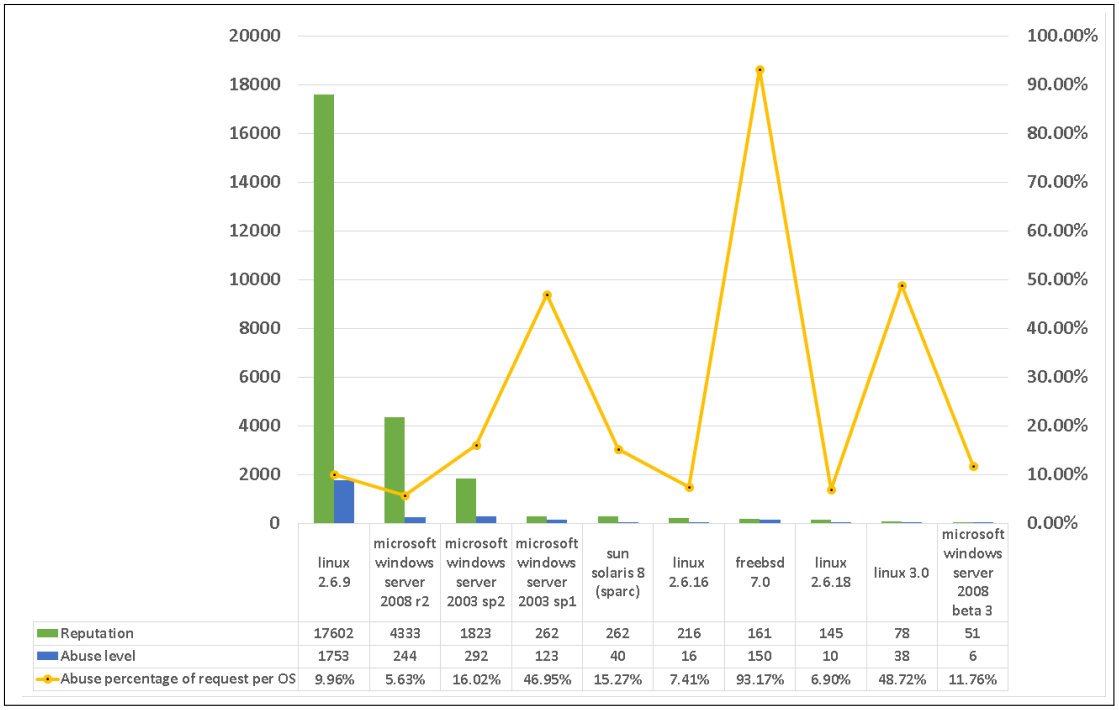

FIGURE 4.12 – Top Operating System Reputation, Abuse Level and Abuse Percentage.

In what follows, we discuss in following three subsections, how DNS service can be hardened and provide recommendations to secure the DNS software.

4.5 Restrict Queries

Surprisingly, name servers intended to provide service for internal (local) users receive a very large number of requests from external (untrusted) users as well. These days, DNS open resolvers are targeted by attackers in order to conduct for different malicious activities such as DNS amplification attacks. To secure the name servers from providing open recursive name server services, one needs to set specific options in the /etc/named.conf file. On the other hand, one also needs to restrict users who are not in that particular network. Here is an example for restricting answers to recursive queries only to specific network users :

$$allow - query\{192.168.100.0/24\}; \tag{4.5}$$

$$allow - recursion\{192.168.100.0/24\}; \tag{4.6}$$

4.6 Dedicated Function for Name Servers

Name servers should have a dedicated function ; an authoritative name server serves resource records (RRs) records from its zone, which is hosted in those servers. A resolving name server should serve 'RRs' from its cache or is resolve directly by iterative queries. A name server could be configured as an authoritative name server, resolving name server or both. Our recommendation for name servers is to provide one dedicated service per server, either an authoritative name server or a resolving name server. An authoritative name server should be ready to provide name resolution only for the zones that are hosted in that particular server. So, it is necessary to turn off the recursion for this type of name servers. By disabling the recursion, one avoids an authoritative name server from answering queries on behalf of other name servers [67]. Moreover, this mitigates the cache poisoning security threats on authoritative name servers and eliminates the vulnerability of being used as reflectors for DDoS attacks [68]. In 'BIND' name servers, recursion is disabled by setting the corresponding options in the 'BIND' configuration file as follows :

$$\textit{Options } \{\textit{recursion no};\}; \tag{4.7}$$

4.7 DNS Geographic and Network Distribution

According to RFC2182 [59], the number of authoritative name servers related to each domain should be at least two or more but preferably no more than 7. Also, it is essential for name servers to be separated in different IP classes and in topologically dispersed locations. It should be noted that, in a network, dispersion must take into account that all name servers should not rely on a single router/switches or firewall or even on a single subnet or the same network connection. Geographic dispersion should ensure that not all DNS servers are in the same location (physically).

4.8 Summary

In this chapter, we have addressed, different aspects of DNS open resolvers security measurements and security issues regarding both resolver name servers and authoritative name servers. These aspects are related to root zone attacks (Upward Referrals), 'BIND Authors' and DNS server software version and their vulnerabilities. Moreover, in order to provide information regarding DNS abuses and the level of maliciousness for open resolvers, we analyzed our DNS open resolvers with malware blacklist database. It allowed us to identify the abuse parameters for DNS open resolvers. By joint analysis of the malware database and the DNS open resolvers data we found out that many DNS open resolvers are related to malicious code

and are associated with existing malware engaged in conducting malicious activities. The malware database provides malware traffic data that allowed us to perform protocol analysis as well.

Chapitre 5

Conclusion and Future Work

In this research study, we provide an approach for identifying DNS open resolvers around the world. We provide an analysis for profiling DNS open resolvers, DNS open resolvers security measurements, DNS open resolvers use and abuse along with the collaboration of DNS open resolvers with existing malware from malware blacklist databases.

This research study explained how DNS open resolvers could be used in DNS reflection DDoS attacks. In addition, it also elaborated on the current “health state" of DNS servers around the world. DNS reflection DDoS attacks happen when a small DNS request could lead to a much larger response. It provides attackers with powerful means to launch a DDoS attack to the targeted systems and networks. The bandwidth amplification factors (BAF) in DNS reflection attacks may reach levels well over 28 to 54 times [69]. In the scope of this study, we investigated DNS open resolvers which represent significant elements of DNS reflection DDoS attacks.

This study offers deep insights regarding several aspects of open DNS resolver identification and awareness generation. With respect to the identification of DNS open resolvers, we used a data set for identifying the DNS open resolvers which is TLD zone files. With respect to the different aspects of DNS resolvers profiling, we elaborated on DNS resolvers GeoLocation, IP allocations, connection speed, device type, and purposes. With respect to the different aspects of the DNS open resolvers security measurements, we illustrated the Root Zone Attacks (Upward Referrals), ’BIND Authors’, which could be a replacement for DNS open resolvers in DNS reflection attacks and DNS server software version distribution along with their corresponding vulnerabilities. In regard to different aspects of the DNS open resolvers in collaboration with malwares, our analysis showed that these DNS open resolvers are exhibiting malicious behaviour and contributing to malicious activities. Moreover, we provided protocol analysis based on specific malware traffic that is associated with DNS open resolvers under investigation.

In this research study, we show that there are several issues with DNS protocol and DNS

software. As well, we found that there are lots of problems due to bad and poor configuration of these services. It should be noted that, authoritative name servers that are not configured correctly could act as amplifiers for attackers with security issues like Upward Referrals. In this context, we provide several recommendations to solve various security issues in DNS services.

One of our important contributions in this research study is that we found an unexpected feature in BIND name servers that can pose a serious vulnerability. To the best of our knowledge, this is the first time that this vulnerability is mentioned and investigated with respect to DNS reflection DDoS attacks. With this vulnerability, attackers do not need to look for the list of DNS open resolvers in order to amplify their traffic which is the goal of any amplification based DDoS attack. This vulnerability provides an attractive alternative to DNS open resolvers in DNS reflection DDoS attacks. This allows to place standard queries even to authoritative 'BIND' name servers for the 'BIND authors' list. Such queries are small in size, but the response is can be large enough to make DNS amplification DDoS attacks.

It is worth mentioning that our result with respect to the device type for the investigated DNS open resolvers shows that there are equipments built by well known manufacturers that are acting as DNS open resolvers. These devices need to get patched in order to eliminate the vulnerability that we found in our analysis.

In the recent past, large-cap organizations such as Google Ideas and Arbor Networks are more and more developing applications like 'Digital Attack Map' which can provide users with historical trends of attacks. The results of this study can be very useful in generating specific rules to significantly improve such awareness on the existing Internet Security platforms.

During our investigation of DNS open resolvers regarding DNS amplification DDoS attacks, we have discovered research directions, which could represent extensions of this research study as follows :

1. **DNS Amplification Detection Module :** According to our result and statistics, we could provide DNS Amplification Detection Module in order to detect these types of DDoS attacks in a small scale. We could use some of our results such as length of domains, top TLD (used by most of the DNS open resolvers), statistics from malware databases regarding protocol analysis and DNS open resolvers list which could act as blacklist. This valuable information could be used to develop an intelligent system for detecting malicious traffic, and we need reasonable factors to give weight or cost to those factors.

2. **Investigating NTP based DDoS Attacks :** Recently, companies such as Cloudflare, experienced a large DDoS attack based on NTP amplification [4]. The Network Time Protocol (NTP) servers support a monitoring option that lets administrators to ask

the server for some statistics of connected clients. This feature is named “monlist" command. The NTP servers, when configured to accept and support the monlist queries, could receive up to 440 bytes payload. On the average, monlist requests amplify the traffic by an amplification factor of 556.9, the highest amplification factor in UDP-based amplification attacks [69]. Thus, monlist may be abused for generating amplification attacks. The attack mechanics consist of sending instructions to the zombie armies to send “get monlist" requests to a vulnerable NTP server. This operation needs to be joined with IP source address spoofing. These types of attacks, which work almost exactly as DNS amplification attacks could be analyzed in depth in order to gather relevant insights for the Internet community.

As a final remark, we can state that presently DNS servers are not secure enough and their weaknesses represent a real threat for Internet security, given that they can be exploited by attackers and allow them the possibility to easily conduct large scale DDoS attacks.

Appendix A

In Fig .1, we present a PHP script for testing open DNS resolvers. In Fig .2, we present a PHP script for fetching open DNS resolvers speed connection based on MAXMIND database. In Fig .3, we present a Part of PHP tools for demonstrating our open DNS resolvers in map, by using Google map API.

```php
<?
// Create connection
$con=mysql_connect("localhost","root","SAn83890#&2#94762");

$db_selected = mysql_select_db("Nameservers", $con);

// Check connection
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }

$z=$argv[1]*1;

for($i=1; $i<=$z; $i++) {

$result = mysql_query("SELECT * FROM ip1 where address=''  ORDER BY RAND() limit 1 ");

if($result === false ) {
      die(mysql_error());
}

while($row = mysql_fetch_array($result))
  {

$ip1 = $row['remote_host'] ;

$input = array("ask.com", "msn.com", "yahoo.com", "bing.com","google.com","youtube.com"
,"Amazon.com","live.com","twitter.com","blogspot.com","linkedin.com","ebay.com","tumblr.
,"wordpress.com","microsoft.com","tmall.com","craigslist.org","pinterest.com","instagram
$rand_keys = array_rand($input, 2);

$ask= $input[$rand_keys[0]];

$output = exec('dig @'.$ip1.' '.$ask.' >> '.$argv[2].'.txt');

$output= $output.'<hr />';
mysql_query("update ip1 set address='".$output."' WHERE id='".$row['id']."' ");

}
  }
mysql close($con);
```

FIGURE .1 – PHP Script for Testing DNS Open Resolvers

```php
<?
$con=mysqli_connect("localhost","root","","          ");

// Check connection
if (mysqli_connect_errno($con))
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }


$result = mysqli_query($con,"SELECT * FROM newip1 where  addtospeed='0' ");

include("geoipcity.inc");
include("geoipregionvars.php");

$gi = geoip_open("                          GeoIPNetspeed.dat",GEOIP_STANDARD);


while($row = mysqli_fetch_array($result))
  {



    $org = @geoip_country_id_by_addr($gi,$row['ip']);

    if ($org == GEOIP_UNKNOWN_SPEED){
  $org1= "Unknown";
}else if ($org == GEOIP_DIALUP_SPEED){
  $org1= "Dailup";
}else if ($org == GEOIP_CABLEDSL_SPEED){
  $org1= "Cable/DSL";
}else if ($org == GEOIP_CORPORATE_SPEED){
  $org1= "Corporate";
}
   mysqli_query($con,"update newip1 set speed = '".@mysqli_real_escape_string($con,$org1)."
  , addtospeed='1' WHERE id='".$row['id']."' ");
//die(mysql_error());

  }


geoip_close($gi);


  ?>
```

FIGURE .2 – PHP Script Fetching Speed Connection Information

```
<html>
  <head>
    <title>Simple Map</title>
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no">
    <meta charset="utf-8">
    <style>
      html, body, #map-canvas {
        margin: 0;
        padding: 0;
        height: 100%;
      }
    </style>
     <script src="js/jquery-1.9.1.min.js"></script> <!-- jQuery library -->
  <script src="js/jquery.easing.1.3.min.js"></script> <!-- jQuery Easing -->
 <script type="text/javascript" src="http://maps.google.com/maps/api/js?sensor=true"></script>
  <script type="text/javascript" src="js/gmaps.js"></script>
  <script type="text/javascript">
    var map;
    $(document).ready(function(){

      // main directions
      map = new GMaps({
        el: '#map-canvas', lat: 51.5073346, lng: -0.1276831, zoom: 6, zoomControl : true

      });
      // add address markers

     <?
            $con=mysqli_connect("                          j");

// Check connection
if (mysqli_connect_errno($con))
  {
  echo "Failed to connect to MySQL: " . mysqli_connect_error();
  }
     if($_GET['speed'] != ''){

         $filter="and speed = '".$_GET['speed']."'";
     }
         $result = mysqli_query($con,"SELECT * FROM newip1 where  latitude !='' and longitude !='' ".$filter." ");

     while($row = mysqli_fetch_array($result))
       {
        ?>
      map.addMarker({ lat: <? echo $row['latitude'] ?>, lng: <? echo $row['longitude'] ?>, title: '<? echo $row['ip'] ?>',
      infoWindow: { content: '<p><? echo $row['ip'] ?> - <? echo $row['city'] ?></p>' } ,icon: "iconss.png"  });

    <?
       }
        ?>

    });
```

FIGURE .3 – Part of PHP Tools to Demonstrate DNS Open Resolvers in Map

Bibliographie

[1] M. Prince. The DDoS that almost broke the internet. March 2013. Cloud Flare, Also available as http://blog.cloudflare.com/the-ddos-that-almost-broke-the-internet.

[2] M. Prince. The DDoS that knocked spamhaus offline (and how we mitigated it). March 2013. Cloud Flare, Also available as http://blog.cloudflare.com/the-ddos-that-knocked-spamhaus-offline-and-ho.

[3] S. Gallagher. How the most massive botnet scam ever made millions for estonian hackers. November 2011. Ars Technica, Also available as http://goo.gl/IC4UOw.

[4] M. Prince. Technical details behind a 400gbps NTP amplification DDoS attack. February 2014. Also available as http://blog.cloudflare.com/technical-details-behind-a-400gbps-ntp-amplification-ddos-attack.

[5] R. Vaughn and E. Gadi. DNS amplification attacks. March 2006. Also available as http://www.isotf.org/news/DNS-Amplification-Attacks.pdf.

[6] G. Ateniese and S. Mangard. A new approach to DNS security (DNSSEC). In *Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 86–95, Philadelphia, USA, 2001. ACM.

[7] A. Ellis. DNS reflection defense. 2013. Also available as https://blogs.akamai.com/2013/06/dns-reflection-defense.html.

[8] Alert (ta13-088a), DNS amplification attacks. US-CERT, July 2013. Also available as https://www.us-cert.gov/ncas/alerts/TA13-088A.

[9] C. Marshall. Do not blame open recursives for DDoS attacks, why you should implement BCP38. April 2013. Dyn, Also available as http://goo.gl/iqjTNu.

[10] BIND the most widely used name server software. Also available as http://www.isc.org/downloads/bind/.

[11] The 'heartbleed' security flaw that affects most of the internet. 2014. Google, Also available as http://www.cnn.com/2014/04/08/tech/web/heartbleed-openssl/.

[12] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. DNS security introduction and requirements. March 2005. ISC, Also available as https://www.ietf.org/rfc/rfc4033.txt.

[13] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. Resource records for the DNS security extensions. March 2005. NIST, Also available as https://www.ietf.org/rfc/rfc4034.txt.

[14] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. Protocol modifications for the DNS security extensions. March 2005. NIST, Also available as https://www.ietf.org/rfc/rfc4035.txt.

[15] B. Laurie, G. Sisson, R. Arends, and D. Blacka. DNS security (DNSSEC) hashed authenticated denial of existence. March 2008. VeriSign and Inc., Also available as https://www.ietf.org/rfc/rfc5155.txt.

[16] Y. Takano, R. Ando, T. Takahashi, S. Uda, and T. Inoue. A measurement study of open resolvers and DNS server version. In *Internet Conference*, pages 23–32. Internet Conference Executive Committee, 2013.

[17] Google public DNS and rate-limiting querie. 2013. Google, Also available as https://developers.google.com/speed/public-dns/docs/security.

[18] Open resolver project. Also available as http://openresolverproject.org/.

[19] A. Aina, J. Akkerhuis, K. Claffy, S. Crocker, D. Karrenberg, J. Ihrn, R. Joffe, M. Kosters, A. Mankin, and R. Mohan. SSAC advisory SAC008 DNS distributed denial of service (DDoS) attacks. *Security and Stability Advisory Committee (SSAC)*, March 2006.

[20] The continuing denial of service threat posed by DNS recursion (v2.0). *US-CERT and Government Organization*, 2006. US-CERT.

[21] S. Singh. Denial of service (DOS)/DDOS attack methods and preventions.

[22] V. Paxson. An analysis of using reflectors for distributed denial-of-service attacks. *ACM SIGCOMM Computer Communication Review*, 31(3) :38–47, 2001.

[23] M. Tanase. IP spoofing : an introduction. *Security Focus*, 11, 2003.

[24] F. Baker and P. Savola. Ingress filtering for multihomed networks. 2004. BCP 84, RFC 3704, March.

[25] IP source guard. Cisco. Also available as http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst6500/ios/12-2SX/configuration/guide/book/ipsrcgrd.html.

[26] R. Beverly, A. Berger, and Y. Hyun. Understanding the efficacy of deployed internet source address validation filtering. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, pages 356–369. ACM, 2009.

[27] Worldwide infrastructure security report. 2010. Arbor Networks, Also available as http://www.arbornetworks.com/dmdocuments/ISR2010_EN.pdf.

[28] M. Antonakakis, D. Dagon, X. Luo, R. Perdisci, W. Lee, and J. Bellmor. A centralized monitoring infrastructure for improving DNS security. In *Recent Advances in Intrusion Detection*, pages 18–37. Springer, 2010.

[29] M. Antonakakis, R. Perdisci, D. Dagon, W. Lee, and N. Feamster. Building a dynamic reputation system for DNS. In *USENIX security symposium*, pages 273–290, 2010.

[30] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon. From throw-away traffic to bots : Detecting the rise of dga-based malware. In *Proceedings of the 21st USENIX security symposium*, pages 491–506, 2012.

[31] L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi. Exposure : Finding malicious domains using passive DNS analysis. In *NDSS*, 2011.

[32] Botnet. Also available as http://en.wikipedia.org/wiki/Botnet.

[33] E. Cooke, F. Jahanian, and D. McPherson. The zombie roundup : Understanding and detecting and and disrupting botnets. In *Proceedings of the USENIX SRUTI Workshop*, volume 39, page 44, 2005.

[34] A. Ramachandran and N. Feamster. Understanding the network-level behavior of spammers. In *ACM SIGCOMM Computer Communication Review*, volume 36, pages 291–302. ACM, 2006.

[35] N. Ianelli and A. Hackworth. Botnets as a vehicle for online crime. *CERT Coordination Center*, 1(1) :28, 2005.

[36] M. Koetter. Know your enemy : Tracking botnets. 2008. Also available as http://honeynet.org/papers/bots/.

[37] M. Feily, A. Shahrestani, and S. Ramadass. A survey of botnet and botnet detection. In *Emerging Security Information and Systems and Technologies and 2009. SECURWARE'09. Third International Conference on*, pages 268–273. IEEE, 2009.

[38] Z. Zhu, G. Lu, Y. Chen, Z. Fu, P. Roberts, and K. Han. Botnet research survey. In *Computer Software and Applications and 2008. COMPSAC'08. 32nd Annual IEEE International*, pages 967–972. IEEE, 2008.

[39] Over one million potential victims of botnet cyber crime. 2007. FBI, Also available as http://www.fbi.gov/news/pressrel/press-releases/over-1-million-potential-victims-of-botnet-cyber-crime.

[40] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver. The spread of the sapphire/slammer worm, 2003.

[41] A. Rafael, G. Macia Fernandez, and P. Garcia-Teodoro. Survey and taxonomy of botnet research through life-cycle. *ACM Computing Surveys (CSUR)*, 45(4) :45, 2013.

[42] P. Vixie. Extension mechanisms for DNS (EDNS0). August 1999. ISC, Also available as https://tools.ietf.org/html/rfc2671.

[43] T. Moschos G. Kambourakis, D. Geneiatakis, and S. Gritzalis. Detecting DNS amplification attacks. In *Critical Information Infrastructures Security*, pages 185–196. Springer, 2008.

[44] X. YE and Y. YE. A practical mechanism to counteract DNS amplification DDoS attacks. *Journal of Computational Information Systems*, 9(1) :265–272, 2013.

[45] C. Sun, B. Liu, and L. Shi. Efficient and low-cost hardware defense against DNS amplification attacks. In *Global Telecommunications Conference and 2008. IEEE GLOBECOM 2008. IEEE*, pages 1–5. IEEE, 2008.

[46] S. Di Paola and D. Lombardo. Protecting against DNS reflection attacks with bloom filters. In *Detection of Intrusions and Malware and and Vulnerability Assessment*, pages 1–16. Springer, 2011.

[47] H. Yu, X. Dai, T. Baxley, and J. Xu. A real-time interactive visualization system for DNS amplification attack challenges. In *Computer and Information Science and 2008. ICIS 08. Seventh IEEE/ACIS International Conference on*, pages 55–60. IEEE, 2008.

[48] T. Rozekrans and J. de Koning M. Mekking. Defending against dns reflection amplification attacks. 2013.

[49] U. Tupakula, V. Varadharajan, and S. Pandalaneni. DoSTRACK : a system for defending against DoS attacks. In *Proceedings of the 2009 ACM symposium on Applied Computing*, pages 47–53. ACM, 2009.

[50] P. Vixie. DNS response rate limiting (DNS RRL). April 2012. Also available as http://ss.vix.su/~vixie/isc-tn-2012-1.txt.

[51] M. Santcroos and M. Kolkman. DNS threat analysis. *NLnet Labs*, 2007.

[52] P. Ferguson and D. Senie. Network ingress filtering : Defeating denial of service attacks which employ ip source address spoofing. 2000.

[53] P. Savola F. Baker. Ingress filtering for multihomed networks. 2004.

[54] J. Stewart. DNS cache poisoning–the next generation. 2003.

[55] L. Grangeia. DNS cache snooping. 2004. Technical report and Security Team Beyond Security.

[56] T. Holz, C. Gorecki, K. Rieck, and F. Freiling. Measuring and detecting fast-flux service networks. In *NDSS*, 2008.

[57] Domain zone file & zone changes downloads. Also available as http://www.premiumdrops.com/zones.html.

[58] Report : More than 250m domain names have now been registered and almost half are .com and .net. April 2013. Also available as http://techcrunch.com/2013/04/08/internet-passes-250m-registered-top-level-domain-names/.

[59] R. Elz, R. Bush, S. Bradner, and M. Patton. Selection and operation of secondary DNS servers. 1997. Harvard University, Also available as https://tools.ietf.org/html/rfc2182.

[60] Digital attack map - DDoS attacks around the globe. October 2013. Also available as http://www.digitalattackmap.com/.

[61] Nmap 6.40 released ! new scripts, new signatures, better performance. August 2013. Also available as http://seclists.org/nmap-announce/2013/1.

[62] C. Fachkha, E. Bou-Harb, and M. Debbabi. Fingerprinting internet DNS amplification DDoS activities. In *New Technologies, Mobility and Security (NTMS), 2014 6th International Conference on*, pages 1–5. IEEE, 2014.

[63] BIND 9.3 security vulnerabilities. Also available as http://www.cvedetails.com/vulnerability-list/vendor_id-64/product_id-144/version_id-21860/ISC-Bind-9.3.0.html.

[64] Virustotal - free online virus, malware and url scanner. Also available as https://www.virustotal.com/.

[65] E. Chien A. Thigpen. W32.sality. 2003. Symantec, Also available as http://www.symantec.com/security_response/writeup.jsp?docid=2006-011714-3948-99.

[66] Rc4. 1987. Also available as http://en.wikipedia.org/wiki/RC4.

[67] R. Chandramouli and S. Rose. Secure domain name system (DNS) deployment guide. *NIST Special Publication*, pages 800–81, 2006.

[68] J. Damas and F. Neves. Preventing use of recursive nameservers in reflector attacks. 2008. BCP 140 and RFC 5358 and October.

[69] Alert (ta14-017a) UDP-based amplification attacks. 2014. US-Cert, Also available as https://www.us-cert.gov/ncas/alerts/TA14-017A.