UNIVERSITY OF TWENTE.

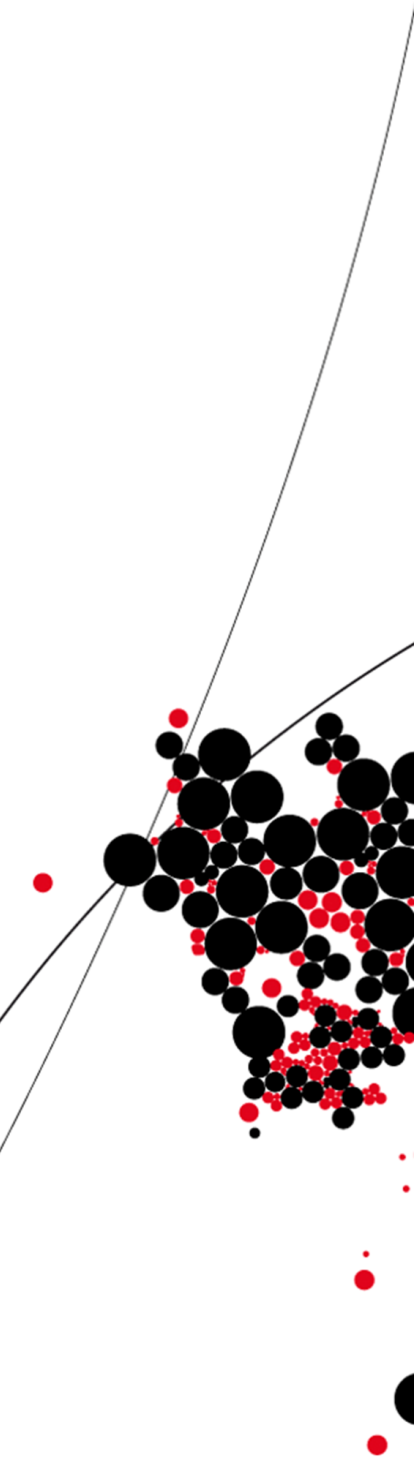
Faculty of Electrical Engineering, Mathematics & Computer Science

The evolution of DNS encryption and the effect on digital forensic research

Anja Scherjon - 434766

Bachelor Thesis - Forensic Research

June 2020



SAXION UNIVERSITY OF APPLIED SCIENCES

DACS Design and Analysis of Communication Systems

Supervisors:
Dr. ir. Roland van Rijswijk-Deij
Dr. ir. Richard Brinkman

Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
Drienerlolaan 5
7522 NB Enschede
The Netherlands

Preface

I found this research because I went looking for prof. Aiko Pras, chair of the Design and Analysis of Communication systems (DACS) group at the University of Twente, after I heard his presentation on ”how a cyberattack could bring down the internet” at the ”Universiteit van Nederland” podcast. When I contacted him, I was welcomed with open arms. An effort was made to find an appropriate topic for my graduating thesis and I am very happy with the research and everything I have learned from the DACS group at the University of Twente. Even though the COVID-19 pandemic hit after the first month of this research, I am proud of the accomplished results. Firstly, I want to thank my supervisor Roland van Rijswijk-Deij for the constant help and support during this process. I also want to thank Etienne Khan for helping me with all my questions and getting the experiment working and measured. Lastly, I want to emphasise the flexibility shown by the experts, and their willingness to do the interviews digitally. I hope all the hard work shines through.

Summary

The Domain Name System (DNS) is responsible for connecting domain names to IP addresses. Your computer sends the (typically) unencrypted DNS request (query) to a resolver. This resolver will ask different servers for the allocated IP address and the response will be sent back to your computer. Recent efforts have worked towards including encryption in the DNS, with the latest protocols being announced in 2016: DNS-over-TLS (DoT) and in 2018: DNS-over-HTTPS (DoH). These protocols use TLS and HTTPS to encrypt the DNS data between the computer and resolver. This could have a major impact on the visible DNS information for incident researchers and law enforcement. Therefore, the research question is: "What are the main restrictions encountered by incident researchers from the encryption of DNS and what are alternatives to gain the wanted information?" This will be answered by researching: current leading types of DNS encryption, currently available investigative options and use of DNS information, experts experiences and advice when encountering these forms of DNS encryption and the potential misuses of encrypted DNS. Results are gathered through qualitative interviews and a quantitative experiment. The main results from the interviews show that most experts were not fully aware of the possible impact of the new encryption protocols. Different ways to circumvent the encryption were stated. However, most have not chosen to implement these yet. It can be concluded that companies that use data gathered between client and resolver will experience the most effect from these encryptions of the DNS. Real-time detection and trace-backs to the original client will become increasingly difficult. The best alternatives as stated by the experts are: changing the location of detection to endpoints and the resolver itself, and offering DoT and DoH within the local network. The main result from the experiment is that it is possible to set up a DNS tunnel over DoH in a day. Therefore, it can be concluded that it is trivial to use the new encryption for criminal activity like the in- and exfiltration of data or control of a botnet by using a DNS tunnel. Further research should look into the effect DoT and DoH will have on the available threat intelligence and other passive DNS sources; the effect of other security aspects after the encryption that have not specifically been looked into in this research. Finally, more possibilities to limit DNS tunneling, normally and over encryption and more software for local DoT and DoH resolvers should be developed and implemented.

Samenvatting

Het Domain Name System (DNS) is verantwoordelijk voor het zoeken van IP-adressen bij domeinnamen. Het werkt door onversleutelde verzoeken (query's) te versturen naar een resolver. Deze resolver zal bij verschillende servers het bijbehorende IP-adres opvragen en het antwoord naar de computer terugsturen. Twee grote nieuwe protocollen voor DNS-encryptie zijn gepresenteerd in 2016: DNS-over-TLS (DoT) en in 2018: DNS-over-HTTPS (DoH). Deze protocollen gebruiken TLS en HTTPS om data tussen de computer en resolver te versleutelen. Dit kan een grote impact hebben op incidentonderzoekers, daarom is de hoofdvraag van dit onderzoek: "Wat zijn de voornaamste beperkingen die incidentonderzoekers tegenkomen na de encryptie van het DNS, en wat zijn alternatieven om deze informatie toch te verkrijgen?" Deze vraag zal beantwoord worden door te kijken naar de laatste vormen van DNS-encryptie, het verkrijgen en gebruik van DNS-informatie, ervaringen en advies van experts over de omgang met de nieuwe encryptievormen en misbruik van versleuteld DNS. De resultaten zijn verzameld aan de hand van kwalitatieve interviews en een kwantitatief experiment. De voornaamste resultaten van de interviews laten zien dat de meeste experts zich niet volledig bewust waren van de mogelijke impact van de nieuwe versleutelingen. Verschillende manieren werden genoemd om de informatie alsnog te verzamelen. Deze zijn echter veelal nog niet geïmplementeerd. Er kan worden geconcludeerd dat bedrijven die informatie verzamelen tussen de client en resolver het meeste effect van de DNS-encryptie zullen ervaren. Realtime detectie en het achterhalen van de originele client zullen moeilijker worden. De beste alternatieven genoemd door de experts zijn: het verplaatsen van de sensoren naar de endpoints of resolvers toe, en het aanbieden van DoT en DoH op lokale resolvers. Het voornaamste resultaat van het experiment is dat het mogelijk is om een DNS tunnel over DoH op te zetten in een dag. Er kan worden geconcludeerd dat het gemakkelijk is om de encryptie voor criminele activiteiten in te zetten zoals het in- en exfiltreren van data via een DNS tunnel. Vervolgonderzoek kan kijken naar: het effect van DoT en DoH op de beschikbare threat intelligence en andere Passive-DNS-bronnen; het effect op andere beveiligingsaspecten na de encryptie waarnaar niet is gekeken in dit onderzoek. Tot slot moeten er meer mogelijkheden komen om DNS tunneling te beperken en moet er meer software voor locale DoT en DoH resolvers ontwikkeld en geïmplementeerd worden.

Contents

List of acronyms

CERT	Computer Emergency Response Teams
CSIRT	Computer Security Incident Response Teams
DNS	Domain Name System
DoH	DNS-over-HTTPS
DoT	DNS-over-TLS
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
IETF	Internet Engineering Task Force
ISP	Internet Service Provider
LEA	Law Enforcement Agencies
NCSC	National Cyber Security Centre
NSA	National Security Agency
RFC	Request for Comments
THTC	Team High Tech Crime
TLS	Transport Layer Security
US	United States

Glossary

Threat intelligence
Information an organization uses to understand the threats that have, will or are currently targeting the organization. Sources can be open source intelligence, social media intelligence, intelligence from the deep and dark web and private or commercial sources.

Passive DNS
Records that contain DNS resolution data for a given location, record, and time period. This historical resolution data set shows domains resolved to an IP address and vice versa. This data set allows for time-based correlation based on domain or IP overlap.

Cobalt Strike
Software for security assessments that replicate the tactics and techniques of an advanced adversary in a network.

(Threat) Actors
A person, group, or entity that attempts to or successfully conducts malicious activities against enterprises.

Encryption
The process of encoding information known as plaintext, into an alternative form known as ciphertext. Only authorized parties can decipher a ciphertext back to plaintext and access the original information.

Chapter 1

Introduction

1.1 Context

In 2013 Edward Snowden shocked the world. In a series of releases he exposed multiple secret projects the United States (US) government was running to spy on people around the world [1]. One of the biggest revelations was the existence of multiple National Security Agency (NSA) surveillance programs. These consisted of two categories: wiretaps that pull data directly from the undersea telecommunication cables, and programs such as PRISM that gather information directly from US service providers such as Google, Facebook and Microsoft [2]. Since these revelations and the more vital role the Internet is playing in our society, the aspects of security and privacy have become more and more of a priority.
In the early days of the Internet, security and privacy were not much of a concern. However, the more the Internet grew, the more people became concerned about their personal information. In 1999 concerns about the safety of personal information and privacy of communications were the main reasons many consumers stayed off the Internet [3]. The main concern was that unauthorised people would find and use the information without consent. At this point the concerns were mainly focused on the companies owning the websites and malicious intruders [3]. This changed in 2013 with the Snowden revelations [1]. These showed that intelligence agencies were gathering large amounts of data from civilians. This sparked a debate on the state of legislation concerning Internet privacy and security [1]. The technicians, however, went back to an old principle: ”The best solution is privacy through technology, not through legislation. The objective must be to bring privacy to the Internet, and bring the Internet to everyday practices [3]”. Over the years, countless efforts have been made to protect data in transit. One of the biggest efforts was the encryption of connections through Transport Layer Security (TLS) [4].
More recently, a vital Internet infrastructure, the Domain Name System (DNS), has also been the focus of privacy improvements. The DNS is responsible for looking up at

which IP address the domain name you are trying to find is, so your computer can connect to that IP [5]. Data can give great insight into the Internet behaviour of individuals such as which social media, banks and other websites are visited. The introduction of encryption through DNS-over-TLS (DoT) in 2016 [6] and DNS-over-HTTPS (DoH) in 2018 [7] by the Internet Engineering Task Force (IETF) have made huge changes to how much, and to whom, DNS data is visible. The implementation of the encryption of the DNS could have a major impact on the research possibilities of incident researchers and law enforcement [8]. This will be further examined in this research.

1.2 Problem statement

The encryption of the DNS has made it so secure it can not be “tracked, spoofed, or blocked [9]”. DNS encryption can take the power away from the Internet Service Provider (ISP) that could exploit your data and complicates censoring by state-run ISPs [9]. However, It has also received a lot of criticism. Although the encryption eradicates snooping on the wire, the information is still visible on the used resolver, such as the one from the state-run ISP. Therefore, mainly in the use of DoH, centralised public resolvers such as Cloudflare and Google are being used [10], [11]. The data on the wire is now encrypted and is sometimes no longer stored within the national borders. This means that other laws and jurisdictions come into play [8], [11].
The value of DNS information in current incident investigations is hard to determine. In an overview from 2012, Wright stated: ”DNS logs, where captured, have value in confirming browsing behaviour versus malware behaviour, identifying system configuration, as well as providing time-line data for investigations” [12]. However, this was before the Internet started becoming more secure with, for example, more of a widespread implementation of TLS for the HyperText Transfer Protocol (HTTP) [13]. This meant that nowadays, less information can be gathered from the HTTP protocol and the value of DNS data has increased.
The possible influence of the encryption of DNS on law enforcement is stated in a recent overview by Europol in 2019 [8]. They state: “The DoH protocol can affect the judicial use of DNS query history in relation to malware investigation, lawful interception, and blocking of IP addresses linked to malware or child sexual exploitation material. It should be noted that if DNS resolution continues to be local and encrypted; Law Enforcement Agencies (LEA)s will continue to access the data through appropriate judiciary requests to the ISP. However, if there is a case of a remote resolver being used for DNS resolution, the data will not be accessible to national authorities”. However, they do not go in depth about the specifics of how DNS data is currently used in investigations or alternative options for still gathering the wanted data after DNS encryption.

Moreover, to the best of our knowledge, no complete and recent information on the use of DNS information in current incident research can be found. Therefore it is also hard to find how incident researchers are adapting to the new implementations of DNS encryption and the availability of data. This will be the main focus of this research.

1.3 Goal

The goal of this research is to create an overview of the use of DNS information by digital incident responders and researchers within the Netherlands. This overview will cover how they gather DNS information, what they use the information for and exploratory questions on encounters with the encryption and expected new criminal techniques. Next, the effects of DNS encryption will be examined by comparing how the different ways of encryption can affect the techniques and information currently gathered. Lastly, this gathered information will be used to look into alternative methods for retrieving the wanted information and possible misuses of the new encryption form.

1.4 Research questions

Main research question

What are the main restrictions encountered by incident researchers due to the encryption of DNS and what are alternatives to gain the wanted information?

Sub research questions

1. What are currently the leading types of DNS encryption in development and being implemented?
2. What are the available investigative options for incident researchers at this moment to gain (unencrypted) DNS information and what is it used for?
3. What are experts experiences and advice when encountering encrypted DNS at the moment and in the future?
4. What potential misuse of encrypted DNS can be expected?

1.5 Research setup

The research is conducted in three stages. The first stage consists of a literature study into the different types of DNS encryption currently being implemented. This information can be found in Chapter 2, together with background information on the DNS. In the second stage, Chapter 3 and 4, this information is used to gather the real-life implications of these encryption types. The second stage consists of interviews with digital incident responders and researchers to gain an overview of how they gather and use DNS information. The interviews also investigate the effect the encryption can have on the current possibilities in digital forensic research, developments in criminality and alternatives after encryption has been implemented fully. This is investigated for law enforcement agencies as well as private companies. The third stage consists of an experiment and investigates possible misuse of the encryption. By testing how easy it is to set up a DNS tunnel for data in- and exfiltration over DoH.

Chapter 2

Background

In this chapter the most important topics of this research are explained. It will start off with a more detailed explanation of the DNS system. Next, a general explanation of encryption in the DNS and two specific types of DNS encryption: DoT and DoH, are explained. Lastly, a specific use of the DNS, DNS tunneling is explained.

2.1 What Is DNS?

Every protocol on the Internet is specified in a so-called Request for Comments (RFC), which are documents maintained by the IETF. In November of 1987 the two main RFCs for the DNS were introduced [5], [14]. In these documents, the concepts, implementations and specifications of the protocol are explained. A simplified version is given below.

The DNS is generally described as the “phone book” of the Internet. This entails that every time you want to connect to a page or service there is a “lookup” that happens first to see where you exactly want to connect to. This lookup, through recursive DNS, consists of a couple of steps and is also visible in Figure 2.1.

The first step is that you, the client, type in an domain in your browser you want to connect to, like *www.example.com*, this is your DNS request. Your browser then connects to the stub resolver built into your client, this is a process that is part of your operating system designed to handle DNS requests. If you have not searched for it recently, it will send the request to a local resolver (1).

The next step is that the resolver, which likely is supplied by your ISP, will try to resolve your request by asking authoritative name servers if they have the IP address connected to your requested domain. This lookup will be from the root down.

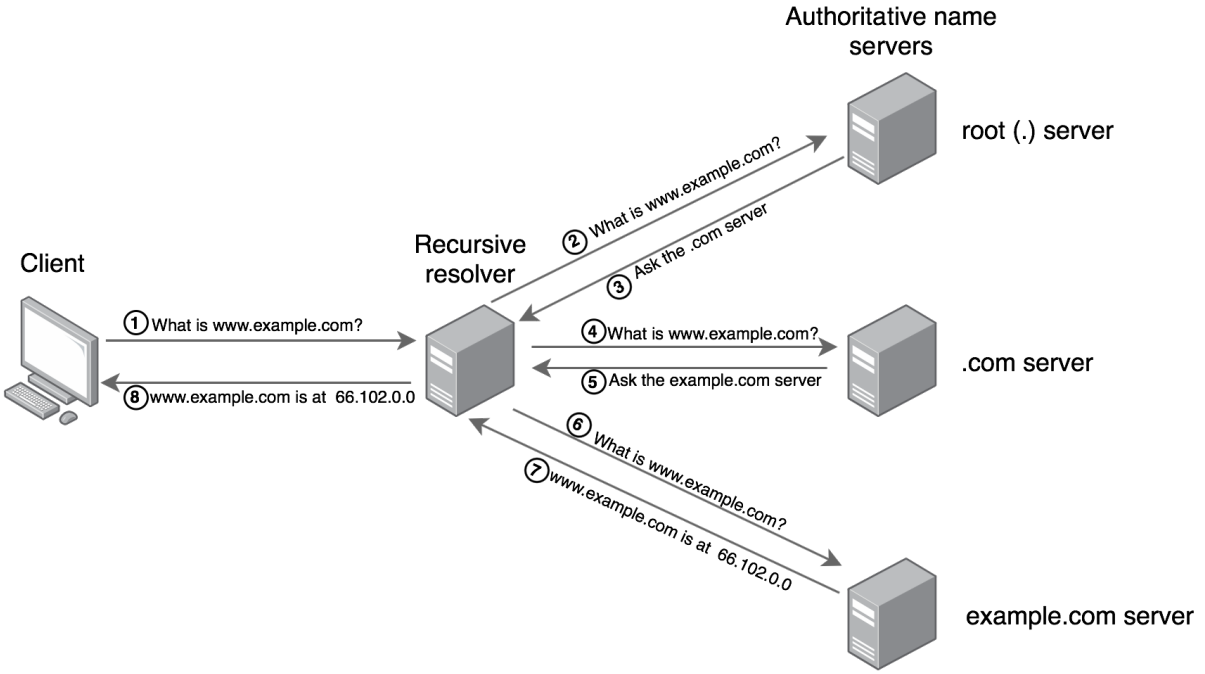


Figure 2.1: DNS recursion overview

If you look at the requested domain *www.example.com* in reality there is an extra (invisible) dot at the end of it (*www.example.com.*). This dot at the end represents the root. This is the top of the DNS hierarchy and signifies a starting point. When a lookup is done the resolver will first request the root server for the information (2). Most likely, the root server will not have the IP address for your specific request, but will know the top-level domain you requested and will send the location for its authoritative name server back to your resolver (3). In this case it would be for the *.com* domain. Again, this server will not have the specific IP address for *example.com* (4) but will know its authoritative name server (5). Eventually the request will be send to the *example.com* authoritative name server (6) that will know the specific IP address related to *www.example.com*. This will be sent to the resolver (7) who will inform the stub resolver and therefore the browser on your computer and the connection can be established (8). The resolver and stub resolver typically keep a database of the most resent requests (cache) so the entire process does not have to be executed for every request every time.

2.2 The encryption of DNS

Since the DNS system was first specified in 1987 it has not changed much. The main design goals consisted of consistency, dealing with size, costs of acquiring data, availability for different protocols, independence and host capabilities [5]. This shows that in that time, privacy was not an aspect considered. In 2015, the IETF made an RFC containing an analysis of the privacy issues associated with the use of DNS by Internet users [15]. This overview showed the best place to “eavesdrop” on information is between the stub resolvers and the recursive resolvers. This is because traffic at this point is not limited by DNS caching and everything is in plain text. To try and change this ease of eavesdropping, two major ways of encrypting DNS have been implemented in recent years. More ways of encrypting DNS are available, but in a comparison of these protocols “DoT and DoH are two leading and mature protocols to secure traditional DNS communications. On top of well-supported and standard protocols, they are both standardised by the IETF and extensively implemented by various DNS software and public resolvers [16]”. Therefore, they are most relevant and the focus of this research.

2.2.1 DNS-over-TLS

The first approach to encrypting part of the DNS data was DNS-over-TLS (DoT) [6]. The concept for this protocol is that first a TLS connection will be established and multiple DNS requests can be sent over this encrypted, established connection. For this connection, only port 853 will be used [6]. The advantages of DoT are that clients and resolvers can exchange encrypted DNS requests and resolvers can be authenticated by verifying TLS certificates. By padding the message no information can be deduced through data analysis [17]. However, by using a designated port, the DNS traffic is still distinguishable from other traffic [16]. Also it has been found that “25% of the DNS-over-TLS service providers use invalid SSL certificates [16]”. This also influences the reliability of this encryption protocol. Most mainstream DNS software supports DoT, which allows its deployment to grow. While a major mobile operating system supports DoT on the client side (Android), the two major operating systems (Windows and MacOS) still do not, which limits the use and growth on the client side [16].

2.2.2 DNS-over-HTTPS

The second and still upcoming way of encrypting DNS is DoH [7]. This, like DoT, uses the TLS protocol but makes the connection through the HyperText Transfer Protocol Secure (HTTPS) application protocol. Therefore, it is allowed to send the requests through port 443, which makes it harder to distinguish from other HTTPS traffic [7]. DoH creates an encrypted connection between client and resolver and stops network operators from blocking DNS encryption by restricting the use of certain ports (like 853 for DoT) [18]. Currently, the blocking of DoH is often still possible because services are offered on a separate server. For example, Google is currently at *https://dns.google.-com/dns-query*, this is a separate server so still blockable. However, if Google changed this to *https://google.com/dns-doh* it would become impossible to block. To block DoH traffic, the only indicator visible is the connection to the google.com server, and by blocking this all other Google services would also be blocked.
Previously, with classic DNS and DoT, applications used the operating system to perform DNS. DoH enables applications to make DNS requests to any chosen server that supports DoH without use of the operating system. This means that application providers can choose the server and its deployment and policy choices [8]. Applications often choose to use public DNS resolvers offering DoH such as Google and Cloudflare. The request sent to these big central resolvers, which are at the moment mostly located in the United States, might result in DNS requests not being resolved within a country's borders. This can bring difficulties for Law Enforcement Agencies (LEAs) of other countries to officially request data or get access through the judicial system [8]. The DoH protocol also has some other difficulties, the main one being: "It can affect the judicial use of DNS query history in relation to malware investigation, lawful interception, and blocking of IP addresses linked to malware or child sexual exploitation material [8]". Though there are large amounts of criticism given on the protocol [8], [10] the implementation is still growing and expected to keep growing for the foreseeable future [16].

2.3 DNS Tunneling

DNS Tunneling is a form of a covert network channel. These covert channels can be used to bypass firewalls, hide data for confidentiality, anonymity or to counter censorship [19].
DNS Tunneling can be done by using software to set up a private network between a client and a server. To make this possible, the receiving side needs to have a domain name server functioning as an authoritative name server. This server also needs to facilitate server side tunneling and decoding programs [20]. In this research the focus will be on tunnels for the use of transferring data inside hostnames (data exfiltration and infiltration). This is most often used in malware and in communication with botnets [21]. By encoding the wanted data with base32 [22], and putting it in front of the ”normal” query such as *base32(data).example.com*, data can be exfiltrated without being detected. In the response of a DNS query, data can also be encoded with base64 for the infiltration of data. In the query response, commands from the command and control server can also be sent to a botnet.

Chapter 3

Expert interviews

In this chapter the full process of the interviews with experts is shown. First, the methodology of the interviews is shown. Next, it shows how the gathered interview data was processed and the experienced limitations. Finally, the results given are divided into subsections to create an overview of the answers sorted by topic.

3.1 Methodology

The approach to determining the experts views on the use of DNS information, the expected effect of the DNS encryption, developments in criminality due to the encryption and alternative ways to gather information after encryption are qualitative interviews in a semi-structured way. Ahead of the interviews, an interviewing procedure and a list of 17 questions were made. These can be found in Appendix A. During the interview, the interviewer went along with the answers given and asked extra questions to go more in depth on interesting and relevant topics. [23].
All interviews were held digitally[1] using different online platforms. When agreed to, the interviews were recorded and in all cases the interviewer wrote along to capture the most important aspects.
Afterwards, the interviews were transcribed in summary and anonymized. These transcriptions can be found in Appendix F. Thereafter, the answers were coded into general categories related to the sub research question (Appendix B) whereupon the results were reformatted into tables that gave a clear overview of the answers given (Appendix C).

[1] Due to the Corona virus in person interviews were not possible

To get a broad view of the field of incident researchers that use the DNS system for different purposes, three different groups were formed. From each group, two to three companies were interviewed. The three groups are:

- **Law enforcement and intelligence services**, these are researchers that work for one of the Dutch ministries and work from a governmental perspective. The interviews were conducted with members of Team High Tech Crime (THTC) of the Dutch national police and the Dutch National Cyber Security Centre (NCSC).
- **In-house security teams**, these are researchers for Computer Emergency Response Teams (CERT) or Computer Security Incident Response Teams (CSIRT) from universities, schools or companies and they keep an eye on the internal safety and incident response. The interviews were conducted with members of the security team of the University of Twente and the network maintenance of Saxion University of Applied Sciences.
- **Companies facilitating a service**, these are researchers working for a company providing security services and consultancy. They are selected by the more well known companies with a focus on Internet safety and incident response. They have also been screened for having a forensic investigative service within the company. The interviews were conducted with staff members of Tesorion, Fox-IT and Northwave.

3.2 Limitations

This research has potential limitations. The most important are stated below:

- Limited sample size
  Because of a limited time of 5 months and the start of the Corona virus pandemic, there was a limit to the availability and amount of experts to be interviewed. This restricts a strong statistical analysis of the results. However, by choosing a broad spectrum of experts, the results will still give a general overview representative to the field.
- Limited comparability of results
  Because the chosen experts and way of interviewing, experts gave a broad set of possible answers to the asked questions. This is a common limitation with semi-structured interviews. Their experience and way of thinking resulted in less comparable answers. However, this gave a broader view of the field and different approaches to the DNS data.

3.3 Results

The questions were made in a way, that from the transcriptions the coded answers could be divided into six main categories. These six categories are:

1. **The general detection system**
   In this category a general overview of the experts systems will be given. This, to show how it is set up and to give a general view of the situation at that company. This will give more insight into the answers given in the other categories.

2. **Data collection**
   In this category the types of DNS data and the way the DNS data is collected will be given. This, to show what the sources are, and which sources might experience impact from the new encryption.

3. **Data processing**
   In this category the way the DNS data is used or processed will be given. This, to show how the system uses the information and responds to it.

4. **Importance and effect**
   In this category the specific usefulness and possible effects of the loss of DNS data is shown.

5. **Encryption**
   In this category everything about the new forms of DNS encryption is shown. It will look at the effect, how much it is observed in the field and the implementation and development expected by the experts.

6. **Criminality and alternatives**
   In this last category the possible new forms of criminality developed because of the DNS encryption will be shown. It will also show alternatives experts could use to still be able to gain the wanted DNS information after the encryption has been implemented.

3.3.1 General detection system

The general buildup of the detection system can be divided in three main topics: the location in the recursion process, whether the organisation has control over the resolver within the network and if the experts are structural and preventative viewers of a system, or incidental viewers because of a incident or crime. Figure 3.1 shows the possible locations for data collection and observation in the recursion process. Most experts stated that the space between the local resolver and name servers (4) is chosen. Some experts say they choose the local resolver (2) or between the client and the resolver (1). A single expert stated to want to be as close as possible to the client (1).

When interviewed about the availability of local resolvers within a network, most experts stated that the companies they work with (or company they work for in case of the UT and Saxion) do have their own local resolver. A single expert stated they do a takeover of a resolver if they want information from it.

Lastly it is good to know that almost all experts are structural and preventative viewers of data in the system. They are hired to keep an eye on the system and help protect it. Only the THTC is most often not a structural viewer in a system. They are most often present in a system to research an incident or a crime. This will also have an effect on the rest of the results given by this expert.

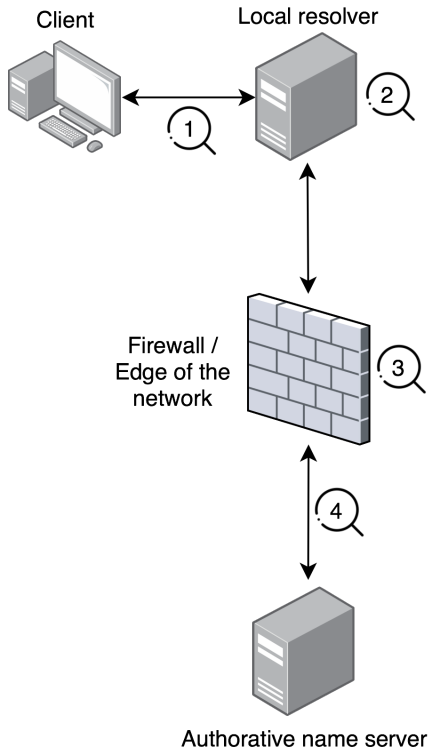


Figure 3.1: Observation locations

3.3.2 Data collection

The data collection is divided into two sections: Which data is being collected? and How and where is data collected?

When interviewed about which data is being collected by experts, almost all experts stated to collect domain names, IP addresses, ports and indicators. These indicators are often collected through threat intelligence. Most experts also stated to collect log files. Some experts collect metadata and flowdata and network traffic from and to the Internet. When interviewed about how and where the data is collected by experts, all use threat intelligence. This is gathered by the experts from different sources: passive

DNS, Cyber threat intelligence, Virustotal, blacklists and gathering own information from, for example, e-mails. Next to threat intelligence, most experts have a device on the edge of a network, at the firewall or at the core switch (3). Most experts also gather data from DNS servers or resolvers (2). A few use a (DNS)tap to gather data.

3.3.3 Data processing

The data processing category is divided into two questions: How is the acquired knowledge used and processed? and What is the response of the system to the acquired knowledge?
When interviewed about the use and processing of the acquired knowledge, one answer was given by all experts: detection. All experts stated they use the knowledge and system for detection, some even for automated real time detection. This detection is used to detect malware, actors, anomalies, domain name generating algorithms and phishing. Next to detection, the acquired knowledge is also used by all experts to correlate marked cases with threat intelligence. Most experts stated they use the information for behavioural analysis and monitoring. Some experts stated they also use the information for research. A few stated they use the information for the tracking of servers. A single expert stated they use the information for dealing with complaints, another stated they use it for blocking requests.
When interviewed about the response of the system to the acquired knowledge, most experts stated hits or alarms are generated at possible suspicious behaviour. Most of them also stated the system tries to identify the original client that generated the hit or alarm. When identified, some experts stated they place the identified client in isolation. A few experts stated they also use the information to check saved data retroactively. Responses of the system to the knowledge stated by a single expert are: proof of malicious activity for prosecution, checking indicators by using log files and the blocking of queries and IP addresses.

3.3.4 Importance and effect

When looking at the importance and effect, the two questions being answered are: How important is DNS information to you? and What if the DNS information was not visible anymore?
When interviewed about the importance of DNS information, some experts stated it is important for threat detection. A few experts stated the information is also available somewhere else, and it is mostly used as an indicator. A single expert stated it is very important with penetration testing software Cobalt Strike [24], and that the importance depends on the suspect and the goal.

When interviewed about what it would mean if DNS information was not visible anymore, most experts stated alternatives are needed to still get the necessary data. Some experts stated threat detection will become harder. A single expert stated that what is not there, can not be found and you need to make sure you are not dependent on the information. Lastly a single expert stated that functionalities of protection will not work anymore.

3.3.5 Encryption

When looking at the new types of encryption being implemented (DoT and DoH), three questions were asked: What will be the effect of the encryption? Have you already seen it in the field? and What will be the implementation and development of the encryption techniques?

Figure 3.2 shows which observation locations will no longer be visible after full implementation of the encryption. When interviewed about the predicted effect of the encryption by experts, it was stated that it depends if applications will chose local encrypted resolvers over open resolvers, or by default will use open resolvers. However, most stated that more effort has to be put in to still making the wanted data visible. A few experts stated that real-time monitoring will become harder. A single expert stated that data will still leak. Another single expert stated that the endpoint will remain visible. Also, the encryption will create more single points of failure and even though you can block DoT at the edge, for DoH and cloud resolving, it is harder.

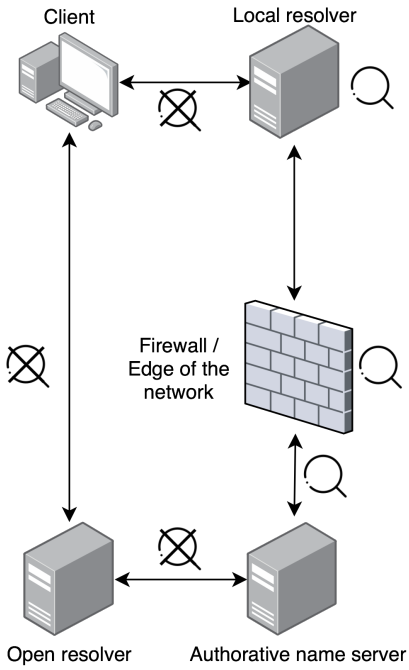


Figure 3.2: Observation locations after encryption

When looking how often it has been seen so far, some experts stated they have rarely seen DoT or DoH being used or have not been paying attention to it yet. Other responses to if it has been seen so far by a single expert state they use the encryption themselves, done research into the use of DoH by malware or blocks all DoT and DoH traffic at the firewall.

Lastly, when interviewed about the implementation and development, some experts stated it will be similar to the implementation of TLS for HTTP. A few experts stated that DoT and DoH will never take over all DNS traffic, and that getting DoT and DoH

supported and working correctly can be difficult. A single expert stated that DoT and DoH will be blocked on networks and jurisdictional adjustments are probably necessary to work properly with DoH. A single expert also stated that less data will be available from the TLS handshake and that network monitoring will only be used for context.

3.3.6 Criminality and alternatives

This last category will look into new kinds of criminal behaviour that experts predict will arise with the new forms of encryption, and alternatives the experts can use to still gather the wanted information.
When interviewed about the new kinds of criminal behaviour that can arise with these forms of encryption, most experts stated that DoH can be used as an in- and exfiltration channel and therefore also used for command and control. A single expert stated criminals can use encryption because it is set as the default mode, or build their own system for protection and security. A single expert also stated that it can be made easier to use by generic tooling and that detecting attacks from your own network will become harder. Lastly, a single expert stated that it can facilitate spoofing attacks by falling back to the NetBios, and cache poisoning and the validation of query and response will become harder to detect.
When interviewed about the alternatives for collecting data, most experts stated that they would change the location for the sensor to endpoints (clients) and the local DNS resolver. At these locations the unencrypted DNS queries will still be available for monitoring. Most experts also stated that some data will leak in the TLS handshake (server name and IP) and in metadata and flowdata. Some experts stated that TLS interception could also be done, but all also stated they prefer not to do this. A few experts stated that offering your own DoT and DoH resolver would also be an alternative. A single expert stated that mapping and predicting criminal behaviour or demanding information at ISPs and open resolvers could also be an alternative for gaining information.

3.3.7 General conclusion

The main results of the interviews is that most experts were not fully aware of the possible impact of the new encryption systems. Most will lose important information for detection from the implementation of the new encryption systems. However, over the interviews the attitude changed and by the end they stated different ways they can circumvent these measures but most have not chosen to implement these yet. Next to that, the impact the measures will have on the available threat intelligence and passive DNS sources is still unknown and could also greatly impact the experts in their work.

Chapter 4

Practical experiment

The goal of the experiment is to show how easy it is to set up a DNS tunnel for data in- and exfiltration over DoH. In this chapter the full process of the experiment is shown. First, a more in depth look at the design is given because this is based upon results from the expert interviews stated in Chapter 3. Next the methodology is shown in which the materials used and the experimental setup are given. Lastly, the gathered results are shown.

4.1 Experiment design

The design of the experiment is based upon the answers given by experts in Chapter 3. They stated that it is highly expected that DoH will be used in the future for in- and exfiltration of data and command and control. However, they also stated criminals most often take the path of least resistance. Therefore, this experiment was designed with existing software and tooling. The goal of the experiment is to test if exfiltration by using a DNS tunnel is possible when using DoH.

4.2 Methodology

The experiment was done in two stages, the first stage was setting up the DNS tunnel and the second stage was setting up a DoH stub resolver by using proxy software. After both stages are set up, the systems can be used together to run a DNS tunnel over DoH.

4.2.1 Stage 1: Setting up the tunnel

The first stage consisted of setting up the DNS tunnel. This needed a client and a server to work. Both the client computer and server were set up in a virtual machine and both were to run on Linux (Ubuntu). Unfortunately, on the client Ubuntu gave some difficulties with running the proxy software. Therefore, the switch to Debian was made. After some research into different software available for DNS tunnelling [25], and searching for others that have used a DNS tunnel over DoH [26], the software Iodine [27] was chosen. This software consists of a client (iodine) and server (iodined) version. For both, version 0.7.0 was used, as can be seen in Appendix D.
The setup of the server, the domain and the software was done by following the accompanied manual on github [27], and the experiences of software developer and blogger David Hamann [28].
The server needed to be an authoritative name server for a domain, which allowed for the client to query the domain and connect to the server. This server was set up to be the authoritative name server for: *anja.dnsjedi.org* on the IP 192.87.172.251. Next, a subdomain was created that routes all the information to the server. This domain is often shorter to allow more data to be trafficked. However, this was not the case in this experiment. In this experiment the domain: *iodine.dnsjedi.org* was used. The specific DNS zone file can be found in Appendix D. To start Iodined on the server, the software was installed and started by entering a password (12345), the new private IP address for the server inside the tunnel (10.10.10.1) and the subdomain to tunnel to (iodine.dnsjedi.org). This can be seen in Figure 4.1. The server side was now up and running and a tunnel could be made from the client side.

```
^Canja@anja:~$ sudo iodined -f 10.10.10.1 iodine.dnsjedi.org
Enter password:
Opened dns0
Setting IP of dns0 to 10.10.10.1
Setting MTU of dns0 to 1130
Opened IPv4 UDP socket
Listening to dns for domain iodine.dnsjedi.org
```

Figure 4.1: Iodined active and running on the server

Next, the client side was set up. This consisted of installing iodine and starting up the software. To start up, the same password and subdomain for the server had to be given. This can be seen in Figure 4.2. Because no resolver was given, iodine automatically uses the local resolver (130.89.0.128), and the client is given a private IP inside the tunnel (10.10.10.2).

```
^Croot@anja:/home/anja1# sudo iodine -f -r -P 12345 iodine.dnsjedi.org
Opened dns0
Opened IPv4 UDP socket
Sending DNS queries for iodine.dnsjedi.org to 130.89.0.128
Autodetecting DNS query type (use -T to override).
Using DNS type NULL queries
Version ok, both using protocol v 0x00000502. You are user #0
Setting IP of dns0 to 10.10.10.2
Setting MTU of dns0 to 1130
Server tunnel IP is 10.10.10.1
Skipping raw mode
Using EDNS0 extension
Switching upstream to codec Base128
Server switched upstream to codec Base128
No alternative downstream codec available, using default (Raw)
Switching to lazy mode for low-latency
Server switched to lazy mode
Autoprobing max downstream fragment size... (skip with -m fragsize)
768 ok.. 1152 ok.. ...1344 not ok.. ...1248 not ok.. ...1200 not ok.. 1176 ok.. 1188
 ok.. will use 1188-2=1186
Setting downstream fragment size to max 1186...
Connection setup complete, transmitting data.
```

Figure 4.2: Iodine active and running on the client

Now, the tunnel has been set up. This can be checked by pinging the server's private IP (10.10.10.1). This is shown in Figure 4.3. If this succeeds, it means direct encoded messages in DNS queries can be sent between client and server. This makes it possible for data to be in- or exfiltrated. The setup for the tunnel is shown in Figure 4.4.

```
root@anja:/home/anja1# ping 10.10.10.1
PING 10.10.10.1 (10.10.10.1) 56(84) bytes of data.
64 bytes from 10.10.10.1: icmp_seq=1 ttl=64 time=3.13 ms
64 bytes from 10.10.10.1: icmp_seq=2 ttl=64 time=2.73 ms
64 bytes from 10.10.10.1: icmp_seq=3 ttl=64 time=3.05 ms
64 bytes from 10.10.10.1: icmp_seq=4 ttl=64 time=2.91 ms
64 bytes from 10.10.10.1: icmp_seq=5 ttl=64 time=2.95 ms
64 bytes from 10.10.10.1: icmp_seq=6 ttl=64 time=2.99 ms
64 bytes from 10.10.10.1: icmp_seq=7 ttl=64 time=2.46 ms
64 bytes from 10.10.10.1: icmp_seq=8 ttl=64 time=3.10 ms
^C
--- 10.10.10.1 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7010ms
rtt min/avg/max/mdev = 2.461/2.916/3.132/0.208 ms
```

Figure 4.3: Pinging the tunnel IP of the server from the client

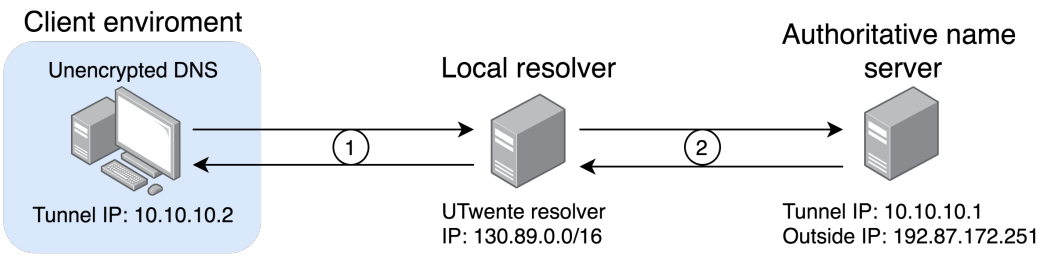


Figure 4.4: The experimental setup of the first stage

4.2.2 Stage 2: Setting up the DoH stub resolver

The second stage of the experiment consisted of tunneling information over DoH so the information between client and resolver is encrypted. To encrypt and send the DNS packets over DoH, a DoH stub resolver is needed to be installed on the client. Though dnscrypt-proxy [29] is software mainly used as a proxy, in this case it was used to act as a stub resolver for DoH, and the manual by Sebastian Neef [26] was used as reference to set it up for tunneling over iodine.
Dnscrypt-proxy was installed and the dnscypt-proxy.toml file was checked and set to a server that supports DoH, in this case Cloudflare. As can be seen in Figure 4.5, the server was set to Cloudflare and no other settings were altered. Next, the resolv.conf file of the client was altered. In this file, the resolver used by the computer needed to be commented out and the IP address at which the DoH stub resolver is located was added. To find the IP address of the stub resolver, a check was performed by doing a status check of the dnscrypt-proxy as seen in Figure 4.6. In this case, the stub resolver is at IP address 127.0.2.1. To switch the system to using this stub resolver proxy, the resolv.conf file was altered as shown in Figure 4.7.

```
root@anja:/home/anja1# cat /etc/dnscrypt-proxy/dnscrypt-proxy.toml
# Empty listen_addresses to use systemd socket activation
listen_addresses = []
server_names = ['cloudflare']

[query_log]
  file = '/var/log/dnscrypt-proxy/query.log'

[nx_log]
  file = '/var/log/dnscrypt-proxy/nx.log'

[sources]
  [sources.'public-resolvers']
  url = 'https://download.dnscrypt.info/resolvers-list/v2/public-resolvers.md'
  cache_file = '/var/cache/dnscrypt-proxy/public-resolvers.md'
  minisign_key = 'RWQf6LRCGA9i53mlYecO4IzT51TGPpvWucNSCh1CBM0QTaLn73Y7GFO3'
  refresh_delay = 72
  prefix = ''
```

Figure 4.5: The settings of the dnscrypt-proxy

```
root@anja:/home/anja1# systemctl status dnscrypt-proxy
● dnscrypt-proxy.service - DNSCrypt client proxy
     Loaded: loaded (/lib/systemd/system/dnscrypt-proxy.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2020-05-27 11:01:20 CEST; 1 weeks 3 days ago
TriggeredBy: ● dnscrypt-proxy.socket
       Docs: https://github.com/DNSCrypt/dnscrypt-proxy/wiki
   Main PID: 481 (dnscrypt-proxy)
      Tasks: 9 (limit: 1132)
     Memory: 8.4M
     CGroup: /system.slice/dnscrypt-proxy.service
             └─481 /usr/sbin/dnscrypt-proxy -config /etc/dnscrypt-proxy/dnscrypt-proxy.toml

May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Network connectivity detected
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Source [/var/cache/dnscrypt-proxy/public-resolvers.md] loaded
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Firefox workaround initialized
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [WARNING] Systemd sockets are untested and unsupported - use at your own risk
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Wiring systemd TCP socket #0, dnscrypt-proxy.socket, 127.0.2.1:53
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Wiring systemd UDP socket #1, dnscrypt-proxy.socket, 127.0.2.1:53
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] [cloudflare] OK (DoH) - rtt: 5ms
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Server with the lowest initial latency: cloudflare (rtt: 5ms)
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] dnscrypt-proxy is ready - live servers: 1
Jun 06 11:07:05 anja dnscrypt-proxy[481]: [2020-06-06 11:07:05] [NOTICE] Server with the lowest initial latency: cloudflare (rtt: 11ms)
```

Figure 4.6: Status of dnscrypt-proxy

```
root@anja:/home/anja1# cat /etc/resolv.conf
# Generated by NetworkManager
#search roaming.utwente.nl
#ameserver 130.89.0.128
#nameserver 130.89.2.4
#nameserver 130.89.2.5
nameserver 127.0.2.1
```

Figure 4.7: The nameserver used by the client

4.2.3 Combining the tunnel and the proxy

After the proxy is set up, the tunnel was run through the proxy DoH stub resolver. This means, that all queries were first sent from the client to the proxy stub resolver, as can be seen in Figure 4.8. This showed that the tunnel was sending the queries to 127.0.2.1, which is the proxy resolver, instead of 130.89.0.128, which is the local resolver.

```
^Croot@anja:/home/anja1# sudo iodine -f -r -P 12345 iodine.dnsjedi.org
Opened dns0
Opened IPv4 UDP socket
Sending DNS queries for iodine.dnsjedi.org to 127.0.2.1
Autodetecting DNS query type (use -T to override).
Using DNS type NULL queries
Version ok, both using protocol v 0x00000502. You are user #0
Setting IP of dns0 to 10.10.10.2
Setting MTU of dns0 to 1130
Server tunnel IP is 10.10.10.1
Skipping raw mode
Using EDNS0 extension
Switching upstream to codec Base128
Server switched upstream to codec Base128
No alternative downstream codec available, using default (Raw)
Switching to lazy mode for low-latency
Server switched to lazy mode
Autoprobing max downstream fragment size... (skip with -m fragsize)
768 ok.. 1152 ok.. ...1344 not ok.. ...1248 not ok.. ...1200 not ok.. ...1176 not ok.. ..1164 ok.. will use 1164-2=1162
Setting downstream fragment size to max 1162...
Connection setup complete, transmitting data.
```

Figure 4.8: The client tunneling over the proxy

The proxy DoH stub resolver encryped the queries for DoH and sent them to an open resolver to be resolved, in this case Cloudflare. Next, the resolver decrypted the DoH packets and sent the queries to the iodined authoritative name server. This setup is showed in Figure 4.9.

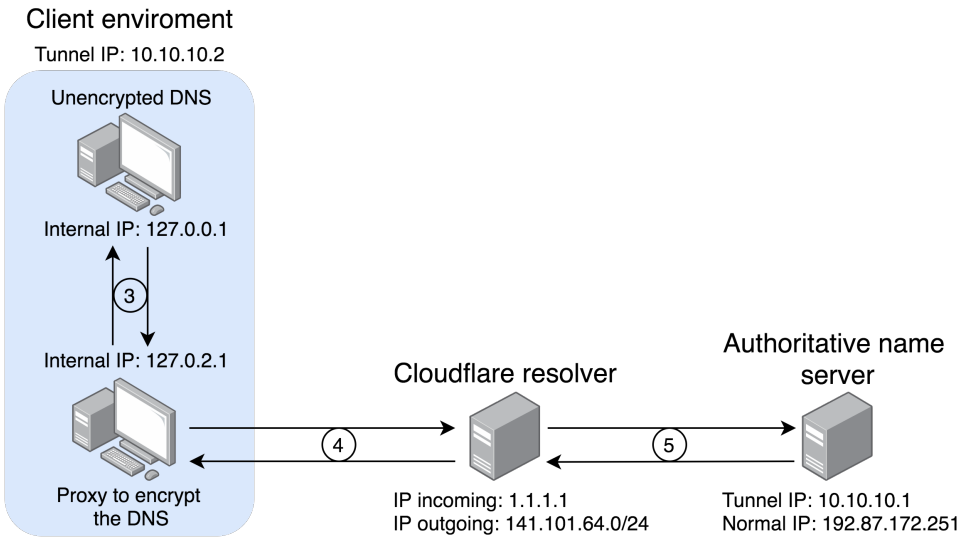


Figure 4.9: The experimental setup of the second stage

Both setups shown above (Figure 4.4 and 4.9) makeup the full experimental setup, as shown in Figure 4.10. The first stage is shown in the top part of the setup and consists of: the client with unencrypted DNS queries, a local resolver and the authoritative name server for the owned domain. The second stage is shown in the bottom part of the setup and consists of: the client with unencrypted DNS queries, a proxy DoH stub resolver to encrypt the DNS query, an open resolver (in this case from cloudflare) and the the authoritative name server for the owned domain.
The numbers in Figure 4.10 show the different points at which measurements of network traffic were done by using Wireshark and tcpdump. This was done to check and prove that everything was working as expected.

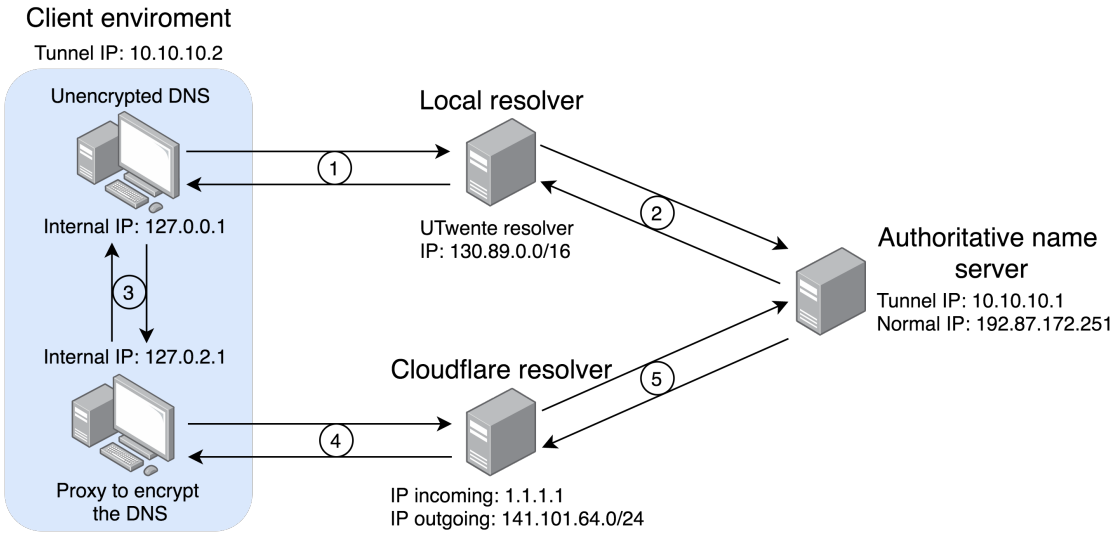


Figure 4.10: The experimental setup of both stages combined

4.3 Results

The results of the individual measuring points in the different stages are shown below. Next to that, screenshots have been made to show all software active and running. These can be found in Appendix D. Lastly, lookups of unknown IP addresses have been done to see who they belong to at that moment, and can be found in Appendix E.

4.3.1 The first stage

The results of the first stage of the experiment can be seen when looking at the measurements done at points 1 and 2. When looking at the measurements done at point 1 (between the client and the local resolver), as can be seen in Figure 4.11, it can be seen that the requests were being sent to and received from a local resolver, in this case: recursor.utsp.utwente.nl.

```
14:26:55.865506 IP anja.39113 > recursor.utsp.utwente.nl.domain: 4553+ [1au] NULL? paaads2a.iodine.dnsjedi.org. (56)
14:26:55.866822 IP recursor.utsp.utwente.nl.domain > anja.39113: 62362 1/0/1 NULL (70)
14:26:56.165588 IP recursor.utsp.utwente.nl.domain > anja.39113: 35174 ServFail 0/0/1 (280)
14:26:56.866893 IP anja.39113 > recursor.utsp.utwente.nl.domain: 12280+ [1au] NULL? paaads2i.iodine.dnsjedi.org. (56)
14:26:56.868641 IP recursor.utsp.utwente.nl.domain > anja.39113: 4553 1/0/1 NULL (70)
14:27:00.870886 IP anja.39113 > recursor.utsp.utwente.nl.domain: 20007+ [1au] NULL? paaads2q.iodine.dnsjedi.org. (56)
14:27:00.873219 IP recursor.utsp.utwente.nl.domain > anja.39113: 12280 1/0/1 NULL (70)
14:27:04.874749 IP anja.39113 > recursor.utsp.utwente.nl.domain: 27734+ [1au] NULL? paaads2y.iodine.dnsjedi.org. (56)
14:27:04.877086 IP recursor.utsp.utwente.nl.domain > anja.39113: 20007 1/0/1 NULL (70)
```

Figure 4.11: Measurements point 1: From client to local resolver

When looking at the measurements done at point 2 (between the local resolver and the server), as can be seen in Figure 4.12, it can be seen that the requests were being received from and sent to the IP address: 130.89.2.156. When looked up, this IP address belongs to the University of Twente, the same as the used resolver on the client side.

```
anja@anja:~$ sudo tcpdump -i ens3 port 53
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens3, link-type EN10MB (Ethernet), capture size 262144 bytes
08:57:15.453475 IP 130.89.2.156.57413 > 192.87.172.251.domain: 57807 NULL? yrbokp.iodine.dnsjedi.org. (43)
08:57:15.454104 IP 192.87.172.251.domain > 130.89.2.156.57413: 57807*- 1/0/0 NULL (103)
08:57:15.455479 IP 130.89.2.156.55531 > 192.87.172.251.domain: 16140 NULL? vaaaakarzka.iodine.dnsjedi.org. (48)
08:57:15.455577 IP 192.87.172.251.domain > 130.89.2.156.55531: 16140*- 1/0/0 NULL (69)
08:57:15.456653 IP 130.89.2.156.64096 > 192.87.172.251.domain: 47119 NULL? labh45z4e02roo2rpd0tfsawrowqdsui.iodine.dnsjedi.org. (69)
08:57:15.458420 IP 192.87.172.251.domain > 130.89.2.156.64096: 47119*- 1/0/0 NULL (110)
08:57:15.495978 IP 130.89.2.156.64808 > 192.87.172.251.domain: 41047 NULL? yrboks.iodine.dnsjedi.org. (43)
08:57:35.828272 IP 130.89.2.156.63694 > 192.87.172.251.domain: 57216 NULL? paaads2i.iodine.dnsjedi.org. (45)
08:57:37.430345 IP 130.89.2.156.63196 > 192.87.172.251.domain: 56891 NULL? paaads2i.iodine.dnsjedi.org. (45)
08:57:38.227649 IP6 2001:67c:2564:a102::1003:1.54911 > 2001:610:1908:ff01:f816:3eff:fe7e:e101.domain: 2884 [1au] NULL? paaads2q.iodine.dnsjedi.org. (68)
08:57:38.228184 IP 130.89.2.156.61240 > 192.87.172.251.domain: 8265 NULL? paaads2q.iodine.dnsjedi.org. (45)
08:57:38.228242 IP 192.87.172.251.domain > 130.89.2.156.52762: 31622*- 1/0/0 NULL (59)
08:57:38.228275 IP 192.87.172.251.domain > 130.89.2.156.63196: 56891*- 1/0/0 NULL (59)
08:57:39.030461 IP 130.89.2.156.63384 > 192.87.172.251.domain: 48852 NULL? paaads2q.iodine.dnsjedi.org. (45)
08:57:39.832343 IP 130.89.2.156.53052 > 192.87.172.251.domain: 62667 NULL? paaads2q.iodine.dnsjedi.org. (45)
08:57:41.433286 IP 130.89.2.156.52164 > 192.87.172.251.domain: 48964 NULL? paaads2q.iodine.dnsjedi.org. (45)
```

Figure 4.12: Measurements point 2: From local resolver to the server

When comparing the measurements from point 1 and 2, it can be seen that the used resolver is in both cases from the University of Twente. What also is noticable is that two tunneling messages can be found in both log files. In Figure 4.11 it can be seen that the client sends queries to paaads2i- and paaads2q.iodine.dnsjedi.org. In Figure 4.12 these queries can also be seen being received by the server.

4.3.2 The second stage

The results of the second stage of the experiment can be seen when looking at the measurements done at points 3, 4 and 5. When looking at the measurements done at point 3 (between the client and proxy stub resolver), as can be seen in Figure 4.13, it can be seen that unencrypted queries and responses were being sent from the local host (127.0.0.1) to the proxy DoH stub resolver (127.0.2.1) and back.

9	−0.000200136	127.0.0.1	127.0.2.1	DNS	98	Standard query 0xb9bf NULL paaaigwq.iodine.dnsjedi.org OPT
10	0.018241150	127.0.2.1	127.0.0.1	DNS	139	Standard query response 0x9b90 NULL paaaigwi.iodine.dnsjedi.org NULL
11	−0.000200136	127.0.0.1	127.0.2.1	DNS	100	Standard query 0xb9bf NULL paaaigwq.iodine.dnsjedi.org OPT
20	0.018241150	127.0.2.1	127.0.0.1	DNS	141	Standard query response 0x9b90 NULL paaaigwi.iodine.dnsjedi.org NULL
21	4.019866015	127.0.0.1	127.0.2.1	DNS	98	Standard query 0xd7ee NULL paaaigwy.iodine.dnsjedi.org OPT
22	4.038052270	127.0.2.1	127.0.0.1	DNS	139	Standard query response 0xb9bf NULL paaaigwq.iodine.dnsjedi.org NULL
23	4.019866015	127.0.0.1	127.0.2.1	DNS	100	Standard query 0xd7ee NULL paaaigwy.iodine.dnsjedi.org OPT
32	4.038052270	127.0.2.1	127.0.0.1	DNS	141	Standard query response 0xb9bf NULL paaaigwq.iodine.dnsjedi.org NULL
49	8.039911392	127.0.0.1	127.0.2.1	DNS	98	Standard query 0xf61d NULL paaaigxa.iodine.dnsjedi.org OPT
50	8.057084670	127.0.2.1	127.0.0.1	DNS	139	Standard query response 0xd7ee NULL paaaigwy.iodine.dnsjedi.org NULL
51	8.039911392	127.0.0.1	127.0.2.1	DNS	100	Standard query 0xf61d NULL paaaigxa.iodine.dnsjedi.org OPT
60	8.057084670	127.0.2.1	127.0.0.1	DNS	141	Standard query response 0xd7ee NULL paaaigwy.iodine.dnsjedi.org NULL

Figure 4.13: Measurements point 3: From the client to the proxy

When looking at the measurements done at point 4 (between the proxy stub resolver and an open resolver), as can be seen in Figure 4.14, it can be seen that the DNS queries were no longer visible. Also, instead of to the utwente.nl resolver, they were being sent to and received from one.one.one.one (1.1.1.1) which is the Cloudflare resolver.

```
anja1@anja:~$ sudo tcpdump -i enp0s3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
11:06:22.627149 IP anja.53788 > one.one.one.one.https: Flags [S], seq 1742699001, win 64240, options [mss 1460,sackOK,TS val 265477800 ecr 0,nop,wscale 7], length 0
11:06:22.629979 IP anja.53790 > one.one.one.one.https: Flags [S], seq 3149324532, win 64240, options [mss 1460,sackOK,TS val 265477803 ecr 0,nop,wscale 7], length 0
11:06:22.630461 IP one.one.one.one.https > anja.53788: Flags [S.], seq 14144001, ack 1742699002, win 65535, options [mss 1460], length 0
11:06:22.630480 IP anja.53788 > one.one.one.one.https: Flags [.], ack 1, win 64240, length 0
11:06:22.633697 IP one.one.one.one.https > anja.53790: Flags [S.], seq 14208001, ack 3149324533, win 65535, options [mss 1460], length 0
11:06:22.633714 IP anja.53790 > one.one.one.one.https: Flags [.], ack 1, win 64240, length 0
11:06:22.656750 IP anja.53790 > one.one.one.one.https: Flags [P.], seq 1:293, ack 1, win 64240, length 292
11:06:22.657015 IP one.one.one.one.https > anja.53790: Flags [.], ack 293, win 65535, length 0
11:06:22.659418 IP anja.53788 > one.one.one.one.https: Flags [P.], seq 1:293, ack 1, win 64240, length 292
11:06:22.659706 IP one.one.one.one.https > anja.53788: Flags [.], ack 293, win 65535, length 0
11:06:22.661915 IP one.one.one.one.https > anja.53790: Flags [P.], seq 1:2832, ack 293, win 65535, length 2831
11:06:22.661931 IP anja.53790 > one.one.one.one.https: Flags [.], ack 2832, win 62480, length 0
11:06:22.665957 IP one.one.one.one.https > anja.53788: Flags [P.], seq 1:2832, ack 293, win 65535, length 2831
11:06:22.665974 IP anja.53788 > one.one.one.one.https: Flags [.], ack 2832, win 62480, length 0
11:06:22.685961 IP anja.53788 > one.one.one.one.https: Flags [P.], seq 293:357, ack 2832, win 63900, length 64
11:06:22.686110 IP one.one.one.one.https > anja.53788: Flags [.], ack 357, win 65535, length 0
11:06:22.689242 IP one.one.one.one.https > anja.53788: Flags [P.], seq 2832:2894, ack 357, win 65535, length 62
```

Figure 4.14: Measurements point 4: From the proxy to an open resolver

When looking at the measurements done at point 5 (between the open resolver and the server), as can be seen in Figure 4.15, it can be seen that the DNS queries were unencrypted and therefore visible. This is expected as the connection between the resolver and server was not encrypted with DoH. What is noticable is that instead of to the utwente.nl resolver, queries were now being sent to and received from 141.101.64.103 which is an IP of a Cloudflare resolver.

```
anja@anja:~$ sudo tcpdump -i ens3 port 53
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens3, link-type EN10MB (Ethernet), capture size 262144 bytes
09:06:22.735249 IP 141.101.64.103.19343 > 192.87.172.251.domain: 47965 [1au] NULL? yrbsxg.iodine.dnsjedi.org. (54)
09:06:22.735560 IP 192.87.172.251.domain > 141.101.64.103.19343: 47965*- 1/0/0 NULL (103)
09:06:22.769163 IP 141.101.64.103.14413 > 192.87.172.251.domain: 54149 [1au] NULL? vaaaakask22.iodine.dnsjedi.org. (59)
09:06:22.769270 IP 192.87.172.251.domain > 141.101.64.103.14413: 54149*- 1/0/0 NULL (69)
09:06:22.783158 IP 141.101.64.103.56431 > 192.87.172.251.domain: 38062 [1au] NULL? lacnlwimrfzximxdu5gr4de45alduv0a.iodine.dnsjedi.org. (80)
09:06:22.783231 IP 192.87.172.251.domain > 141.101.64.103.56431: 38062*- 1/0/0 NULL (110)
09:06:22.865846 IP 141.101.64.103.20363 > 192.87.172.251.domain: 62899 [1au] NULL? yrbsxj.iodine.dnsjedi.org. (54)
09:06:42.758774 IP 141.101.64.103.10977 > 192.87.172.251.domain: 51584 [1au] NULL? paaaeway.iodine.dnsjedi.org. (56)
09:06:43.009254 IP 141.101.64.103.10977 > 192.87.172.251.domain: 51584 [1au] NULL? paaaeway.iodine.dnsjedi.org. (56)
09:06:43.259175 IP 141.101.64.103.10977 > 192.87.172.251.domain: 51584 [1au] NULL? paaaeway.iodine.dnsjedi.org. (56)
09:06:43.508804 IP 141.101.64.103.10977 > 192.87.172.251.domain: 51584 [1au] NULL? paaaeway.iodine.dnsjedi.org. (56)
09:06:43.758734 IP 141.101.64.103.10977 > 192.87.172.251.domain: 51584 [1au] NULL? paaaeway.iodine.dnsjedi.org. (56)
09:06:43.960643 IP 141.101.64.103.22312 > 192.87.172.251.domain: 32044 [1au] NULL? paaaeway.iodine.dnsjedi.org. (56)
09:06:44.210568 IP 141.101.64.103.22312 > 192.87.172.251.domain: 32044 [1au] NULL? paaaeway.iodine.dnsjedi.org. (56)
```

Figure 4.15: Measurements point 5: From an open resolver to the server

4.3.3 General conclusion

The main result of this experiment is that it is possible to set up a DNS tunnel when using DoH to encrypt the traffic between the client and the resolver in about two afternoons.

Chapter 5

Discussion

When analysing the results gathered in the expert interviews, a few trends are visible. The University of Twente and Saxion University of Applied Sciences both use a protection and detection system made by Tesorion for their local network. Therefore, it is expected that answers given by these three experts are similar. However, often answers were not the same or not given by one of the three experts. This shows how differently questions were interpreted by the different experts (as can be seen in Appendix C). In addition, even though the NCSC was grouped as "Law enforcement and intelligence services", the answers given were more similar to the "companies facilitating a service" group. This is understandable when taking the premise and system of the company into account, however, this was not anticipated.
The interview result section (3.3) shows the main restrictions that the experts expect to encounter from the encryption of DNS. However, this does not cover all possible restrictions, and is limited to the restrictions the experts control. The possible restrictions on the threat intelligence and passive DNS sources are rarely mentioned, though they form a major source of DNS information for most experts. Therefore, it can be questioned to which degree the interviewed experts are aware of the full effect this encryption might have.
When looking at the criticism given on the new forms of encryption in Chapter 1 and 2, there are surprising differences. Europol (2019) [8] states that: "The implementation of encryption could have a major impact on the research possibilities of incident researchers and law enforcement" and that "If there is a case of a remote resolver being used for DNS resolution, the data will not be accessible to national authorities." However, the qualitative interviews showed that organisations in a network because of an incident or crime rarely gather DNS information on the wire. They prefer takeovers of resolvers and servers or claiming information from a service like an ISP. In the interview they stated that this can be done nationally or internationally and borders hold no restrictions at the moment because there are good relations with foreign authorities. Therefore, it is estimated this encryption will have little effect on them.

Chapter 6

Conclusion

This research aimed to show "THE MAIN RESTRICTIONS ENCOUNTERED BY INCIDENT RESEARCHERS DUE TO THE ENCRYPTION OF DNS AND ALTERNATIVES TO GAIN THE WANTED INFORMATION". This is investigated on four aspects: The current leading types of encryption, the available investigative options to gather DNS information, the experiences and advice when encountering encrypted DNS and the potential misuse of encrypted DNS. When looking at the current leading types of encryption, it can be concluded that DoT and DoH are being implemented more and are growing in use. Although DoH got a lot of criticism, implementation is growing and is expected to keep growing for the foreseeable future.

The qualitative interviews showed that organisations that are in a network because of an incident or a crime, rarely gather DNS information on the wire. They prefer takeovers of resolvers and servers or claiming information at an ISP. This is very interesting when looking at the available investigative options to gather DNS information and the experiences and advice when encountering encrypted DNS. Because, in contrary to what is stated by Europol (2019) [8], the interviews show that the encryption of DNS will have little effect on their investigations. The companies that structurally and preventative viewers do gather information over the wire. These experts will also experience more effect from the encryption of the DNS. Companies that use data gathered between client and resolver will experience the most effect from this encryption. Real-time detection and trace-backs to the original client will become increasingly difficult. Generally, more effort will be needed to get the wanted DNS information. The best alternatives stated by the experts are: changing the location of the sensors to endpoint detection and the resolver itself, or offering DoT and DoH within the local network. This will ensure all data will still be visible on the resolver and will stay within the local system if applications prefer local resolvers over open resolvers.

When looking at the potential misuse of encrypted DNS through the experiment performed, it can be concluded that it is trivial to use the new encryption for criminal activity like the in- and exfiltration of data through a DNS tunnel.

Chapter 7

Recommendations

Further research should look into the effect DoT and DoH will have on the available threat intelligence and other passive DNS sources. Threat intelligence is used by all experts and limiting this information could potentially have a big effect on their work.
Further research should also look into the effect of other security aspects after the encryption, that have not specifically been looked into in this research. For example, the blocking of child pornography or other illegal websites.
In addition, more research should be focused on limiting DNS tunneling, normally and over encryption. During the experiment it was found to be an easy way to gather information from clients or to control a botnet. Detection should be built to create a barrier and stop the ease with which DNS tunneling is possible at the moment. Criminals prefer to take the path of least resistance, and currently, tunneling is without resistance.
My final recommendation is that more software for local DoT and DoH resolvers becomes available and is implemented. This is because, at the moment, encrypted traffic has to be sent to open resolvers because it is not offered locally. When this encryption becomes offered at local resolvers, there is a higher chance DNS data will stay within the observable systems and is secure on the wire. This will allow security teams to observe and secure the devices and environment like before, and offer more security on the wire.

Bibliography

[1] S. Landau, “Making sense from snowden: What’s significant in the nsa surveillance revelations,” *IEEE Security & Privacy*, vol. 11, no. 4, pp. 54–63, 2013.

[2] T. Sottek and J. Kopfstein, “Everything you need to know about PRISM,” *The Verge*, pp. 1 – 8, 2013. [Online]. Available: https://www.theverge.com/2013/7/17/4517480/nsa-spying-prism-surveillance-cheat-sheet

[3] M. Attaran and I. VanLaar, “Privacy and security on the internet: How to secure your personal information and company data,” *Information Management and Computer Security*, vol. 7, no. 5, pp. 241–246, 1999.

[4] E. Rescorla and Mozilla, “The Transport Layer Security (TLS) Protocol Version 1.3,” *IETF RFC8446*, pp. 1–160, 2018. [Online]. Available: https://tools.ietf.org/html/rfc8446

[5] P. Mockapetris and ISI, “Domain names - concepts and facilities,” *IETF RFC1034*, pp. 1 – 55, 1987. [Online]. Available: https://tools.ietf.org/html/rfc1034

[6] Z. Hu, L. Zhu, J. Heidemann, A. Mankin, D. Wessels, P. Hoffman, ICANN, USC/ISI, Independent, and Verisign Labs, “Specification for DNS over transport layer security (TLS),” *IETF RFC7858*, 2016.

[7] P. Hoffman, P. McManus, ICANN, and Mozilla, “DNS Queries over HTTPS (DoH),” *IETF RFC8484*, 2018.

[8] Europol and Eurojust Public Information, “Second report of the observatory function on encryption,” Europol and Eurojust Public Information, Tech. Rep., 2019.

[9] E. Falcon and Electronic Frontier Foundation (EFF), “DNS over HTTPS Will Give You Back Privacy that Big ISPs Fought to Take Away,” *EFF*, no. October, 2019. [Online]. Available: https://www.eff.org/deeplinks/2019/10/dns-over-https-will-give-you-back-privacy-congress-big-isp-backing-took-away

[10] L. H. Newman, “A Controversial Plan to Encrypt More of the Internet — WIRED,” *Wired*, vol. 15, pp. 1–8, 2019. [Online]. Available: https://www.wired.com/story/dns-over-https-encrypted-web/

[11] V. Bertola, “DNS-over-HTTPS Public Policy Briefing,” *Open-Xchange*, no. November, 2018.

[12] N. Wright, “DNS in Computer Forensics,” *Journal of Digital Forensics, Security and Law*, vol. 7, no. 2, 2012. [Online]. Available: https://commons.erau.edu/jdfsl/vol7/iss2/2

[13] E. Rescorla and RTFM Inc., “HTTP over TLS,” *IETF RFC2818*, vol. May, pp. 1–29, 2000. [Online]. Available: https://tools.ietf.org/html/rfc2818

[14] P. Mockapetris and ISI, “Domain names - implementation and specification,” *IETF RFC1035*, pp. 1–55, 1987. [Online]. Available: http://tools.ietf.org/html/rfc1035

[15] S. Bortzmeyer and AFNIC, “DNS Privacy Conciderations,” *IETF RFC7626*, pp. 1 – 17, 2015. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7626.html

[16] C. Lu, B. Liu, Z. Li, S. Hao, H. Duan, M. Zhang, C. Leng, Y. Liu, Z. Zhang, and J. Wu, “An End-to-End, Large-Scale Measurement of DNS-over-Encryption: How Far Have We Come?” in *Proceedings of the Internet Measurement Conference*, 2019, pp. 22–35.

[17] S. Siby, M. Juarez, N. Vallina-Rodriguez, and C. Troncoso, “DNS Privacy not so private: the traffic analysis perspective,” 2018.

[18] O. van der Toorn, M. Müller, S. Dickinson, C. Hesselman, A. Sperotto, and R. Van Rijswijk-Deij, “Addressing the Challenges of Modern DNS A Comprehensive Tutorial,” 2019.

[19] T. Van Leijenhorst, K. W. Chin, and D. Lowe, “On the viability and performance of DNS tunneling,” *5th International Conference on Information Technology and Applications, ICITA 2008*, pp. 560–566, 2008.

[20] D. Tatang, F. Quinkert, N. Dolecki, and T. Holz, “A Study of Newly Observed Hostnames and DNS Tunneling in the Wild,” pp. 1 –16, 2019. [Online]. Available: http://arxiv.org/abs/1902.08454

[21] A. Hinchliffe, “DNS Tunneling: how DNS can be (ab)used by malicious actors,” pp. 1–11, 2019. [Online]. Available: https://unit42.paloaltonetworks.com/dns-tunneling-how-dns-can-be-abused-by-malicious-actors/

[22] S. Josefsson and SJD, “The Base16, Base32, and Base64 Data Encodings,” *IETF RFC4648*, pp. 1–18, 2006. [Online]. Available: https://tools.ietf.org/html/rfc4648

[23] W. C. Adams, “Conducting Semi-Structured Interviews,” in *Handbook of Practical Program Evaluation: Fourth Edition*, 2015, no. August 2015, pp. 492–505.

[24] “Cobalt Strike, advanced threat tactics for penetration testers,” https://www.cobaltstrike.com, accessed: 05-06-2020.

[25] B. Voss, “Information Security Reading Room,” *Sans Institute (Information Sceurity Reading Room)*, vol. 3, no. 2, pp. 2–6, 2019.

[26] S. Neef, “Performance of Iodine over DNS-over-HTTPS,” https://0day.work/performance-of-iodine-over-dns-over-https/, accessed: 05-06-2020.

[27] “Official git repo for iodine dns tunnel,” https://github.com/yarrick/iodine, accessed: 05-06-2020.

[28] D. Hamann, “Tunneling network traffic over DNS with Iodine and a SSH SOCKS proxy,” https://davidhamann.de/2019/05/12/tunnel-traffic-over-dns-ssh, accessed: 05-06-2020.

[29] “dnscrypt-proxy 2 - A flexible DNS proxy, with support for encrypted DNS protocols,” https://github.com/DNSCrypt/dnscrypt-proxy, accessed: 05-06-2020.

[30] “Interviews afnemen? Gebruik de LSD-techniek,” https://www.studiemeesters.nl/studietips/interviews-afnemen-lsd-techniek/, accessed: 05-06-2020.

Appendix A

Interview questions

1. Wat is uw (naam), vooropleiding, ervaring, bedrijf, rol van het bedrijf?
2. Wat is uw functie binnen ”het bedrijf”?
3. Wat houd deze functie in? wat zijn uw voornaamste werkzaamheden?
4. Hoeveel mensen met een vergelijkbare functie zijn er binnen uw bedrijf?
5. Waar in uw functie komt u voornamelijk DNS informatie tegen?
6. Op welke manier verkrijgt u deze DNS informatie? Zijn er verschillende manieren bij u bekend? Welke gebruikt u en waarom deze?
7. Waar in de communicatiestroom/het recursieproces verkrijgt u op dit moment de meeste informatie? Waarom daar? (afbeelding voor hulp)
8. Welk mogelijke toepassingen zouden er zijn voor deze infromatie?
9. Wat is er met deze informatie eventueel te bewijzen?
10. Hoe belangrijk is DNS als informatiebron in uw werkproces?
11. Wat zou het effect zijn als deze informatie totaal onbereikbaar/onbruikbaar zou worden?
12. Bent u bekend met de nieuwe manieren die geimplementeerd worden om DNS informatie te versleutelen zoals DoH en DoT?
13. Bent u deze vormen van encryptie al eens in het werk tegengekomen? Wat was het effect dat u merkte van deze versleutelingen?
14. Welke impact verwacht u verder dat deze vesleutelingen op uw werk zullen hebben?
15. Welke ontwikkelingen in criminaliteit zal deze encryptie met zich meebregen?

16. Weet u al van alternatieven die u in kan zetten om de informatie toch te verkrijgen na de versleuteling? Welke zijn dit? of heeft u zelf een idee hoe u dit zou aanpakken?

17. Heeft u nog andere zaken met betrekking tot het onderwerp waarvan u denkt dat deze van belang zijn? of heeft u nog suggesties voor aanvullend vragen?

A.1 Conducting the interview

1. Afspreken opnames maken en meeschrijven
2. Opnames starten
3. Afspreken annonimiseren en inzage uitwerking
4. Kort voorstellen: wie ben ik en studie/afstuderen
5. Welkom heten, korte context van het interview en onderzoek
6. Het stellen van de vragen, hierbij **luisteren, samenvatten en doorvragen**
7. Controle van beantwoording van alle vragen
8. Controle of geinterviewde nog vragen en/of verdere opmerkingen heeft
9. Opname stoppen
10. Vragen hoe het interview ervaren. suggesties/tips voor volgende keer
11. Bedanken voor het meehelpen

Source: [30]

Appendix B

General coding overview

	Hoe zit het systeem in elkaar?				Dataverzameling		Dataverwerking/-gebruik		Belang en effect		Encryptie			Overig	
	Sensor-/netwerkopbouw	Plek in recursieproces	Eigen resolver/daarop controle?	Gewenste meekijkers?	Welke data wordt er verzameld	Hoe en waar wordt data verzameld	Inzetten/verwerken van kennis	Reactie van "systeem" op kennis	Hoe belangrijk is de DNS-informatie?	Wat als niet meer zichtbaar?	Effect	Voorkomen/ tegenkomen	Implementatie en ontwikkeling	Criminaliteit	Alternatieven
THTC		Zo dicht mogelijk bij de client	Recursive resolvers nemen we eerder over, hoeven we niet "af te luisteren"	Nee	Metadata/Flowdata	Taps	Query's van een client voor een beter beeld van de gebruiker	Er moeten duidelijke aanwijzingen zijn, objectief verzameld die samen in alle waarschijnlijkheid iets aantonen	Belang is afhankelijk van welke verdachte en om welk doel het gaat.	Afhankelijk van gebruiker en doel	Er zal altijd nog iets van data lekken.	Nog vrij weinig gezien, minder dan verwacht bij huisaansluiting en	DoH en DoT zullen nooit volledig alles overnemen.	Criminelen zullen het niet snel vertrouwen	Data zal worden gevorderd bij ISP's en open resolvers. Wat er niet is kun je niet vorderen
					Domeinnamen	Cyber threat intelligence (minimaal, niet als bewijs)	Services die de domeinnaam gebruiken als merk			Data dat er niet is kan niet gevorderd worden.	Eindpunt zal alsnog duidelijk blijven	Meer encryptie dan dat mogelijk door VPN	Veel controle op uitgaand DNS-verkeer en zicht verliezen is onwenselijk	Of geen kennis ervan en dan de default instellingen gebruiken	DoH lekt altijd nog wel iets van data, eindpunt zal altijd nog duidelijk blijven
					IP-adressen en poorten	Passive DNS	Services die de domeinnaam gebruiken als afscherming				Afscherming is een belangrijk aspect voor criminelen		DoH zal eerder niet toegestaan worden op een netwerk. (blokkeren?)	Of wel kennis ervan en misschien eigen systeem bouwen	
						Data opvragen bij ISP's en open resolvers	Bij tussenpersonen achterhalen waar servers draaien						Er zijn juridische aanpassingen nodig voor DoH, andere retentietijd of gegevens bewaren	Afscherming is een belangrijk aspect	
													Mate waarin criminelen open resolvers zullen vertrouwen	DoH kan gebruikt worden als exfiltratiekanaal (ook in combinatie met DNS-tunnels)	
														Kosten-batenafweging	
NCSC	Netwerksensoren binnen de overheid voor een nationaal detectienetwerk	Meest tussen resolver en nameserver	Afhankelijk per deelnemer	Ja	Indicators of compromise	Threat intelligence (publiekelijk nieuws, publieke tooltjes)	Detectie van malware	Hits van mogelijk verdacht gedrag	Redelijk belangrijk	Detectie van dreigingen wordt moeilijker	Meer moeite gedaan worden om data inzichtelijk te houden	DoH komen we beperkt tegen (geen hinder of impact)	Zelfde probleem als met TLS, op netwerkniveau versleuteld	DNS tunnelling wordt gebruikt door sommige actoren	Informatie te halen uit TLS handshake bij DoT en DoH
	Indicatoren gaan in het systeem	Tussen resolvers intern	De meeste grote organisaties wel		Bedreigingsinformatie	Intell providers (commerciële bronnen)	Detectie van (statelijke) actoren	Opgeslagen data wordt met terugwerkende kracht gecheckt	Belangrijk voor detectie van dreigingen	Zie alternatieven	DoH verkeer van BYOD volledig missen	DoT heel beperkt (geen hinder of impact)		Command and control over DoH	Andere locatie voor de sensor (endpoint detectie)
	Daar komen "hits" uit en die moeten verder onderzocht worden	Tussen client en resolver			AI het netwerkverkeer van en naar het internet van deelnemers	Sensoren in het netwerk op punt van en naar het internet (edge)	Landelijk beeld voor verbetering		Informatie (hostnames, IP-adressen) ook anders te verkrijgen	Zie encryptie		Voornamelijk nog UDP		Zodra er een publiek tooltje van is, zal die worden gebruikt	Koppelen aan DNS resolver van de organisatie
	Sensoren kijken naar al het netwerkverkeer van en naar het internet				IP-adressen en domeinnamen geassocieerd met malware, campagnes en actoren	Sensoren in sommige gevallen op meerdere plekken in het netwerk									TLS-interceptie (middle box)
	Niet zozeer intern verkeer, te veel en niet nuttig genoeg					Contact met CERT over de hele wereld (CTI-data)									
	SIEM om oorzakende client te achterhalen														

	Hoe zit het systeem in elkaar?				Dataverzameling		Dataverwerking/-gebruik		Belang en effect		Encryptie			Overig	
	Sensor-/netwerkopbouw	Plek in recursieproces	Eigen resolver/daarop controle?	Gewenste meekijkers?	Welke data wordt er verzameld	Hoe en waar wordt data verzameld	Inzetten/verwerken van kennis	Reactie van "systeem" op kennis	Hoe belangrijk is de DNS-informatie?	Wat als niet meer zichtbaar?	Effect	Voorkomen/ tegenkomen	Implementatie en ontwikkeling	Criminaliteit	Alternatieven
FOX-IT	Op de sensor alleen active directory als DNS resolver zichtbaar, niet individuele clients	Na de resolver	Active directory (DNS resolver) zichtbaar, niet clients	Ja	Domeinnamen en IP-adressen	Zelden is er DNS logging aanwezig (veel verkeer)	Reverse DNS voor achterhalen of er VPN in het spel is, en welke		Veel incidenten hebben niet per se DNS-componenten		Meer blind worden	Nog niet op geiet of expliciet tegengekomen	Vergelijkbaar met HTTP naar HTTPS	Spoofing aanvallen door Windows terug te laten vallen op NetBios resolving	Nee, implementatie zal niet zo snel gaan, voorlopig niet nodig
	Sensor niet in het netwerk omdat intern verkeer niet interessant of relevant is				Log bestanden	Sensor op de edge van een netwerk (voor het internet op)	Correlatie van opvallende dingen met passive DNS bronnen		Vaak ook niet het zwaartepunt bij DNS		Realtime monitoren via het netwerk wordt moeilijker	Zelf een beetje gebruiken voor anonimiteit	Minder data uit TLS handshake, ziet alleen DoT/DoH server	DNS tunnelling lastiger omdat vaak één DNS verzoek en daarna geen connectie	Logging op de servers zelf
	Intern verkeer vergt meer capaciteit en is ingewikkelder				Beschikbare data (intell), welke name servers worden gebruikt	Passive DNS providers (virustotal)	APT-campagnes die twee domeinnamen naar zelfde IP-adres brengen is verdacht		Het werkt als indicator en daarvoor is het handig		Meer single points of failure die veel data hebben				Logging op de interne recursor
	Client alsnog te achterhalen a.d.h.v. Connectielogs met het geresolvede IP-adres						anomaly detection op basis van DNS logs		Wel zwaartepunt bij pen testing framework Cobalt Strike (gebruikt DNS-tunnels)						Endpoints
	Client niet te achterhalen in geval van DNS-tunnel (eenrichtingsverkeer)						Generieke detectie voor Domain name generating algorithms								Gedrag van criminelen in kaart brengen en voorspellen
							Ingezet voor onderzoeken								
							Monitoren van typosquats, forward lookups								

	Hoe zit het systeem in elkaar?				Dataverzameling		Dataverwerking/-gebruik		Belang en effect		Encryptie			Overig	
	Sensor-/netwerkopbouw	Plek in recursieproces	Eigen resolver/daarop controle?	Gewenste meekijkers?	Welke data wordt er verzameld	Hoe en waar wordt data verzameld	Inzetten/verwerken van kennis	Reactie van "systeem" op kennis	Hoe belangrijk is de DNS-informatie?	Wat als niet meer zichtbaar?	Effect	Voorkomen/ tegenkomen	Implementatie en ontwikkeling	Criminaliteit	Alternatieven
Northwave	Bij implementatie wordt een netwerksensor opgehangen	Tussen client en resolver (voornamelijk)	Meeste gevallen een interne resolver	Ja	Kopie van het netwerkverkeer	Netwerksensor bij de core switch naar het internet	Signature-based detection om netwerkverkeer te filteren dat op kwaadaardige dingen duidt	Alarmen		Dan ben je ongeveer alles kwijt	Het feit dat je het uit het netwerkverkeer weghaalt, legt een extra horde neer en maakt dat je netwerkmonitoring een stuk lastiger wordt	Nog niet tegengekomen	Netwerkmonitoring zal veranderen naar relevant voor context, maar niet meer als primaire bron van informatie	Gemakkelijker om command en control verkeer over DNS te laten lopen (DNS tunnelling)	Op de resolver extra tools die specifiek de data over query's teruggeven
	Deze maakt een kopie van het netwerkverkeer	Tussen resolver en het internet	Geen controle daarop omdat niet al het verkeer per se via de resolver gaat		Metadata (hoelang duurde de connectie, welk protocol, bij HTTP en DNS ook extra informatie zoals domein of URL's)	Threat intelligence (passive DNS en data van virusscanners)	Aan de hand van packet inspection alarmen genereren	Na het detecteren van een alarm op een host ook netwerkverkeer van deze host achterhalen		Dan heb je alleen nog de metadata (waar het vandaan komt en waar het heen gaat)		Wil niet zeggen dat het er niet is	Vergelijkbaar met HTTP naar HTTPS voor impact op security	Gemakkelijker ongedetecteerd data in queryveld te stoppen	Meer endpoint monitoring
	Deze kopie is in meeste gevallen van de switch naar het internet toe (core switch)		Normale netwerkverkeer via de switch, dus daar meer data te verzamelen		Logbestanden		Indicatoren worden gecontroleerd aan de hand van de logs	Indicatoren worden gecontroleerd aan de hand van de logs		Wel nog zichtbaar wanneer de query naar een verdacht domein zou gaan vanuit de resolver		Wordt niet 24/7 naar netwerkverkeer gekeken		DNS cash poisoning niet meer kunnen detecteren	Metadata die dan wel nog zichtbaar is (zie wat als niet meer zichtbaar)
	Metadata wordt bij een 1 GB/s verbinding verzameld en opgeslagen				Indicatoren		Indicatoren controleren aan de hand van de geschiedenis	Indicatoren controleren aan de hand van de geschiedenis		Verbinding met een IP dat bij een domein hoort is dat ook nog zichtbaar in de metadata		Wordt ook niet op gemonitord		Moeilijker om query en response te checken of validiteit (kan wel nog als host)	
	Bij een grotere verbinding wordt er meer gefilterd									Moeilijk te achterhalen waar de originele DNS-query vandaan komt		Niet nodig om te monitoren, niet per definitie kwaadaardig		Niet bekend in hoeverre dat gemakkelijker wordt	
	Extra sensor (naast die op de switch) waarnaar het verkeer gemirrord wordt en hierop word de detectie gedaan													DNS tunnelling wordt wel gezien, maar is next level aanvalsmethode	
	Bij een match gaat er een vlaggetje omhoog													Steeds vaker state-actors kunnen zijn die dat gebruiken	
	Indicatoren worden verzameld en hier wordt detectie op gebouwd														
	Intern en extern verkeer is zichtbaar														

	Hoe zit het systeem in elkaar?				Dataverzameling		Dataverwerking/-gebruik		Belang en effect		Encryptie			Overig	
	Sensor-/netwerkopbouw	Plek in recursieproces	Eigen resolver/daarop controle?	Gewenste meekijkers?	Welke data wordt er verzameld	Hoe en waar wordt data verzameld	Inzetten/verwerken van kennis	Reactie van "systeem" op kennis	Hoe belangrijk is de DNS-informatie?	Wat als niet meer zichtbaar?	Effect	Voorkomen/ tegenkomen	Implementatie en ontwikkeling	Criminaliteit	Alternatieven
Tesorion	DNS detector waarbij DNS-data wordt vergeleken met verzamelde threat intell	Tussen client en resolver (voornamelijk)	Meeste gevallen een interne resolver van de klant	Ja	Complete kopie van het netwerkverkeer	DNS-server	Geautomatiseerd realtime detectie op netwerken	Ingrijpen op netwerken en clients isoleren	DNS vaak niet de enige bron van informatie	Zorgen dat je er niet afhankelijk van bent	Je mist dingen als je alleen DNS doet	AI malwareonderzoek gedaan waarbij gebruik gemaakt werd van DoH	Duurt nog lang voordat alles overgestapt is naar DoT of DoH	Minder opvallen wat je aan het doen was, weinig verschil zitten in de vorm	Beeld meer verleggen naar endpoints toe
	Een van de detectiemethoden	Bij de server (zie dataverzameling)			Metadata	Edge van het netwerk (doorgestuurd vanuit firewall en coreswitch)	Tracken en isoleren van malware	Achterhalen welke client het was	Je mist dingen als je alleen DNS doet	Erg laagdrempelig om op te vangen en te analyseren	DoT is bij de edge te blokkeren	Voor zover bekend nog niets gemist	Veel browsers nog standaard geen DoH	DNS over DoH kan exfiltratie over externe servers opleveren	Meer gebruik maken van flowdata
	O.a. Een complete kopie van het netwerkverkeer				Domeinnamen	DNS-tap	Tracken van phishing		Belangrijk voor de detectie van dreigingen	Ook malware die geen DNS gebruikt	DoH is bij de edge bijna niet te blokkeren	Nog vrij weinig tegengekomen	Op veel OS staat het niet standaard aan	Malware exfiltratie nu vaak via andere manieren dan DNS	Deep packet inspecting mogelijk bij gebruik van TLS (vaak ongewenst)
	DNS-tap en syslog					Syslog	Telemetrische metingen aan de hand van bloom filters			Andere tools nodig	Bij veel cloudproviders is het blokkeren van DoH niet mogelijk zonder risico te lopen om ook reguliер webverkeer te blokkeren		Malware waarschijnlijk ook later pas volgen aangezien het over OS werkt		TLS handshake gebruiken voor server naam en IP (hier ook al onderzoek naar gedaan)
	Geautomatiseerd realtime ingrijpen op netwerken en clients isoleren					Threat intell	Anomaly detection				Er moet meer moeite gedaan worden om data inzichtelijk te houden				
							Detectie van Domain name generating algoritmes a.d.h.v. Nx clustering				Realtime monitoren wordt moeilijker				

	Hoe zit het systeem in elkaar?				Dataverzameling		Dataverwerking/-gebruik		Belang en effect		Encryptie			Overig	
	Sensor-/netwerkopbouw	Plek in recursieproces	Eigen resolver/daarop controle?	Gewenste meekijkers?	Welke data wordt er verzameld	Hoe en waar wordt data verzameld	Inzetten/verwerken van kennis	Reactie van "systeem" op kennis	Hoe belangrijk is de DNS-informatie?	Wat als niet meer zichtbaar?	Effect	Voorkomen/ tegenkomen	Implementatie en ontwikkeling	Criminaliteit	Alternatieven
CERT-UT	query logging (wie heeft wat opgevraagd en wanneer)	Op eigen resolver	Ja	Ja, mee akkoord in privacyovereenkomst	IP-adressen	Registreren van IP-adressen voor management system	Afhandelen van klachten over IP-adressen	Alleen verdacht gedrag wordt opgeslagen		Ook andere dingen dan query's waar data op gebaseerd wordt	Waarschijnlijk klein, malware zal OS gebruiken i.p.v. browser	Nog niet tegengekomen	Firefox: altijd naar externe resolver		DoT en DoH op lokale resolvers aanbieden
	Quarantainenet van Tesorion op de resolver				DNS en DHCP logging	Quarantainenet van Tesorion op de resolver	Analyse van de query log van de nameserver	Doorsturen van belangrijke informatie naar Quarantainenet servers		Basis malwaredetectie zal veel moeilijker gaan worden		Vrij en open netwerk	Chrome: checken of lokale resolver DoT of DoH ondersteunt, anders "normaal DNS"		
	Analyse van de query log van de nameserver				query logging (wie heeft wat opgevraagd en wanneer)	Uit e-mails worden verdachte domeinnamen verzameld	Ingezet voor detectie in Quarantainenet	Kansberekening op de geïnfecteerdheid van een machine		Besmette machines zullen niet snel genoeg gedetecteerd en in quarantaine gezet worden					
	Doorsturen van belangrijke informatie naar Quarantainenet servers						Ook gedragsanalyse van clients	Geeft melding bij verbinding met het verdachte domein							
	Kansberekening op de geïnfecteerdheid van een machine							Bij hoge waarschijnlijkheid op kwaadaardig gedrag client in quarantaine geplaatst							
	Bij hoge waarschijnlijkheid op kwaadaardig gedrag client in quarantaine geplaatst														
Saxion	Gebruik van externe resolvers voor eigen studenten en medewerkers geblokkeerd	Op eigen resolver	Ja	Ja	Domeinnaam aanvragen	In de name server	Analyse of domeinnamen malicious zijn	Blokkeren van de aanvraag	Wordt niet heel veel mee gedaan	De functionaliteiten van Quarantainenet zal vervallen		Op de firewall al regelmatig tegengekomen en geblokkeerd		DDOS-aanvallen gehad die door studenten over het netwerk besteld waren, achterhalen dan niet meer mogelijk	Zelf DoT en DoH aanbieden, maar daarbij nog veel obstakels met certificaten
	Gasten kunnen dat nog wel (eduroamafspraken)	Bij de firewall (tussen resolver en internet)	Draait met Unbound		IP-adressen	In de firewall (palo alto)	Forensisch onderzoek bij incidenten	Blokkeren van IP-adressen	Voor Quarantainenet redelijk belangrijk	Blacklists zullen ook bijna waardeloos zijn					SSL-decryptie
	De firewall kan op applicatieniveau blokkeren, dit blokkeert DoT en DoH naar bekende IP-adressen en TLS-certificaten				kopie van DNS-aanvragen doorgestuurd naar Quarantainenet	Blacklists van Palo Alto	Bepalen wat er is gebeurd, wie er aanvragen heeft gedaan en/of verbinding heeft gemaakt	Automatische blokkering van Quarantainenet bij redelijke zekerheid van ongewenst gedrag	Voor detectie van malware vrij belangrijk	Als mensen onze DNS service niet gebruiken, dan ziet de firewall het ook niet (bij DoT en DoH)					
	Op het wireless netwerk wordt de controle gedaan door Quarantainenet						Malwaredetectie	Servicedesk checkt één keer per dag alles met niet genoeg zekerheid							
							Blokkering op de firewall								

Appendix C

Specific coding overviews

C.1 The general detection system

	Plek in het recursieproces						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Zo dicht mogelijk bij de client	Ja	x	x	x	x	x	x
Tussen client en resolver	x	soms, tussen interne resolvers	x	Ja (voornamelijk)	Ja (voornamelijk)	x	x
Bij de resolver	Door overnemen	x	x	x	Ja (dataverzameling)	Ja	Ja
Tussen resolver en nameserver	x	Ja	Ja	Ja	x	x	Firewall
	Eigen resolver/daarop controle?						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Interne resolver bij bedrijf/clients	x	Afhankelijk van de deelnemer	Ja (active directory)	Ja, geen controle op	Ja	x	x
Eigen resolver	x	x	x	x	x	Ja	Ja (unbound)
Overname van recursive resolvers	Ja	x	x	x	x	x	x
	Gewenste meekijker?						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Gewenste meekijker	Nee	Ja	Ja	Ja	Ja	Ja	Ja

C.2 Data collection

Welke data wordt er verzameld?

	THTC	NCSC	Fox-IT	Northwave	Tesorion	UT-Cert	Saxion
Metadata/flowdata	Ja	x	x	Ja	Ja	x	x
Domeinnamen, IP-adressen en poorten	Ja	Ja	Ja	x	Ja	Ja	Ja
Indicatoren	Ja, Threat intell (zie hoe en waar)	Ja (en bedreigingsinformatie)	Ja, Intell	Ja	Ja, Threat intell (zie hoe en waar)	x	Ja, blacklists
Netwerkverkeer van en naar het internet	x	Ja	x	Ja	Ja	x	x
Logbestanden	x	x	Ja	Ja	Syslog	Ja, DNS, DHCP en query logging	x

Hoe en waar wordt data verzameld?

	THTC	NCSC	Fox-IT	Northwave	Tesorion	UT-Cert	Saxion
Taps	Ja	x	x	x	DNS-tap	x	x
Threat intell	Passive DNS en CTI	Intell providers, CTI	Virustotal en Passive DNS	Virustotal en Passive DNS	Ja	Domeinnamen uit verdachte e-mails	Blacklist van Palo alto
Data van servers	Vorderen	x	x	x	DNS detector op DNS-servers	Quarantainenet op resolver	Quarantainenet op resolver
Edge van een netwerk	x	Ja, soms meerdere plekken	Ja	Core switch	firewall of coreswitch stuurt data door naar appliance	x	Palo alto op firewall

C.3 Data processing

Hoe wordt de kennis ingezet en verwerkt?

	THTC	NCSC	Fox-IT	Northwave	Tesorion	UT-Cert	Saxion
Gedragsanalyse	Ja, beter beeld gebruiker	x	x	x	Ja --> quarantainenet	Ja	Ja
Achterhalen van servers	Ja	x	Als er reverse DNS of VPN in spel is	x	x	x	x
Correlatie van opvallende zaken met passive DNS	Ja, vaak lastig om te bewijzen	Ja (CTI)	Ja	Ja	Ja	Ja	Blacklists op firewall
Onderzoeken	x	x	Ja	x	Telemetrische metingen a.d.h.v. bloom filters	x	forensisch onderzoek bij incidenten
Monitoren	x	Landelijk beeld voor verbetering	Typosquats, forward lookups	Indicatoren controleren a.d.h.v. logs en geschiedenis	x	x	Controle en blokkeren op servers en firewall
Afhandelen van klachten	x	x	x	x	x	Over IP-adressen	x
Blokkering	x	x	x	x	x	x	Ja, op de firewall
Detectie:	A.d.h.v. domeinnaam (als merk of voor afscherming)	Ja	Ja	Signature-based detection voor filtering op kwaadaardige aspecten, packet inspection voor alarmen	Geautomatiseerde realtime detectie	Ja	Ja
: Malware	x	Ja	x	x	Ja	x	Ja
: Actoren	x	Ja	APT-campagnes	x	x	x	x
: Anomaly	x	x	Op basis van DNS logs	x	Ja	x	x
: Domain name generating algoritmes	x	x	Ja	x	Ja, Nx clustering	x	x
: Phishing	x	x	x	x	Ja	x	x

Reactie van het systeem op de kennis

	THTC	NCSC	Fox-IT	Northwave	Tesorion	UT-Cert	Saxion
Aantonen (bewijzen) van kwaadaardig gedrag voor vervolging	Ja, strafrechtelijk onderzoek	x	x	x	x	x	x
"Hits/Alarmen" bij mogelijk verdacht gedrag	x	Ja	x	Ja	Ja	Ja	Ja
Opgeslagen data met terugwerkende kracht controleren	x	Ja	x	Ja	x	x	x
Indicatoren controleren a.d.h.v. logs	x	x	x	Ja	x	x	x
Achterhalen van de originele client (na alarm/hit)	x	x	x	Ja	Ja	Ja	Ja
Isoleren van de originele client	x	x	x	x	Ja	Ja	Ja
Blokkeren van de query's/IP-adressen	x	x	x	x	x	x	Ja

C.4 Importance and effect

	Hoe belangrijk is DNS-informatie?						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Afhankelijk van verdachte en doel	Ja	x	x	x	x	x	x
Belangrijk voor detectie van dreigingen	x	Ja	x	x	Ja	x	Ja
Informatie ook anders te verkrijgen	x	Ja	x	x	Ja	x	x
Werkt veelal als indicator	x	x	Ja	x	x	x	Ja
Wel zwaartepunt bij Cobalt Strike	x	x	Ja	x	x	x	x

	Wat als DNS-informatie niet meer zichtbaar is?						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Wat er niet is kan niet gevorderd worden	Ja	x	x	x	x	x	x
Detectie van dreigingen wordt moeilijker	x	Ja	x	Ja, achterhalen originele query	x	Ja, niet snel genoeg in quarantaine	x
Alternatieven nodig om aan de data te komen	x	Ja	x	Ja, metadata?	Ja, andere tools	Ja	x
Zorgen dat je er niet afhankelijk van bent	x	x	x	x	Ja	x	x
Functionaliteiten van bescherming zal vervallen	x	x	x	x	x	x	Ja

C.5 Encryption

	Effect van de encryptie						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Altijd nog data lekken	Ja	x	x	x	x	x	x
Eindpunt altijd nog duidelijk zijn	Ja	x	x	x	x	x	x
Meer moeite gedaan worden om data inzichtelijk te houden	x	Ja	Ja	Ja	Ja	x	x
Realtime monitoren wordt moeilijker	x	x	Ja	x	Ja	x	x
Meer single points of failure die veel data hebben	x	x	Ja	x	x	x	x
DoT is bij de edge te blokkeren, DoH bijna niet, bij cloud niet bekend wat te blokkeren	x	x	x	x	Ja	x	x

	Voorkomen/tegenkomen van encryptie						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Vrij weinig gezien	Ja	Ja, DoH meer dan DoT	x	x	Ja	x	x
Niet gezien	x	x	Niet op gelet	Niet op gelet	x	Ja	x
Zelf gebruiken	x	x	Ja	x	x	x	x
Malware onderzoek met DoH	x	x	x	x	Ja	x	x
Blokkeren op firewall	x	x	x	x	x	x	Ja, regelmatig

	Implementatie en ontwikkeling						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
DoT en DoH zullen nooit volledig alles overnemen	Ja	x	x	x	Ja	x	x
DoT en DoH zullen geblokkeerd worden op netwerken	Ja	x	x	x	x	x	x
Juridische aanpassingen nodig voor DoH	Ja	x	x	x	x	x	x
Zelfde als met TLS (HTTP to HTTPS)	x	Ja	Ja	Ja	x	x	x
Minder data uit handshake (ziet alleen DoT/DoH server)	x	x	Ja	x	x	x	x
Netwerkmonitoring alleen nog maar voor context	x	x	x	Ja	x	x	x
Ondersteuning voor DoT en DoH goed werkend is moeilijk (geen standaard, OS-ondersteuning, certificaten)	x	x	x	x	Ja	Ja	x

C.6 Criminality and alternatives

	Verwachte criminaliteit						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Gebruik van encryptie door "default"	Ja	x	x	x	x	x	x
Eigen encryptiesysteem bouwen voor afscherming en zekerheid	Ja	x	x	x	x	x	x
DoH kan gebruikt worden als exfiltratiekanaal (in combinatie met DNS-tunnels)	Ja	Ja, actoren	Ja	Ja, stat actoren	Ja	x	x
Command en control over DoH	x	Ja	x	Ja	x	x	x
Vergemakkelijking van gebruik door tooltjes	x	Ja	x	x	x	x	x
Spoofingaanvallen door terugvallen naar NetBios	x	x	Ja	x	x	x	x
Cash poisoning moeilijker te detecteren	x	x	x	Ja	x	x	x
Validiteit checken van query en response moeilijker	x	x	x	Ja	x	x	x
Achterhalen van aanvallen vanuit eigen netwerk moeilijker	x	x	x	x	x	x	Ja

	Alternatieven voor dataverzameling						
	THTC	**NCSC**	**Fox-IT**	**Northwave**	**Tesorion**	**UT-Cert**	**Saxion**
Vorderen bij ISP en open resolvers	Ja	x	x	x	x	x	x
TLS-interceptie	x	Ja (middle box)	x	x	Ja, deep packet inspection	x	Ja, SSL-decryptie
Gedrag van criminelen in kaart brengen en voorspellen	x	x	Ja	x	x	x	x
Zelf DoT en DoH resolvers aanbieden	x	x	x	x	x	Ja	Ja
Andere locatie van sensoren:	x	Ja (impliciet)	Ja (impliciet)	Ja (impliciet)	x	x	x
: Endpoint detectie	x	Ja	Ja	Ja	Ja	x	x
: DNS resolver	x	Ja	Ja	Ja	Ja (doen al)	x	x
Er lekt altijd nog iets van data:	Ja	Ja (impliciet)	x	Ja (impliciet)	Ja (impliciet)	x	x
: TLS handshake voor servernaam en IP	x	Ja	x	x	Ja (onderzoek gedaan)	x	x
: Metadata/flowdata	x	x	x	Ja	Ja	x	x

Appendix D

setup and settings on experiment software

D.1 Iodine on the client

```
root@anja:/home/anja1# sudo iodine -v
iodine IP over DNS tunneling client
version: 0.7.0 from 2014-06-16
```

Figure D.1: Iodine version on the client

```
^Croot@anja:/home/anja1# sudo iodine -f -r -P 12345 iodine.dnsjedi.org
Opened dns0
Opened IPv4 UDP socket
Sending DNS queries for iodine.dnsjedi.org to 130.89.0.128
Autodetecting DNS query type (use -T to override).
Using DNS type NULL queries
Version ok, both using protocol v 0x00000502. You are user #0
Setting IP of dns0 to 10.10.10.2
Setting MTU of dns0 to 1130
Server tunnel IP is 10.10.10.1
Skipping raw mode
Using EDNS0 extension
Switching upstream to codec Base128
Server switched upstream to codec Base128
No alternative downstream codec available, using default (Raw)
Switching to lazy mode for low-latency
Server switched to lazy mode
Autoprobing max downstream fragment size... (skip with -m fragsize)
768 ok.. 1152 ok.. ...1344 not ok.. ...1248 not ok.. ...1200 not ok.. 1176 ok.. 1188
 ok.. will use 1188-2=1186
Setting downstream fragment size to max 1186...
Connection setup complete, transmitting data.
```

Figure D.2: Iodine active and running on the client

```
^Croot@anja:/home/anja1# sudo iodine -f -r -P 12345 iodine.dnsjedi.org
Opened dns0
Opened IPv4 UDP socket
Sending DNS queries for iodine.dnsjedi.org to 127.0.2.1
Autodetecting DNS query type (use -T to override).
Using DNS type NULL queries
Version ok, both using protocol v 0x00000502. You are user #0
Setting IP of dns0 to 10.10.10.2
Setting MTU of dns0 to 1130
Server tunnel IP is 10.10.10.1
Skipping raw mode
Using EDNS0 extension
Switching upstream to codec Base128
Server switched upstream to codec Base128
No alternative downstream codec available, using default (Raw)
Switching to lazy mode for low-latency
Server switched to lazy mode
Autoprobing max downstream fragment size... (skip with -m fragsize)
768 ok.. 1152 ok.. ...1344 not ok.. ...1248 not ok.. ...1200 not ok.. ...1176 not ok.. ..1164 ok.. will use 1164-2=1162
Setting downstream fragment size to max 1162...
Connection setup complete, transmitting data.
```

Figure D.3: Iodine active and running on the client over the proxy

D.2 Iodined on the server

```
$ dig +norecurse NS iodine.dnsjedi.org @ns1.surfnet.nl
...
;; AUTHORITY SECTION:
iodine.dnsjedi.org.  10800    IN    NS    anja.dnsjedi.org.

;; ADDITIONAL SECTION:
anja.dnsjedi.org.    10800    IN    A     192.87.172.251
anja.dnsjedi.org.    10800    IN    AAAA  2001:610:1908:ff01:f816:3eff:fe7e:e101
```

Figure D.4: server settings domain

```
anja@anja:~$ sudo iodined -v
iodine IP over DNS tunneling server
version: 0.7.0 from 2014-06-16
```

Figure D.5: Iodined version on the server

```
^Canja@anja:~$ sudo iodined -f 10.10.10.1 iodine.dnsjedi.org
Enter password:
Opened dns0
Setting IP of dns0 to 10.10.10.1
Setting MTU of dns0 to 1130
Opened IPv4 UDP socket
Listening to dns for domain iodine.dnsjedi.org
```

Figure D.6: Iodined active and running on the server

D.3 Dnscrypt-proxy on the client

```
root@anja:/home/anja1# systemctl status dnscrypt-proxy
● dnscrypt-proxy.service - DNSCrypt client proxy
     Loaded: loaded (/lib/systemd/system/dnscrypt-proxy.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2020-05-27 11:01:20 CEST; 1 weeks 3 days ago
TriggeredBy: ● dnscrypt-proxy.socket
       Docs: https://github.com/DNSCrypt/dnscrypt-proxy/wiki
   Main PID: 481 (dnscrypt-proxy)
      Tasks: 9 (limit: 1132)
     Memory: 8.4M
     CGroup: /system.slice/dnscrypt-proxy.service
             └─481 /usr/sbin/dnscrypt-proxy -config /etc/dnscrypt-proxy/dnscrypt-proxy.toml

May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Network connectivity detected
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Source [/var/cache/dnscrypt-proxy/public-resolvers.md] loaded
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Firefox workaround initialized
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [WARNING] Systemd sockets are untested and unsupported - use at your own risk
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Wiring systemd TCP socket #0, dnscrypt-proxy.socket, 127.0.2.1:53
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Wiring systemd UDP socket #1, dnscrypt-proxy.socket, 127.0.2.1:53
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] [cloudflare] OK (DoH) - rtt: 5ms
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] Server with the lowest initial latency: cloudflare (rtt: 5ms)
May 27 11:01:22 anja dnscrypt-proxy[481]: [2020-05-27 11:01:22] [NOTICE] dnscrypt-proxy is ready - live servers: 1
Jun 06 11:07:05 anja dnscrypt-proxy[481]: [2020-06-06 11:07:05] [NOTICE] Server with the lowest initial latency: cloudflare (rtt: 11ms)
```

Figure D.7: Dnscrypt-proxy active and running on the client

Appendix E

Located IP addresses

E.1 Authoritative name server

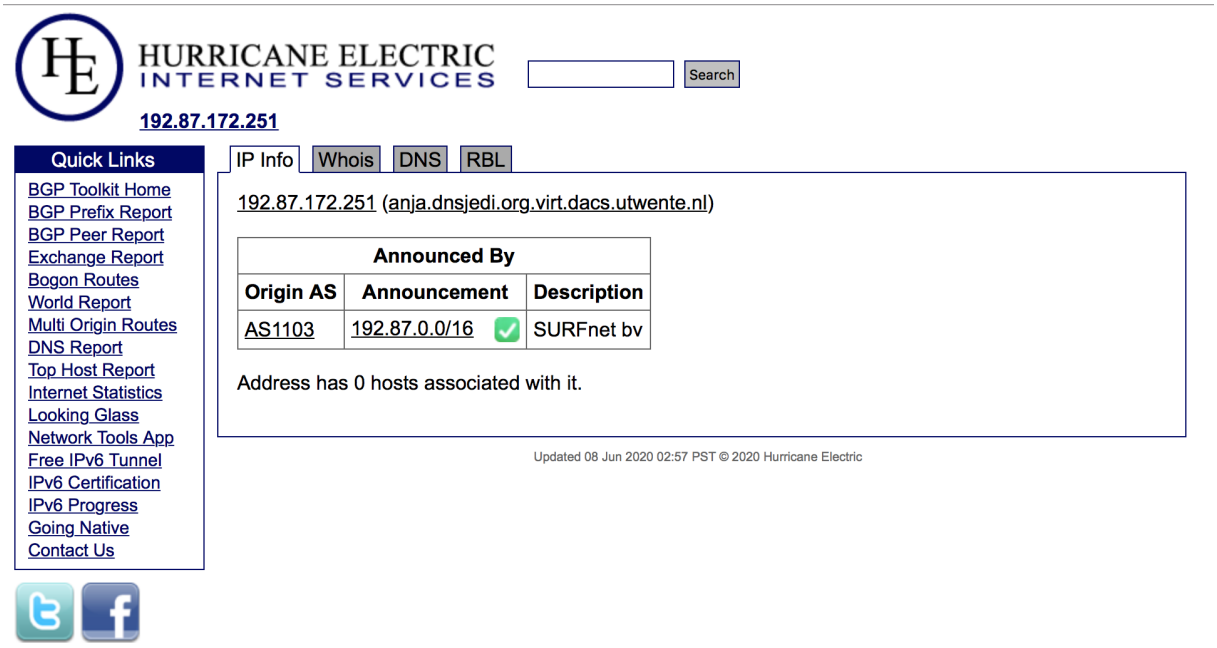


Figure E.1: The IP address of the server

E.2 University of Twente

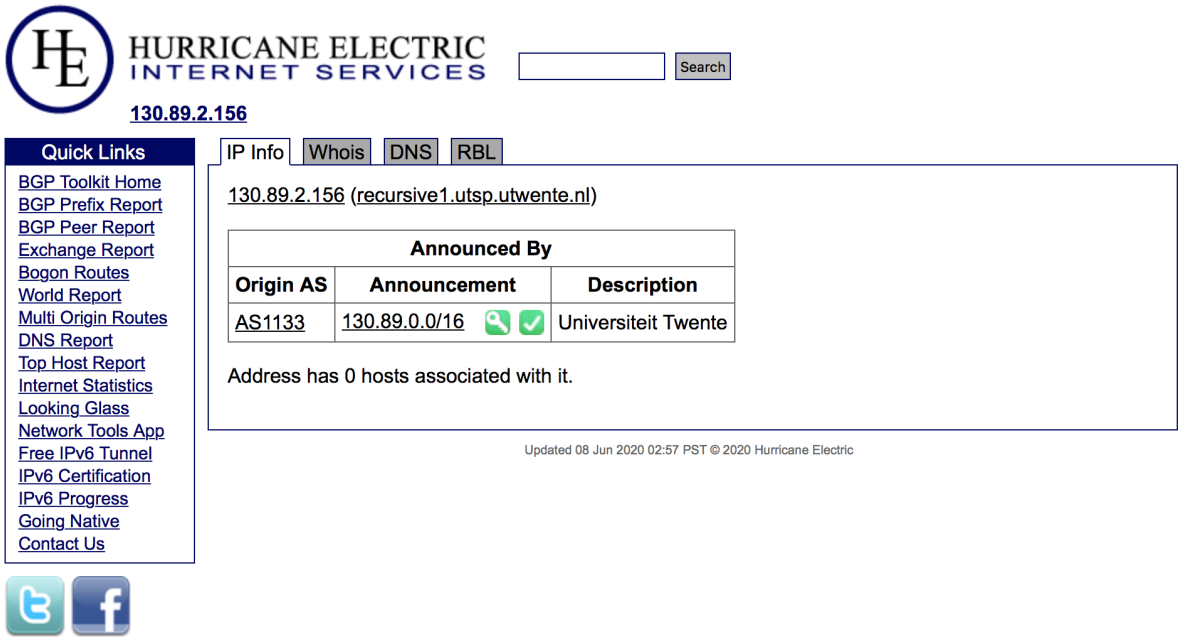


Figure E.2: The IP address of the University of Twente

E.3 Cloudflare incoming

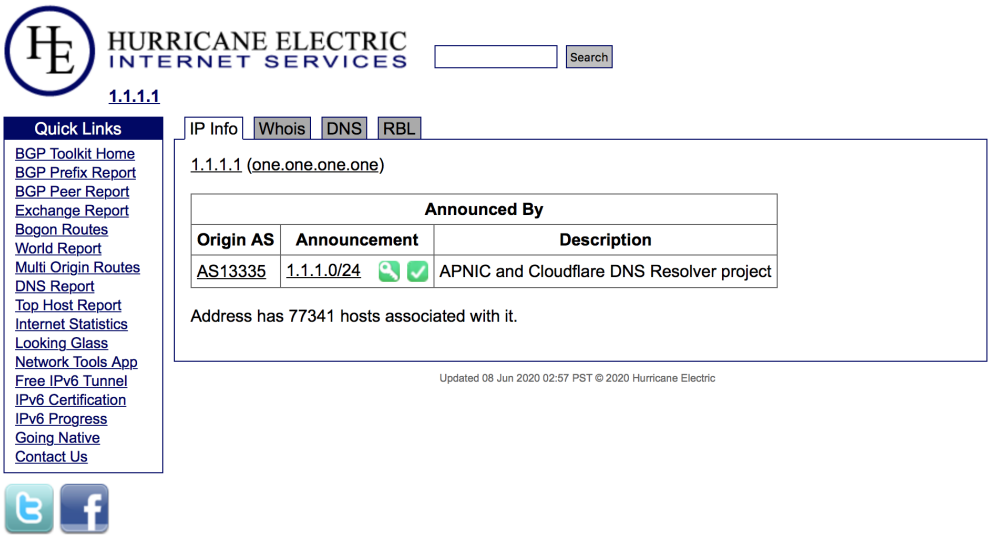


Figure E.3: The IP address of the Cloudflare resolvers for incoming queries

E.4 Cloudflare outgoing

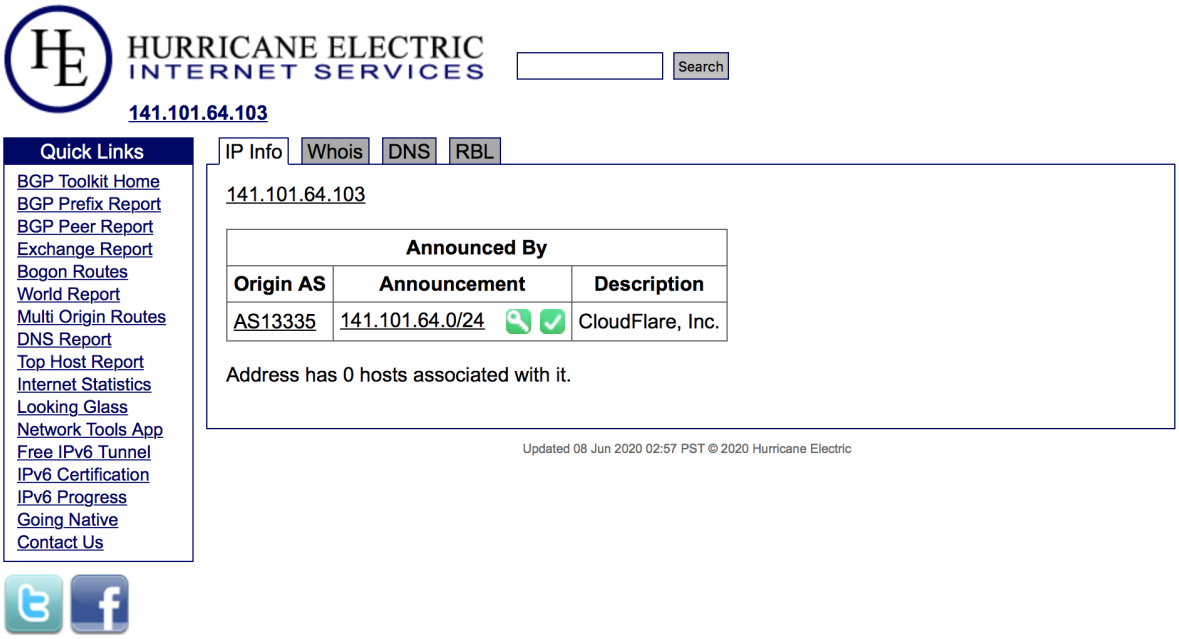


Figure E.4: The IP address of the Cloudflare resolvers that sent requests to servers

Appendix F

Interview Transcripts

To respect the privacy of the experts, the transcripts of the interviews have been omitted from this version of the thesis. Access to the transcripts is available on request.