IMPROVING ACCESS TO DNS DATASETS THROUGH THE LARGE-SCALE COLLECTION OF ACTIVE-DNS DATA

A Dissertation
Presented to
The Academic Faculty

By

Athanasios Kountouras

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
Computer Science
College of Computing

Georgia Institute of Technology

May 2023

IMPROVING ACCESS TO DNS DATASETS THROUGH THE LARGE-SCALE COLLECTION OF ACTIVE-DNS DATA

Thesis committee:

Dr. Manos Antonakakis
Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Roberto Perdisci
Computer Science
University of Georgia

Dr. Mustaque Ahamad
College of Computing
Georgia Institute of Technology

Dr. Charles Lever
College of Computing
Georgia Institute of Technology

Dr. Angelos Keromytis
Electrical and Computer Engineering
Georgia Institute of Technology

Date approved: Feb 24, 2023

It is difficult to predict, especially the future.

Niels Bohr

For my mother.

ACKNOWLEDGMENTS

As with any worthwhile target in life, the important part is the journey, and I am very thankful to all the people that have helped me, and allowed me to continue to the eventual destination. I can honestly admit that without the help of my fellow researchers, friends, and family, I could not have made it to this point, for all the support I am and I will be truly thankful.

The first person I have to mention is my advisor Manos Antonakakis to whom I am deeply indebted, for all his support, mentoring, and valuable advice and insights. Thanks to Manos, I've made it this far in my academic journey, and through his mentoring, I learned how to be better in all aspects of life, not just the professional.

I'm also extremely grateful to my closest collaborator and friend, Panagiotis Kintis, who helped me throughout my academic journey. He is the person that after my advisor, has helped me from the moment I joined the program and up to now; from mundane issues to important ones, he was always willing to help, and I will always thank him for it.

Furthermore, I also have to recognize all the help I received from all the other collaborators I've had the pleasure of working with. I could not have undertaken this journey without the help of Yizheng Chen, that supported me from the beginning and taught me how to operate and work in the new working environment I found myself. I am also thankful for Chaz Lever, who, throughout my time in the program, helped me understand the requirements of academic work. Also, this entire thesis and endeavor would not have been possible without Dave Dagon and his in-depth DNS knowledge and brilliant ideas. Thanks should also go to Yacin Nadji, who was a great help and influence at the beginning.

I would also like to recognize all the newer members of the COEUS center for all their help and their friendship. I am genuinely proud to have been a member of such a vibrant and friendly group of people. First, I would like to thank Thanos Avgetidis, who has assisted me since he joined the center, and Zane Ma, who provided excellent feedback

on my thesis. I also want to thank Aaron Faulkenberry and Brianna Herron, who have been good supporting friends, and even their little pet Pollyanna that I kept company for a few weeks. I cannot forget to mention the rest of the COEUS team: Thomas Papastergiou, Omar Alrawi, Miuyin Yong Wong, Kleanthis Karakolios as well as the technical staff that provides support for our projects, William Garrison and Alex Neal, as well as our excellent accountant Michael Mitchel. Each and every one of you, have helped me get here, and I thank you for it.

Of course, I would be remiss in not mentioning my family in Greece, especially my parents and brother, for their constant support and love. They sacrificed so much for me to be here and have this opportunity, so this thesis is dedicated to them. I also have to thank all the friends I've made along the way and who have stood by me in good and difficult times and made me feel like I am at a home away from home: Orestis, Christopher, Eva, Thalia, Kostas, I want to thank you for being such good friends. Also I want to thank my slightly older and more experienced friends such as Theologos Buntourelis for all the free flights and amazing times as well as George Kotsalis who supported me with his experience and friendship. I'd also like to mention two people that have helped me tremendously to confront any non-academic issues I encountered and gave me the skills to go forward; thank you, Dr. Croft and Dr. Thomas.

Lastly, I would like to thank the members of my defense committee. Dr. Mustaq Ahamad, is a great teacher and a very helpful collaborator. Dr. Roberto Perdisci, a longtime collaborator of the center and always a great person. Dr. Angelos Keromytis, who is a great motivator and exceptionally knowledgeable in cybersecurity.

Thank you, to each and everyone of you, for helping me get this far, I couldn't be here without all of you!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The Internet has changed significantly in size, interconnectedness, speed, capability, and usability over the years. Especially after a few years of remote work and remote learning, we can safely say that the Internet is an essential resource for the modern world. While the internet has gone through massive expansion, the backbone of interconnected networks still rely on many of the same fundamental technologies. The Domain Name System (DNS) is one of those fundamental Internet technologies; its main task is to translate human readable domain names into resources on the ever-growing network.

Due to the vantage point it provides, the security community continues to leverage the Domain Name System for studying current abuse and as a building block for tools that combat new and existing internet threats. In order to develop, evaluate, and deploy defensive mechanisms, researchers and threat analysts need access to quality datasets. Such datasets will enable new algorithms and methodologies that can assist with early detection, better tracking, and a fuller understanding of the lifetime of modern Internet threats.

To that end, this thesis presents the concept of Active DNS data collection through a distributed querying infrastructure. More specifically, we show how this new public dataset, which we name Active DNS, compares against traditionally utilized passive DNS datasets. We document our system's unique features that enable it to function as an alternative to passive DNS data in many applications. We then demonstrate the ability of Active DNS data to detect online abuse by utilizing it to amplify already known malicious web infrastructure and potentially identify new abusive infrastructure before use. Finally, we show how our distributed querying system, *Thales*, allows us to study the operational aspects of the global DNS infrastructure, explicitly investigating the proliferation of a new DNS extension and measuring the impact and efficacy of this new DNS extension through active probing.

CHAPTER 1
INTRODUCTION

The Internet is the first truly global network consisting of billions of computers and other electronic devices. With access to the Internet, it is possible for users to access almost any information, communicate with anyone else in the world, and do so much more. One very important component of the Internet that assists in connecting resources together is the Domain Name System (DNS) which primarily maps domain names to IP addresses or other resources, thus providing an essential service to most Internet-connected applications. Due to this reliance, it is not an exaggeration to claim that DNS is one of the Internet's core protocols, providing an agile and reliable infrastructure for Internet-based applications. As the Internet and its uses have evolved through decades of constant development, so has DNS, resulting in changes that affect the ability of researchers to collect and process DNS data, especially in the recent past.

Malicious actors also utilize DNS for the same reasons that any Internet application does, to provide an agile and reliable infrastructure for their applications, even if their eventual purpose is malicious. That is one reason why the security community has been studying the Domain Name System and DNS datasets since the introduction of the dataset we call Passive DNS [1]. However, in the past decade, how DNS operates for many users has changed as the protocol has evolved. For example, DNS over TLS (DoT) [2, 3] encapsulates DNS packets inside Transport Layer Security (TLS) packets in an effort to improve user privacy. Another recent change introduced as an extension to DNS is called EDNS Client Subnet (ECS) and tries to optimize the resolution process for CDN traffic. These changes come with unintentional security consequences, especially for users that are unaware of these changes.

Considering the constantly changing situation, we must evaluate the primary type of

DNS dataset used in research by the security community. The *de facto* dataset for studying DNS and conducting DNS research, as we mentioned, is called Passive DNS [1] and relies on monitoring the DNS traffic of a local network, nowadays usually by some endpoint security appliance, to obtain the associations between domain names and resources. The main issue with this type of dataset is that it is limited to the local network visibility and thus can only capture a small subset of the global DNS associations. It also relies on the activity on the network, which makes the dataset entries inconsistent from a longitudinal standpoint. Also, this DNS dataset type is increasingly affected by the adoption of DNS extensions, such as the already mentioned DoT, that encrypts DNS traffic and thus prevents the effective collection of DNS data. Overall, passive DNS can be useful for cybersecurity research, but, as the situation stands, it faces issues with dataset accessibility, data uniformity, and completeness.

Other methods that network operators have used to capture DNS data for use in the security community include sandbox execution or dynamic execution of malicious software in an effort to capture the domain name resolution attempts. These, however, are only snapshots and cannot provide researchers with longitudinal data. As with any useful resource, there are, of course, commercial providers of DNS data, but as we will further discuss, this is far from an ideal solution to the issues we have already mentioned, as it is subject to the same limitations as any other passive DNS collection. Additionally, these datasets can be expensive to obtain and often require restrictive legal agreements, limiting the potential sharing of datasets and other information, which can be essential for new research to be evaluated and recreated by other researchers. Ultimately, most DNS datasets involve network-level packet captures that require filtering and processing before they can be utilized, further raising the barrier of entry for including DNS data in security research, thus limiting the repeatability of research in network security.

Providing the security community with a free and open DNS dataset will inevitably lead to further innovation in the types of security research conducted using DNS data.

The community has already proposed a number of analytical systems [4, 5, 6, 7] that rely upon DNS data to identify and react to new threats in order to keep networks secure. By offering an easy-to-obtain, already processed (parsed), wide-reaching, longitudinal dataset without legal encumberment, we wish to encourage fresh innovation in DNS research and enable repeatability of security research through a DNS dataset we call Active DNS. We are confident that our dataset will be a great tool for security researchers since it has served us so well over the years and has enabled us to conduct experiments such as the ones we demonstrate in chapter 4 where we will also show how we utilize Active DNS data to enrich other network related datasets.

To that end, we identified that a system that can systematically collect DNS data for a large portion of the global domain name space would offer a partial solution to some of the issues researchers and security professionals face when they try to source DNS datasets. This system would be able to interface with the global DNS system and record a large and expansive DNS dataset efficiently while also operating for years without major data loss.

In this thesis, we show how we designed our answer to these requirements; we present "*Thales*," a system capable of generating billions of DNS queries per day while collecting and processing the corresponding responses in order to generate what we call an Active DNS dataset. We document the design, implementation, and evolution of the infrastructure that supports and enables *Thales* to operate. We then discuss and document the unique properties of the resulting dataset, which we name "Active DNS" and provide extensive comparisons showing how it compares with traditional passive DNS data captures. We provide empirical evidence which suggests that the Active DNS data that *Thales* generates can be used in several security applications and related research. Furthermore, we present a short taxonomy of existing research created using Active DNS data as an artifact of its introduction and sharing with the community. This taxonomy also documents the reception the system received from the community, which we further document in section 3.5. Based on these observations, we note that the resulting dataset can act as a standardized

and well-documented dataset for validating existing and new research in network security. Ultimately, providing this dataset will increase the repeatability of research in our field.

After the introduction of *Thales* and the Active DNS dataset, we present a study demonstrating how Active DNS data can be utilized in a security application. More specifically, we present a study in which we assisted in studying the issue of Technical Support Scams (TSS), in which scammers dupe their victims into sending hundreds of dollars for fake technical support services. The TSS scam problem started with scammers making cold calls to victims claiming to be legitimate technology vendors but has now evolved into sophisticated online abuse tactics to get customers to call phone numbers under the scammers' control. We use our newly documented Active DNS dataset through a novel enrichment technique to expose previously unseen portions of networks operated by cybercriminals to scam users. In fact, after initially identifying more than 5,000 TSS-related domains operated by TSS scammers, we utilize Active DNS data to amplify our information and expose an even more significant portion of their network, *expanding our identified scam domains to over 9,000*. We then further expand on documenting the capabilities offered by Active DNS data by showing how the dataset can help identify other already studied types of abuse, such as domain reputation abuse.

Our final study focuses on the expansive capabilities of our previously mentioned collection system, *Thales*, and how its contributions can go beyond just generating the Active DNS dataset. In this study, we show how *Thales* enables us to conduct a study of a recent addition to the DNS protocol itself, called EDNS Client Subnet (ECS), that changes how client IP information is shared with the DNS infrastructure in order to optimize for Content Delivery Network (CDN) selection. For that purpose, ECS provides more granular client information to authoritative DNS servers that reveal information about the underlying client making a request. We show how the protocol's real-world adoption grew before and after its official adoption as an RFC in 2016. We then examine the steady adoption rate over the years, noting that numerous popular DNS providers support it. We then dis-

cuss how the newly introduced, as part of the protocol extension, information can assist researchers when they utilize it as a tool for forensic investigations and the ability to conduct overall security measurements for a large portion of the Internet. Finally, by utilizing a new capability of *Thales* to conduct a novel worldwide study, we demonstrate that many ECS-enabled domains do not benefit from ECS use but rather activate it without regard for the potential impact on the end-users' privacy.

1.1 Motivation

As with any scientific discipline, to understand, measure, and make accurate observations with any predictive ability, we need to start by assembling a dataset that adequately represents the object of our study within the proper context for the specific study. In our field of information security and even more specifically, network security, one of the more widely utilized datasets is based on one of the fundamental protocols of the Internet, DNS, which provides information that ties the human understandable domain names into Internet addresses that software can use to serve the client. This dataset is called passive DNS and requires the monitoring and recording of all DNS traffic in a sizable network. The value of the dataset comes from all the recorded associations that DNS serves. Thus, the more extensive the network, the more valuable the dataset can be, as it contains more data points and captures a more significant portion of the overall associations across the Internet.

Even though passive DNS is widely used, it faces several limitations that diminish its effectiveness for researchers, such as limited global visibility or poor visibility into newly registered domains. In Passive DNS, the data collected is limited by the domains resolved within the network. For example, an overwhelming number of new domain names appear on the Internet daily and many will never be queried from the collecting network. Further, DNS is changing while also becoming increasingly encrypted, limiting the data's wide-reaching visibility while potentially challenging experiment repeatability. Of course, since this is such a valuable tool for network security, researchers have the option to utilize

commercial dataset offerings, which come at a cost and with various legal restrictions (privacy, anonymization, access control) if the collection is not a practical option (small-size network, insufficient resources).

Our primary motivation for this work was to generate and share with the information security community a DNS dataset that can provide the main benefits of the already established passive DNS datasets while simultaneously avoiding the main issues preventing its more widespread usage and introducing an important element of standardization and consistency in the data collected. To overcome the limitations of existing passive DNS datasets, we present a new type of DNS collection system called *Thales* to the community, and we document the resulting dataset, which we name "Active DNS." We specifically engineered *Thales* to be an expansive, efficient, distributed collection system to generate a large, diverse, longitudinal, and legally unencumbered DNS dataset collected in a systematic, consistent, and widely scalable fashion. Researchers can use the collected data as either an alternative to traditional Passive DNS or to enrich and complement other datasets, including Passive DNS. Furthermore, we document the unique characteristics of our distributed collection system, *Thales*, and we show how we utilize its powerful active querying capabilities to conduct operational research based on active probing. We further hope our dataset can serve as a well-documented and standardized dataset with a consistently broad reach of global Internet resources that can help verify security research for years to come.

1.2 Thesis Statement

This thesis describes the design and implementation of *Thales*, a data collection system that can actively, efficiently, and reliably collect and store the largest open DNS dataset, *Active DNS*. Active DNS, which includes domain names from the majority of Top-Level Domains, can complement or replace passive DNS datasets for network measurements, infrastructure expansion, and cyber threat analyses and detection.

1.3 Contributions

Active DNS collection system: As we have already established, DNS datasets have plenty of important security applications [4, 5, 6, 7]. Passive DNS has long been the only evaluated and well-understood dataset for DNS-based security research. However, as we have already mentioned, Passive DNS datasets come with several limitations, including being expensive to obtain, either by purchasing or collecting, and are often burdened with legal restrictions due to the presence of user activity in the dataset. Further, Passive DNS datasets now provide incomplete coverage due to the increasing adoption of encrypted DNS communications through new extensions such as DoT [2, 3]. This thesis presents a new DNS dataset created by an active, distributed system sending DNS queries to most operating TLDs through a distributed DNS collection system we call *Thales*. Our system collects billions of DNS records efficiently, and we provide the resulting dataset, which we call, Active DNS, for free to the community in an accessible manner. We document the system and its constituent components along with an introduction to the unique attributes of actively collected DNS data that make it a well-received tool for studying online abuse and a dataset that can enable repeatable research. We then offer comparisons between our free Active DNS dataset and passive DNS and discuss the limitations of each dataset.

Infrastructure expansion through free Active DNS datasets: Since it was initially made available, the Active-DNS dataset has been utilized by researchers and has both enabled new studies as a free alternative of passive DNS and has aided other works as Internet infrastructure amplification. A case of the later and one of the first uses of Active DNS, is our assistance to Bharat Srinivasan [8] in conducting the first systematic study of Technical Support Scam (TSS) abuse of search-and-ad channels. Technical Support Scams is an old issue in which virtual actors trick unaware victims into spending large amounts of money for fake support services. Our contribution to this work was developing network enrichment techniques that leveraged the Active-DNS dataset. More importantly, we showcase

how we have utilized Active-DNS to amplify the scam domain names found by Bharat through their infrastructure increasing the number of TSS domain names by 130% of the originally found domain names. We then demonstrate how other existing applications of abuse detection, which were initially developed for other types of datasets, such as passive DNS datasets, can be adapted to perform over our Active DNS dataset and how Active DNS can aid in other security relevant tasks such as enhancing public blocklists and tracking malicious domain names.

Studying the operational impact of changes to the global DNS infrastructure through distributed active probing: Finally, we demonstrate how Thales, the system behind Active-DNS has enabled us to study the operational deployment of a new at the time DNS extension called EDNS Client Subnet (ECS) [9], which was initially proposed as an experimental Internet draft in 2011. Our primary concern in monitoring these changes in the global DNS hierarchy and conducting a study is that ECS changes how client IP information is shared with DNS infrastructure to optimize for Content Delivery Network (CDN) selection. Our study utilizes our *Thales* infrastructure to actively probe popular domains and identify whether the users of these domains benefit from having support for ECS. Our study shows that the vast majority of highly ranked ECS-enabled domain names do not currently benefit from the use of the new extension.

1.4 Dissertation Overview

Initially, chapter 2 contains a background of related material necessary in the context of this thesis. It provides the foundation for the research discussed later, including technical details around DNS, information about the contents of DNS packets, the resolution process, and overall data collection considerations. Further, in section 3.5, we quickly review some of the existing works from the research community that utilize Active DNS data to document the impact of Active DNS data in security research.

Then, in chapter 3, we focus on our Active DNS collection infrastructure, *Thales*, where

in section 3.2, we describe how our system operates. We document each component of our system and the changes implemented over the years of operation. We then compare the Active DNS dataset we generate to passively collected DNS data in section 3.4 in the context of network security research. Finally, we include an analysis of each dataset's various properties and unique attributes in subsection 3.4.2.

In chapter 4, we present one of the many applications for Active DNS data, enabling abuse detection by pairing our dataset with information enrichment techniques for network data. More specifically, in subsection 4.2.1, we present part of a collaborative work headed by Bharat Srinivasan in detecting Technical Support Scam (TSS) abuse domains by utilizing search engine query results and then amplifying the visibility and reach of our results with our Active DNS dataset in order to reveal the extended infrastructure that malicious actors are using to target users [8]. Then in section 4.3, we present further case studies examining our ability to adapt the experiments so we can utilize Active DNS data to conduct abuse detection research.

In chapter 5, we examine the operational capabilities of *Thales*, our distributed collection infrastructure, by studying the operational impact that the deployment of a new DNS extension called EDNS Client Subnet (ECS) [9]. This extension introduced new security and privacy implications due to including a portion of the user's IP in the DNS packet for more effective CDN selection. In section 5.3, we present a study that quantifies the real-world adoption of ECS since its inception. Then, in section 5.4, we present the results of a novel experiment that utilized *Thales* in order to quantify whether the users of popular domains that support ECS benefit from it but discovered that the vast majority of highly ranked ECS-enabled domain names do not currently benefit from the use of the new extension.

We conclude the thesis in chapter 6. In section 6.1, we discuss the overall contribution of the systems and research presented in this work. We then discuss the limitations of Active DNS data collection and considerations for anyone utilizing actively queried DNS

datasets in section 6.2. Following that, in section 6.3, we discuss potential improvements for the system and future work that can follow up by utilizing Active DNS datasets. We conclude with some closing remarks in section 6.4.

CHAPTER 2
BACKGROUND AND PREVIOUS WORK

2.1 Background

The domain name system (DNS) [10, 11] operates as the phonebook for the Internet. Its primary task is converting human-friendly identifiers called domain names to computer-readable IP addresses that computers can use to reach resources. This simple process has, over the years, shaped the modern web, but it has also become a resource used by cyber-criminals due to the agility that DNS offers. For that reason, DNS datasets have been a key tool for security researchers as they battle the increasing sophistication of modern attacks. The security community has proposed new analytical systems [4, 5, 6, 7] that utilize passively captured DNS data to shorten the response time necessary to react to new threats and secure networks. These systems rely on the efficient collection and presentation of passive DNS datasets. However, such datasets are difficult to find, challenging to collect, and often require restrictive legal agreements. These obstacles can make further innovation difficult and impede the repeatably of research.

2.1.1 The Domain Name System

Because studying the various aspects of the Domain Name System is fundamental to this work, we present some necessary information about DNS and its important components in this chapter.

The most visible component of the DNS system is the domain name. A domain name is simply a string of text that will be queried to provide the computer with an IP address to contact the appropriate resource. In even simpler terms, a domain name is a text, usually in plain English, that a user types in order to reach a particular webpage. Domain names can

be broken up into various levels, each separated by a dot `[.]`. The section of the domain name to the right of the last dot is called the Top Level Domain (TLD). Usually, the label to the left of that dot is the second level domain (2LD), or in some cases where the TLD is comprised of two labels separated by a dot, such as in many `.co.uk` domains, where we label the second level domain an effective second level domain (e2LD) as this is the portion of the domain that denotes delegation to an entity or organization. Anything that follows to the left of the e2LD before another dot `[.]` is the third level domain (3LD), and the delegation continues by adding labels to the left for increasingly more child labels, as we call them.

To understand why domain names have the above structure, we describe the hierarchy of organizational structures that underlie the Domain Name System. The rightmost label is always the TLD and can be one of the various terms, but not unlimited. The Internet Corporation for Assigned Names and Numbers (ICANN), which has authority over all TLDs used on the Internet, delegates the responsibility of managing these TLDs to individual organizations. There is now an abundance of TLDs created by ICANN for various uses. The first type of TLDs are gTLDs, which include the very common, `com, net, org...`. The second type is *country code TLDs* (ccTLDs), reserved for use by countries and sovereign states. These consist of two-letter identifiers for the corresponding country, such as `".gr",".uk",".ch"` .etc. The third type of TLD is the sponsored TLDs, which ICANN recently introduced as a way for companies to create their own hierarchical communities and include TLDs such as `".app",".edu"` .etc. The final few types of domain TLDs are the infrastructural TLDs, such as `.arpa` and the reserved TLDs, for use in local networks or demonstrations, such as `.localhost` and `.example`.

Three types of nameservers respond to DNS queries, corresponding to the levels of the domain name hierarchy. Root nameservers consist of a list of known DNS contact servers that can then point to the next step of the hierarchy, the TLD nameserver. The TLD nameserver is the server that holds the registrar's record of domain delegations inside

that TLD and is tasked with providing the next point of contact in the hierarchy, which is delegated to a separate organization. After the recursive resolver traverses the root and TLD nameservers, it will arrive at the **authority nameserver**; the authoritative nameserver is the last stop in the nameserver query. If the authoritative name server has access to the requested record, it will return the IP address for the requested hostname to the DNS resolver.

The hierarchy of the domain name system can be abstracted away from the clients through a recursive DNS resolver, referred to as recursive or recursive resolver. Internet service providers (ISPs) often operate recursive resolvers for their clients. However, it is becoming more common for clients to turn to cloud DNS providers such as Google DNS or Cloudflare. When a client wants to access a domain name, it queries the recursive resolver and awaits the final response. In turn, the recursive resolvers must navigate the DNS hierarchy to retrieve the requested resource for the client.

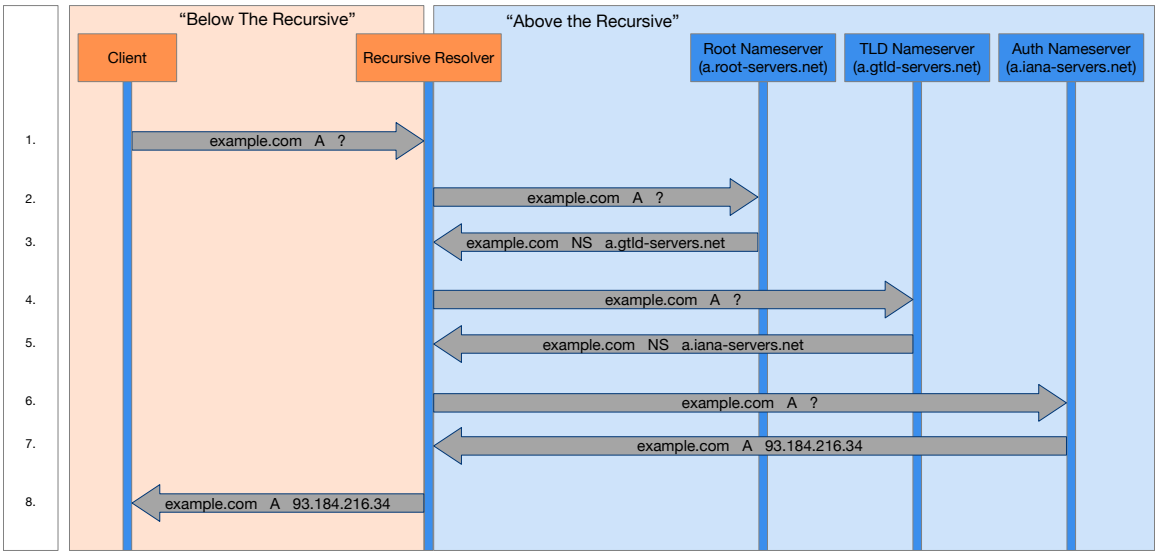


Figure 2.1: Example of the DNS resolution process.

Now that we have established the primary components of the DNS hierarchy, let us explore how the DNS resolution process works in steps.

Figure 2.1 shows the process of resolving the domain name `example.com`, which is the domain suggested for use in such examples by IANA. The DNS resolution or DNS

lookup process can be split into the following eight steps.

- `1:` A client needs to connect to a resource with a domain name. A DNS request is sent to the client's recursive resolver.
- `2:` The resolver (assuming no previous caching) will contact one of the root DNS nameservers.
- `3:` The root DNS server will respond to the recursive resolver with the TLD DNS nameserver (in this case, .com) responsible for the domain names under the TLD zone.
- `4:` The resolver sends the request to the TLD nameserver (`.com`).
- `5:` The TLD nameserver responds with the address of the Authoritative nameserver for `example.com`.
- `6:` The resolver sends the request to the Authoritative nameserver for the domain it was instructed to look up.
- `7:` The Authoritative nameserver responds with the `A` record (IP Address) for the domain `example.com`.
- `8:` The DNS resolver then responds to the client that made the initial request for resolution.

As we have shown, the steps that a recursive resolver follows during a typical resolution process are simple and correspond to a hierarchical search of a distributed database. Also, we note that DNS recursive resolvers utilize a caching mechanism as part of the DNS protocol. That cache allows the resolver to skip the steps to obtain the answers to some resources if an active record that can provide the reply is in the resolver's cache.

2.1.2 DNS Records and Types

After introducing the various elements of the Domain Name System's structure, we now briefly describe how a DNS packet is structured and what different types of information it can provide. We briefly explain the relevant information in each DNS packet and specific query types that provide us and our dataset with more information than a typical A-type query.

In order to put these various types of information into context, we need first to discuss the structure of a DNS record. Each DNS record can contain the following sections, **Header, Question, Records**. The record section is partitioned into separate sections named **Answer, Authoritative, Additional**.

Each DNS packet contains a header section. The information in the packet's header section is crucial for the operation of the entire DNS and the efficient routing of the packets. For that purpose, the header contains information codes and flags that designate various technical properties about the DNS packet, such as the size, if it is a response or a request, if the server provides an authoritative response or not, and many more. Here we will briefly discuss the labels relevant to our study and this thesis. The portion of the DNS header with the highest relevance to us relates to the `RCODE`, which stands for the response code written to the package by the DNS server replying. This information is important to us because it contains information as to the status of the query and informs us if the resolution has failed. One of the primary reasons for a query failure in our system is an `NXDOMAIN`, which is the `RCODE` that designates that the domain name does not exist.

The other portion of the DNS header, which provides us with information relevant to our data collection system, is the sequence of flags that describe the contents of the packet and inform us as to what the contents of the packet describe and what type of response we have received. Other important identifiers are the query type `QTYPE`, which indicates the type of response the client requests, and the record type or `RTYPE`, which indicates the type of record provided in one of the answer fields by the DNS server. Various record

types in DNS support additional types of information to be recorded. The most important record type is the `A` type record, which encodes the IPv4 address of a domain name. Other important types of records include `AAAA`, which provides the IPv6 address of a domain name. Another popular DNS record type is the `CNAME` which forwards a domain to another domain record but does not provide an IP address. This type of record is also referred to as a *Canonical name* and is not a type that the client queries for, but DNS authorities can use it to point to another domain zone. The other type of record that is informative for the DNS servers and the DNS resolver but that the average user has little knowledge of is the `NS` record, which stores the DNS nameserver for a particular domain. The recursion process generates `NS` records as the recursive attempts to locate the Authoritative nameserver for the user query.

Further types of records support other uses and protocols outside the typical user's perception of a web browsing session. One of these DNS record types is the `MX` record, which informs the client of the email server domain address and supports global email. Another record type is the `TXT` record that allows the zone administrator to store text in the DNS zone. The text can be anything encoded inside a DNS packet, but nowadays, it is mostly utilized for email security.

DNS also supports more record types that are not widely utilized in day-to-day operations but can reveal a great deal of information about a DNS zone and how it operates. Information that, when recorded, helps analyze topics such as domain name transfer of ownership, as shown in subsection 4.3.1. One such record type that our system queries for is `SOA` record type, which stores administrative information about a domain and is essential for the process of a zone transfer where a DNS nameserver transmits data to a secondary DNS nameserver. Finally, we have the DNS record type `PTR`, for which systems or users can query to obtain the domain associated with an IP address in a process called a reverse query. Our system collects, processes, and stores the information included in these data types when available, except for PTR. We do not query for PTR records as this is a reverse

lookup process from IP to the domain, and our interest is in recording and mapping the domain landscape and its associations. Specifically, we send queries that request replies for `A, AAAA, MX, TXT, SOA` records and collect and process every type that comes through our recursive resolvers, including all of the `NS` and `CNAME` type records generated as part of the lookup process.

2.1.3 DNS vantage points

As we have now established some basic knowledge about the operation of the Domain Name System and the kinds of information that it encodes, we will discuss one more topic that is important to be well understood before we move into the chapters that detail the operation of our system and compare it with other DNS data solutions. We need to discuss the various proposed and implemented types of DNS data collection for these comparisons. Dr. Weimer published the initial work describing the motivation and methods for DNS data collection more than fifteen years ago[1]. Since then, the security community has utilized datasets captured in that fashion as the basis of what we call a Passive DNS dataset. In order to understand the design decisions that we present in chapter 3, we need to have a brief discussion and explanation of the available approaches for DNS data collection.

Figure 2.1 notes the two popular collection points of Passive DNS data: *below the recursive* and *above the recursive*; each approach yields a different kind of dataset. The first point of DNS data capture is below the recursive. If our capture point in the network is just before the recursive resolver inside the local network, our dataset will contain all requests and replies from all devices on the network to the DNS recursive. That type of visibility records each request for resolution from every one of the clients with their full IP address recorded.

The other popular collection point for DNS data is the point just above the recursive. This collection point can record every packet that the recursive resolver sends as part of the DNS resolution process and every reply the resolver receives. From this vantage point,

we have no visibility into the IP addresses of the clients that submitted each query to the recursive, but on the other hand, we record a number of packets that contain valuable information regarding the DNS hierarchy, such as `NS` type DNS records, that are sent to the recursive as part of the resolution process. That is why this is the observation point usually utilized for Passive DNS data collection, as it retains the benefits of having better visibility into the DNS hierarchy and is also more efficient than the alternative because it does not record every DNS request and reply in the internal network, which in many cases includes a large number of cached replies, and thus, redundant information for the types of research we are targeting with our system *Thales*.

Passive DNS datasets require network monitoring. For the dataset to be of any representative value, the network should contain a certain amount of traffic representing a proportion of the global or regional DNS traffic. A number of academic researchers have utilized datasets captured from university campuses that tend to include many categories of users, such as residential for student housing, business users in administrative positions, and more. Others have obtained passive DNS data from a commercial provider. In this short segment, we present some of the actual costs of obtaining passive DNS data from commercial providers. Unfortunately, many providers do not publicly disclose their pricing policies and require contacting their sales team or going through an online subscription, and of course, the costs depend on the number of queries made, with full access being prohibitively expensive for many researchers. For example, the following is the pricing for a commercial provider of passive DNS data that provides three service tiers. The first tier of service is quoted as "PRO" and provides access to a passive DNS API with daily limits of 2,500 daily and 25,000 monthly queries for a yearly fee of $10,000. The second tier is labeled "ENTERPRISE" and provides access with API limits of 25,000 daily and 250,000 monthly queries for a cost of $60,000 per year. The final option is the "ENTERPRISE +" option, which provides unlimited access to the query API and "real-time data

feeds" for a yearly cost of $120,000[1]. This is only a sample pricing for access to passive DNS data; unfortunately, most major providers do not offer public pricing and require a sales inquiry[2], an additional barrier that security community members must clear in order to obtain quality DNS datasets. Furthermore, all commercial providers require signing an agreement that limits the information that can be disclosed and usually prohibits sharing of the purchased data, which is essential for study repeatability. This information is always subject to change, but we include it in order to put into context some of the actual costs and limitations of obtaining commercial passive DNS datasets and what drove us to contribute our very own DNS dataset, which is well documented, free, and open to the academic community.

2.2 Previous Work

As we have already mentioned in the Introduction, the first one to describe and advocate for the use of passive DNS data was Weimer et al. [1] as a method that network operators could use to investigate security events in their environments. In this section, we discuss the kinds of work that influenced and shaped our approach to studying such a complex issue as the creation of Active DNS datasets. These datasets were meant to have applications in the cybersecurity space without the issues of user data and privacy considerations as part of the design. So we were inspired and guided by what others had done before us in an effort to move the space further ahead. Then we document the works that inspired us to find and document other use cases for our dataset.

2.2.1 DNS collection and measurements

After the introduction of Passive DNS, Zdrnja et al.'s work [12] was the first that presented how passive DNS data analysis can identify and study security breaches. Subsequently, research using DNS datasets, such as Notos [13] and Exposure [5], began to impact security

[1] https://www.spamhaus.com/product/passive-dns/
[2] https://www.farsightsecurity.com/order-services/

by introducing the concept of reputation to DNS tree nodes through statistical analysis of resolved passive DNS traffic characteristics. Numerous researchers have since utilized proprietary passive DNS data to develop security systems for identifying internet abuse [6, 7, 14, 15, 16, 17].

Passive DNS has become an invaluable tool for network operators and security researchers to combat internet abuse. Our Active DNS project aims to offer open access to high-quality DNS datasets, similar to passive DNS, while addressing concerns about personal information, financial costs, and legal barriers for repeatable DNS research.

There have been numerous attempts by both private companies and government organizations to establish passive DNS repositories. However, the fees associated with these commercial offerings can be prohibitive for researchers and network administrators. Perhaps the most successful has been `passiveDNS[.]cn`, which was swiftly dismissed as an untrustworthy source of DNS information due to its place of origin. The logic behind this development seems clear to us: the Chinese operators simply collected DNS records that had already been censored by their egress sensors. In our project, we do not censor the views of the recursive DNS servers that *Thales* uses to resolve the seed domain names on a daily basis. Regarding active scanning efforts, the majority have been carried out by industry. However, recent research from the academic community [18] has emerged that enables researchers to scan the entire IPv4 space and use the results for open security research. This is the work that most closely resembles our proposed *Thales* collection system, but with the significant distinction, Censys was created to scan the IP space, specifically IPv4, rather than the domain name space. Therefore, while researchers may be able to locate some DNS logs within this extensive public project, our work supplements Censys and is also intended to address DNS scanning and provides a well-documented and already, in only a few years, well-utilized Active DNS data as we document in section 3.5. Several works [19, 20] offer slight variations on our data collection model for active queries and primarily document the scaling issues actively scanning for DNS data brings. They also

offer new ideas on how other researchers can utilize actively collected DNS data.

2.2.2 Technical Support Scams

Miramirkhani et al. [21] performed the first work we know of that defined the Technical Support Scam by focusing on scams delivered via malvertising channels and interacting with scammers to identify their *modus operandi*. As we will focus on the dataset enrichment methods we contributed to this project, we limit our discussion of related work to those closest to our contributions. Specifically, we focus on existing studies that cluster abusive or malicious infrastructure. As we have mentioned in the related work to Active DNS, several studies uncoverer previously unknown infrastructure and link it through DNS datasets to known malicious infrastructure. This discovery is accomplished with assistance from clustering algorithms to better identify potential campaigns based on infrastructure information. These studies use different types of information in order to uncover these campaigns, such as URL information [22], or IP and DNS infrastructure [4, 5], while others focus on the content of the pages [23]. Hierarchical clustering techniques have also been used effectively in a number of published works [24, 25, 26]. We expect to continue observing new studies utilizing these methods to identify malicious resources and the infrastructure that hosts them to tackle emerging threats such as more sophisticated or targeted TSS scams.

2.2.3 DNS Extension EDNS Client Subnet

Finally, we will discuss our work on EDNS client subnet (ECS) and other work that studied information leaks from Internet protocols. The interaction of DNS and anonymity networks has been well studied. Krishnan et al. [27] have shown how DNS prefetching can leak information regarding users' activity online to the degree that information regarding web searches can be inferred by simply logging a browser's resolution requests.

Zhao et al. [28] perform a deep analysis on each step of a domain name resolution pro-

cess, showing information that can be inferred from users' private data by only looking at public data. They also propose a simple range query scheme that can be used to protect the user. In the same context, Guha and Francis [29] describes an attack by passively monitoring DNS-related traffic that can provide a variety of information about a user, including location, habits, and commute patterns. Moreover, Bortzmeyer, in RFC 7626 [30] attempts to enumerate the attacks and privacy implications, aggregated into six different categories, made possible only using DNS; they concluded their work with several security considerations on the matter. Lastly, Bortzmeyer also describes potential privacy issues and attacks via monitoring DNS traffic and examining the domain names included in packets, which can be solved by implementing RFC 7816 [31]. On the other hand, ECS is a relatively new technology and is motivated by the performance challenges related to the growing use of public recursives [32], as discussed by Huang et al. [33].

The guidelines in the corresponding RFC [9] provide a general outline of how ECS should be deployed and how ECS-enabled servers should be operated. Streibelt et al. [34] demonstrate how one could utilize ECS-enabled authorities to uncover details about an ECS-enabled zone's infrastructure and the owner using it. Recently, Al-Dalky et al. [35] studied a more specific aspect of ECS that has to do with the caching behavior of DNS resolvers when it comes to ECS-enabled answers and the variety of different caching behaviors that can be examined. Our work focuses on a long-term study of the behavior and adoption of ECS. Lastly, Otto et al. [36] have measured how adopting ECS can increase the accuracy with which authorities can identify a client's geographic location and provide better content delivery.

CHAPTER 3

ACTIVE COLLECTION OF DNS DATA

3.1 Motivation

The Domain Name System (DNS) is a fundamental component of the Internet. Especially considering that the Internet is still expanding both in active users and services being offered, which inevitably means that the amount of traffic that traverses each network is ever-growing. Even though there have been changes over the years to how the Internet has grown and how its component applications are being utilized, it is still the case that most network communications on the Internet start with a DNS lookup, which maps a domain name to a corresponding set of IP addresses, or other resources to contact. It has also been well documented that cyber criminals frequently leverage DNS to be able to provide high levels of network agility for their illicit operations. As an example, malware still relies on DNS to locate its command-and-control (C&C) servers. Such servers are used to send commands from the attacker, exfiltrate secret information, and send malware updates.

Initially, network operators relied on static blocklists in order to detect and block DNS queries to domains associated with malware operations and stop the abuse. However, static blocklists can no longer keep pace with modern threats' quantity or network agility. That's why the research community has proposed analytical systems [4, 5, 6, 7] that utilize passively captured DNS data to shorten the response time necessary to react to new threats and keep networks secure.

However, such datasets are difficult to find, challenging to collect, and often require restrictive legal agreements. Furthermore, we have already noticed a shift towards encrypted DNS communications, especially in the case of commercial open recursives. This situation likely arose from the growing awareness of users about their privacy but has led to a situa-

tion where established detection methods that rely on processing network-level information are getting less and less effective at detecting malicious behavior. All these obstacles can make further innovation in the network security space difficult. It also creates a divide between researchers with access to such datasets and researchers without access to large amounts of funds that can enable them to purchase a commercially available dataset or without access to a sizable network to monitor and collect their own limited dataset. This situation ends up limiting the repeatability of research in the space and also makes it more difficult for new researchers to explore the value that DNS datasets can offer in the context of security research.

The lack of open and freely available DNS datasets, thus, puts the security community at a disadvantage because they lack access to the datasets describing a critical component and infrastructure used by adversaries on the Internet. Clearly, the security community is in need of open, freely available DNS datasets that can help increase situational awareness around modern threats. This is further illustrated by the fact that most modern threats rely on DNS for their illicit activities.

This chapter aims to provide a solution to this issue by introducing and documenting the concept of Active DNS datasets while we discuss and present a new large-scale distributed collection system we call *Thales*, which is able to systematically query and collect large volumes of active DNS data. The output of this system is a distilled dataset that can be easily shared with the security community. Our distributed collection system *Thales* has been reliably active for more than six years and has collected many terabytes of DNS data while causing only a handful of abuse complaints. The system has evolved and has been upgraded to keep it in line with modern network demands and to make its operation more consistent and easy to monitor while also ensuring that the resulting dataset is a complete and accurate representation of the activity in the DNS.

3.2 System Overview

In this chapter, we introduce *Thales*, the name of our Active DNS collection system. We will begin by discussing the network and system infrastructure necessary to collect active DNS datasets systematically and reliably. Then, we will discuss the details of the domain names that compile the daily seed for Thales. The section will be concluded by discussing the long-term measurement behind the collected active DNS datasets.

The reliable collection of DNS data is far from easy. Thales was designed to retain high availability, efficiency, and scalability levels. The goal of Thales is clear; the generation of active DNS datasets that will provide periodic snapshots of the DNS infrastructure several times per day. These datasets will enable the security community to construct a timeline of the evolution of threats on the broader Internet.

Our system, Active-DNS, comprises two main modules, as seen in Figure 3.1: (a) the traffic generator and (b) the data collector. The first is responsible for generating large numbers of DNS queries using a list of seed domain names as input to the system. The second module is responsible for collecting the network traffic and guiding these raw DNS datasets for further processing (i.e., data deduplication). Since 2019 the implementation of each of the modules has changed, but the overall data flow and results are similar. The changes were made to improve the reliability of the system as well as increase collection capabilities.

3.2.1 Query Generation Infrastructure

In order to achieve high availability, redundant systems are used to generate traffic. Linux containers (LXC) [37] are set up across several physical systems, creating a DNS scanning cluster of 30 LXC containers. Each LXC contains its own local recursive software [1] and is assigned a job where a subset of the overall daily seed domain names will have to be resolved by a particular container. High efficiency is achieved by increasing the rate of

[1] We used the Unbound (https://www.unbound.net/) recursive software in every LXC container.

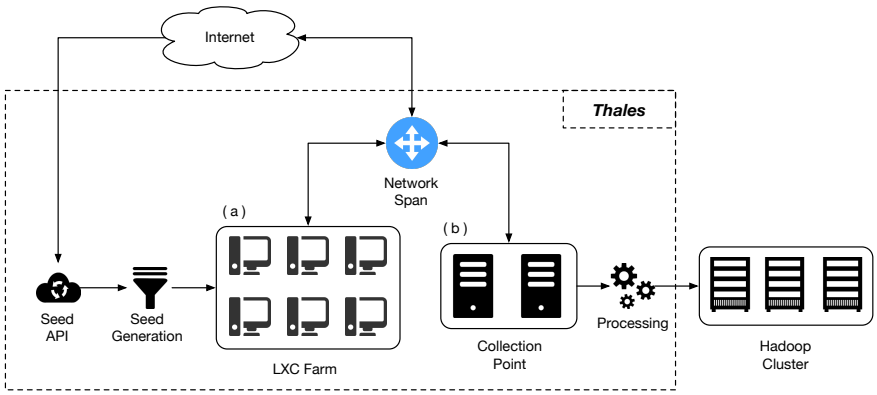


Figure 3.1: The Seed API is responsible for collecting the seed domains from various sources, and the Seed Generation reduces them to a list of unique domains. The container Farm corresponds to the query generator which is connected to the Internet through a Network Span. That, in turn, is sending traffic to the Collection Point from where data is being reduced and stored for the long term on our Hadoop Cluster.

DNS resolution requests (a.k.a. queries per second) that can be handled by the recursive in the LXC container. However, just increasing the resources of the LXC container will not suffice for the container to handle a large enough number of DNS requests. This is because the local recursive in the LXC is bounded by the maximum number of ports that can be used for UDP sockets. This means that the number of requests that a host can send has to be limited to the number of available concurrent ports that the local recursive (in the LXC container) can handle.

At any given time, a container could theoretically handle up to 64,512 ($2^{15} - 1024$) sockets per IP address – and, therefore, 64,512 UDP query packets in transit. The LXC containers support custom network interfaces, which support assigning a different IP address to each container. More specifically, we use 30 contiguous IPs from an assigned IP block of 63 available addresses (/26). Thus, they can send and receive up to $30 \times 64,512 \approx 2^{21}$ simultaneous DNS resolution requests from the infrastructure. These results are achieved by deploying the containers on two physical systems. These two systems have 64 processing cores and 164GB of RAM. It is worth pointing out that using LXC containers allows us to scale the infrastructure horizontally by simply adding more systems to our scanning

cluster.

As we have also mentioned, *Thales* was upgraded in 2019. One of the main changes to the Active DNS collection system was the introduction of a more flexible container management system based on Docker containers after they started supporting better VLAN management options while also improving the performance and reliability of the system by introducing automated monitoring systems into the system architecture. By utilizing, the more modern Docker containers, we can more easily deploy more, self-contained workers to expand our collection capability or introduce new features.

3.2.2 Domain Seed

Before Active-DNS can begin scanning the domain name system, it has to be provided with a list of domain names that will act as candidates for resolutions. We will refer to these domain names as the *seed* for Thales. The seed is an aggregation of publicly accessible sources of domain names and URLs that we have been collecting for several years. These include but are not limited to Public Blocklists, the Alexa list, the Common Crawl project, and various Top Level Domain (TLD) zone files.

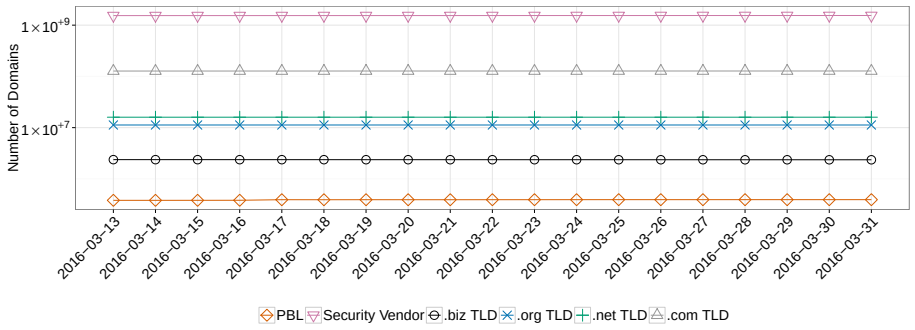


Figure 3.2: Number of domains over time per seed input. The security vendor list contains about 1.5 billion domains, and from the TLDs com is obviously the largest one, with about 127 million domains.

More specifically, we are using the zone files published daily by the administrators of the zones for com, net, biz, and org. In Figure 3.2, we present the number of domains ob-

tained by each zone file. Because of the relative number of small daily changes, compared to the size of the zone files, the daily changes are not that apparent in Figure 3.2. We note that the number of domains obtained by zone files changes as new domains get registered, and old ones expire (and get removed from the zone). In Active-DNS, we input these zone files that we collect daily to our domain seed. This way, our seed includes the current state of each zone every day.

We also add the entire Alexa [38] list of popular domains to the domain seed. This ability provides us with a large number of domains that would most likely be queried in a network by users.

To capture domains that might not be available in one of the zone files, we built a crawler that collects and parses domains seen in the Common-Crawl dataset [39]. The Common-Crawl dataset is an open repository of web crawl data that offers large volumes of crawled pages to anyone. We used components (i.e., URLs, HTML code) from the common crawl dataset to extract only the domains of the pages visited. Due to the size of even the Common-Crawl "metadata section" from the common crawl, we are still using the data published last September 2015 and will start updating that list regularly. Because the common crawl data is published in monthly releases, the domain list that we extract from it and use in our seed list remains the same between updates.

A different list of data we utilize in our domain seed is a feed of "interesting" domains provided to us by a security company. This feed provides us with domains that have been observed to engage in forms of potentially malicious Internet activity. Because the feed constantly provides us with new domain names, we gather all new information and append it to the existing list of interesting domains. We push the updated list to our collection infrastructure daily. The feed provides us with tens of thousands of new domains each day, making this list one of the fastest-growing lists we use. Also, we use a collection of public blocklist data to provide our data with interesting hand-curated domains that originate from malicious activity. More specifically, the public blocklists we employ are: Abuse.ch [40],

Malware DL [41], Blackhole DNS [42], sagadc [43], hphosts [44], SANS [45] and it-mate [46]. We aggregate these lists daily and input them into our domain seed by replacing the old list.

3.2.3 Data Collection

In this section, we will discuss some of the unique details about how our DNS traffic capture system was implemented before its eventual update in 2020. Some more references about the general issues surrounding the topic of DNS data collection are documented in subsection 2.1.3. The requests submitted by Active-DNS were initially collected at two vantage points. The first point is on the container that has submitted the resolution request for a given domain name. The second one is at the `SPAN` of the switch that routes traffic for all of our containers. As mentioned earlier, we are utilizing several IP addresses from several local virtual LANs (VLAN). These VLANs have been "trunked" to a single 1Gbit interface on a host that collects all port 53 UDP traffic. We are collecting traffic at both points for redundancy and verification of correctness for the daily active DNS datasets.

3.2.4 Parsing and Deduplication

Capturing network traffic resulted (on average) in a massive 1.7TB of raw data in *packet capture* format (pcap). This data was transferred to a local Hadoop cluster composed of more than 20 data nodes, by that time. The Hadoop cluster was responsible for parsing the pcap files, deduplicating the resource records (RRs), and converting the RRs into meaningful DNS tuples of the following format: `(date, QNAME, QTYPE, RDATA, TTL, authorities, count)` as seen in Figure 3.3. Deduplication is a critical and important step since many responses we collect remain the same throughout the day. That means we only store the DNS tuples we referenced as unique per day for each tuple. Thus, after removing duplicate RRs, we were left (on average) with approximately 85GB of processed data daily.

3.2.5 Storage Schema and considerations

As previously mentioned, operating Active-DNS generates more than a terabyte of packet capture data daily. In order to be able to store and utilize the data efficiently, we have to process the network captures into some form of storage. Initially, we attempted to parse the collected data and keep the entirety of every DNS packet we received. Unfortunately, this proved impossible within a matter of weeks of continuous data collection and storage of the data. We then took a detailed look over the dataset that we had been collecting. We tried to identify the necessary information we need to preserve while removing the types of information that a typical DNS packet contains, which would be redundant to store. The decision of what information we discarded came down to the scope of the intended use cases and the unique considerations of operating an Active scanning infrastructure. Specifically, disregard information from the lower layers that are also captured in the process. So we do not parse any information about Medium Access Control (MAC) or any IP or UDP data since they are not the focus of our work and account for a significant portion of the initial capture storage.

The obvious next step in our process is to process the DNS replies we receive as part of Active-DNS. Because all DNS packets contain information provided by the client in the reply, we can safely ignore that information as part of our long-term storage schema since we provide that information. Thus, it is entirely predictable and under our control and provides low informational value for the types of research we have developed this system. There is also the consideration that keeping all of this data long-term would be redundant, thus requiring more expensive storage. At the same time, it would reveal information about our internal network layout to the people with whom we share our data. Considering that we built Active-DNS with the explicit purpose of being able to share our datasets, it was an easy choice to opt not to retain client information in the storage schema.

After considering all our limitations and design goals, we also decided to apply a deduplication step. That became necessary because of the high volume of queries we send and

because we try to query every domain in our seed list multiple times daily. This results in multiple DNS replies that are identical from the standpoint of which fields in the DNS packet remain static from query to query for the same domain. So we store only the unique information for each queried domain per day.

As we have established, the processing and deduplication of the data were necessary to run the program and achieve our research goals. The way we initially implemented the parsing was by utilizing our Hadoop cluster. This allowed us to store the processed information in HDFS, a high-level file system that allows easy data replication across a cluster of nodes while also being easy to work with from within the Hadoop paradigm. Our process ingests packet capture files and runs them through an application written based on the Map-Reduce programming paradigm that works well for distributed systems.

Initially, the Map stage parses the network capture information and keeps only the fields inside the DNS packet we intend to store. This step results in a tuple of date and time, query name, query type, response data, TTL, and authority IP.


```
{
    "date": "20160303",
    "qname": "0746jiaoyou.com.",
    "qtype": 1,
    "rdata": "61.151.239.202",
    "ttl": 3600,
    "authority_ips": "58.216.26.232,120.52.19.142",
    "count": 5,
    "hours": 32778,
    "sensor": "active-dns"
}
```


Figure 3.3: A sample record from our original dataset that shows the data fields that are stored. The `authority IPs` field represents the authoritative nameservers that replied for this domain name and the `hours` variable captures the hour of the day that this record was seen in a 24-bit integer.

In this step, each DNS reply is a tuple of the mentioned data fields. We then utilize the reduce step of the paradigm to deduplicate the DNS information. The final output is a unique tuple for each queried domain per day, including the date of scanning and the information we extracted in the previous step flattened so that each domain occupies only

one tuple per day. For that, we have to modify how we store the data. Instead of keeping the exact time and date of the query, we simply reduce the resolution of the date field to contain only the resolution up to the date. `RDATA` is stored as a string of the response contained in the response data in the capture and is described by the `rtype` as to what type of information it provides. Because multiple authoritative name servers can serve domains, as is good practice, we modify the authority IPs fields to contain a list of all the authoritative IPs that have given us this exact answer for that day. We also calculate how many tuples were flattened and add it as a count field. Finally, we add some information about the sensor that sent the query.

The eventual output is in the Avro format and was stored as mentioned on our Hadoop cluster. The average final file size for a typical day is around 75GB to 80GB. This makes utilizing our collected data more efficient as we can utilize the programming capabilities of Hadoop to run applications that will generate a result or simply process them for storage on a database-like system such as Hbase of Apache Hue.

3.2.6 Thales 2.0

After several years of sucessful operation, *Thales* was upgraded in 2019. This upgrade took advantage of newly available technologies and allowed Active DNS to continue to be an asset to security community [47].

3.3 System Reliability

The original Active-DNS publication presented initial measurements from the dataset that had been collected for a little less than six months during the time of the original publication of this paper. Over six months of operation, Active-DNS was able to identify approximately 10,714,784 unique IP addresses, 199,110,841 unique domain names and 662,319,389 unique RRs per day. Figure 3.4 shows the average distribution of IP addresses, domain names, and RRs per day from October 5th to March 3rd, 2016.

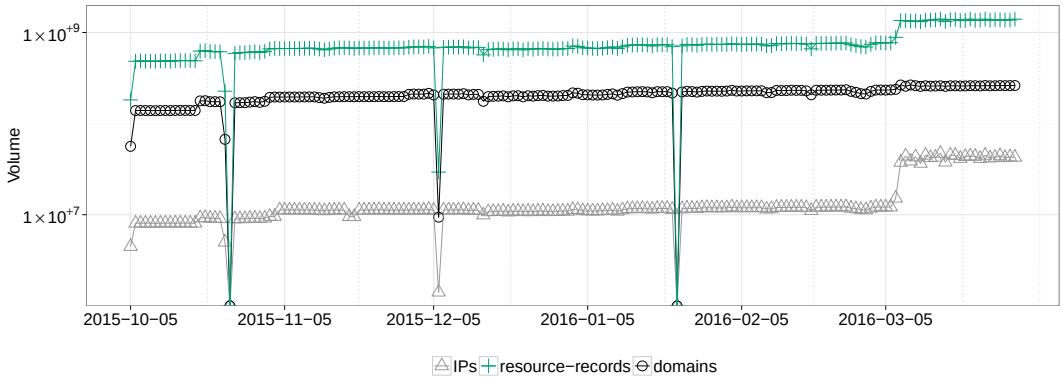


Figure 3.4: Volumes of IPs, resource records, and domains observed with Active-DNS. March 7th was the day when we started querying for the QTYPEs: SOA, AAAA, TXT, and MX. There were two full outages on October 25, 2015, and January 23, 2016. On December 6, 2015, we had an outage that lasted for most of the day, but we were able to recover the system later in the day.

During these months, we experienced two minor outages. The first was when the system was initially set up because of an update that was not rolled out correctly and caused the system to go offline. Therefore, there is no data available for the day of October 25, 2015, and our operational policy has been updated to avoid future interruption. On January 23, 2016, our campus data center was undergoing maintenance for the cooling infrastructure, which caused a temporary shutdown of all our systems that were outside our immediate control.

Active DNS can now mitigate such cases because of all the work we have made to render the system portable, allowing us to move it to another location within a day's prior notice. This was an active area of improvement as it also offers capabilities such as scanning from specific locations, etc. Also, on December 6, 2015, early in the day, we encountered a hardware failure on our system that was detected early in the morning. We recovered the system and performed a check of all important components by the same afternoon. After the system check, we immediately restarted the collection process, but there was not enough time in the day to go through the entirety of our seed list. The significant dip in the data depicts this. This incident was not a total outage since we collected some data

for the day. Over the years, the system's stability has been similar to the performance we maintained during our first year of operation. Most outages are resolved within a day and have only resulted in a minor data loss. After some targeted updates to various system components of *Thales* we managed to successfully operate a large-scale network collection system that has been scanning for more than six years and has thus amassed an archive of all these years.

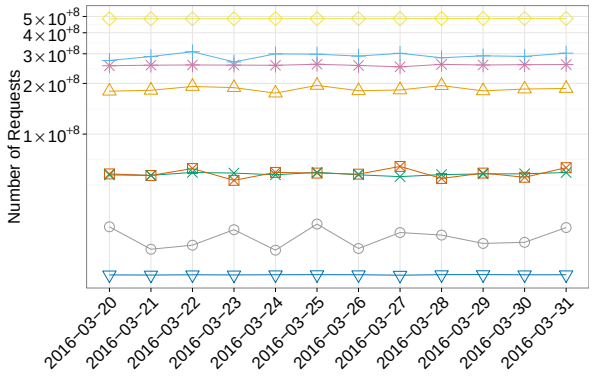
3.4 Comparison of Active and Passive DNS datasets

Passive DNS has been an invaluable weapon in the community's arsenal for research combating malware, botnets, and malicious actors [4, 6, 7, 48, 49]. Passive DNS, though, is rare, difficult to obtain, and often comes with restrictive legal clauses (i.e., Non Disclosure Agreements). At the same time, laws and regulations against personally identifiable information (PII), the high financial cost of the passive collection, and storage infrastructure are some of several reasons **that make passive DNS cumbersome**. The primary goal for the Active DNS dataset is to reduce the barrier to (repeatable) security research on DNS or to use of DNS data. This section shows how active DNS relates to and contrasts with traditional passive DNS. We will see that, while not a true replacement for passive DNS, Active DNS is able to create DNS datasets that, in many cases, contain an order of magnitude more domain names and IP addresses.

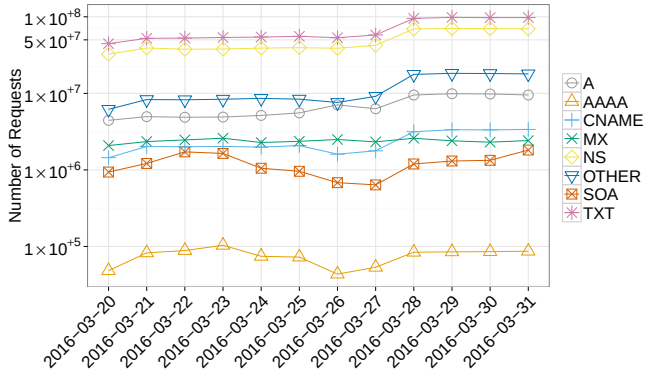
3.4.1 Documenting Datasets

We will first discuss how we obtain our passive DNS datasets. Our passive DNS dataset consists of traffic collected at our university network. The collection point is both *below* and *above the recursive*. This means that we collect the responses on both paths; (1) between the (anonymized) clients and the local recursives and (2) between the local recursives and the upper layers of the DNS hierarchy (i.e., name servers, top-level domains, etc.). For the active and passive DNS comparison, we decided to utilize datasets collected during the

entire month of March 2016.



(a) Active DNS.



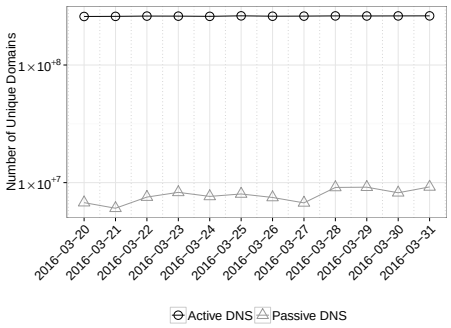
(b) Passive DNS.

Figure 3.5: The distribution of different query types (QTYPE) in the Active (left) and passive (right) DNS datasets. The Active DNS dataset is sustaining the same volume of records per day, by design, whereas the passive DNS dataset is fluctuating more over time. Note the growth after March 28, when the Spring Break was over, and the Institute was operating at full capacity again.

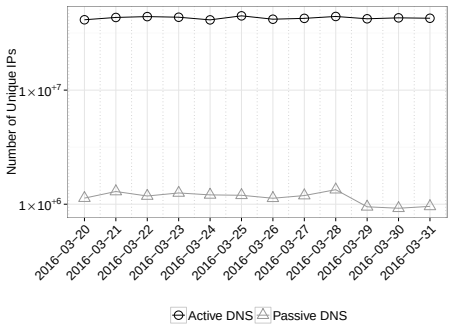
Figure 3.5 and Figure 3.6 show eight plots of the distribution of records in both our Active and passive DNS datasets. Note that all plots are log-scale for the y-axis. As we can see, the Active DNS dataset is stable with few fluctuations compared to passive DNS. This is primarily an artifact of the collection technique since the daily changes in the domain seed is minimal. On the other hand, the passive DNS dataset is primarily driven by the behavior of the users on the local network, which may vary on weekends, holidays, and during specific periods such as exams. This explains the sudden increase in traffic for passive DNS since our campus network experienced a reduction in traffic from March 21^{st}

until March 25^{th} during spring break. Therefore, Figure 3.6c shows an increase to more than double the unique resource records (RRs) identified per day after Monday, March 28^{th}, when the spring break ended.

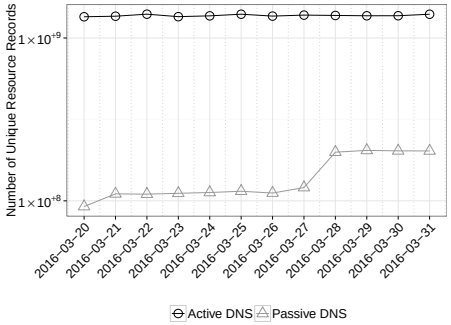
It is worth noting that *Thales* can generate an order of magnitude more unique domain names, IP addresses, and `RDATA` in the Active DNS dataset (see Figure 3.6, subfigures a to e), compared to the passive DNS data, collected at a large university. This means that in actual DNS records, the Active DNS dataset is more than comparable to the passive DNS that someone can collect in a large university. Now, as we can see from Figure 3.6, (f), Active DNS cannot create as dense graphs of resource records as someone would expect to find in passive DNS data. This is somewhat expected, as in Active DNS, Thales is scanning all possible domain names that can be seen in our public sources. This inevitably will include rare domain names, and in the context of a graph compiled by RRs, they will form islands. While not necessarily bad, we advise researchers to take cautionary sanity steps when they utilize the Active DNS data for spectral processes. With the diversity of the various query record types (`QTYPEs`), we are able to identify the two different datasets compared can be seen in Figure 3.5a and Figure 3.5b. Although there is a significant difference regarding the volume of the records available, on average, the visibility is very similar since we are collecting the most popular `QTYPEs` when querying for the active DNS datasets. However, as seen in Figure 3.7, Active DNS collects consistently more response types (`RTYPEs`) per domain for the same week and domains compared to passive DNS with a median of 3 `RTYPEs` in passive DNS and 5 `RTYPEs` in active DNS. This is because Active DNS queries for all of its seed `QTYPEs` per day and per domain while in passive DNS, the `QTYPEs` and therefore `RTYPEs` have to be generated by organic traffic (e.g., Active DNS will scan for MX records (`QTYPE` 15) for all of its seed domains per day while in passive DNS this activity may not be observed consistently).



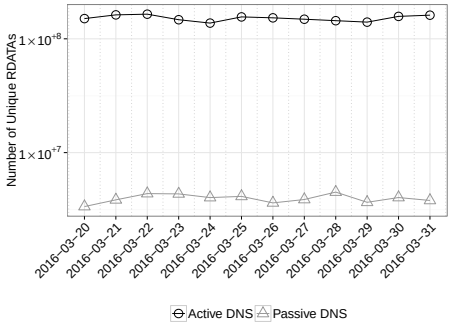
(a) Unique domain names per day.



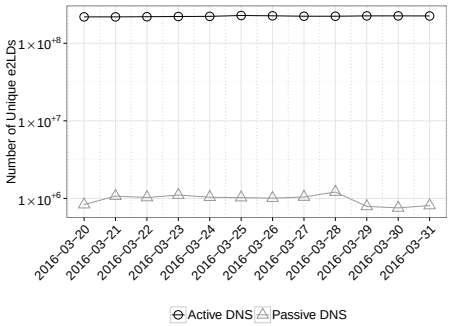
(b) Unique IP addresses per day.



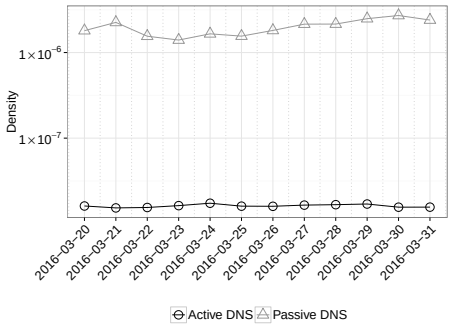
(c) Unique resource records (RR) per day.



(d) Unique responses (RDATA) per day.



(e) Unique, effective second-level domain names per day.



(f) The density of the Resource Records graph in the Active and passive DNS dataset.

Figure 3.6: The distribution of different records in our Active and passive DNS datasets. The plots show that ACTIVE DNS can generate orders of magnitude more data than the passive DNS collection engine (Figures a to e) and is much more diverse (Figure f).

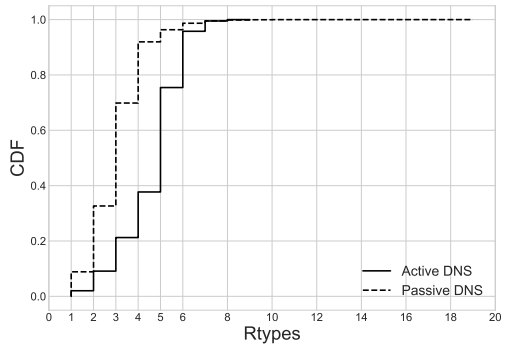


Figure 3.7: Unique RTYPEs per domain CDF.

3.4.2 Passive and Active DNS data applications

We do not intend Active DNS to replace Passive DNS in all security research applications. Here we explicitly discuss some of the advantages and weaknesses of each. Understanding when it is appropriate to use passive or Active DNS, the conclusions we can draw from each dataset, and their limitations are fundamental to the success of projects built on DNS datasets. We conclude this section with a taxonomy of research projects we have identified that successfully utilize Active DNS, highlighting the impact of our work.

Coverage Breadth: The generated queries are the main factor that limits the breadth of coverage for both Active and passive DNS datasets. In the case of passive DNS, these queries are generated organically by the users and system of the monitored network. Web browsing, automatic updates, security appliances, and email are just some of the activities that result in DNS requests. The size, security posture, and type of network will all play a role in what domains resolutions are included in the dataset. A university's research network will query for a different subset of second-level domains than a network consisting of primarily residential customers will.

On the other hand, in Active DNS, the breadth of queries we perform results directly from the seed lists we aggregate. *Thales* queries every domain in the seed every day. When available, zonefiles make an excellent core component of the seed list because they allow us to enumerate the domains of the respective TLD, irrespective of popularity or usage.

However, not all TLDs make their complete zones public. For example, Thales will not query for every `.RU` domain, only those that also appear on one of our additional seed lists because we cannot access the `.RU` zone files.

With this in mind, Active DNS is better suited for mapping the infrastructure of second-level domains in most cases because if provides a large breath. chapter 4 highlights this point as we could identify related TSS domains before the scammers used them. However, researchers with access to passive DNS can continue successfully using it to map infrastructure, provided the domain in question is queried for by a system on a monitored network. In certain specific instances, passive DNS may be preferable, for example, when the second-level domain does not appear in our seed list. Combining both will provide the largest available breadth of coverage for infrastructure mapping.

Coverage Depth: When comparing the depth of coverage of passive vs. Active DNS, we examine the coverage subdomains beyond the second-level domain and the diversity of RTYPEs observed. For subdomain depth, Active DNS focuses on the second-level domains. In this way, it sacrifices depth for breadth, taking only a surface-level snapshot of each second-level domain. This disparity becomes more noticeable when considering disposable domains and dynamic DNS, which can contain rich behavior at the third or fourth-level domain. With this in mind, tasks such as determining the network footprint of a specific organization utilizing a small number of second-level domains is best performed using Passive DNS datasets (provided they have sufficient coverage). The second component of coverage depth is the number of unique resource records obtained for a specific domain. On average, Active DNS provides more `RTYPES` per domain, which we discussed in the previous subsection.

Temporal Considerations: Active DNS queries all domains systematically, with several hours between successive resolutions. The temporal patterns of passive DNS arise organically. In many cases, this will result in passive DNS datasets having more frequent resolutions for popular domains in the network (especially those with short TTLs) and more

specific coverage of less popular domains within the capturing network.

Human factors play a key role in passive DNS, while the periodicity with which Active DNS performs queries is independent of our cultural calendars. Work hours, weekends, and holidays all affect the volume and diversity of DNS activity, particularly in corporate and university networks.

For example, Active DNS cannot capture a full picture of fast-fluxing domains, which may change how they resolve much faster than the multi-hour timescale on which Thales queries. The same is true for malicious domains registered and mitigated within a 24-hour time window before being added to our seed list.

Behavioral Considerations: Passive DNS datasets can provide additional context surrounding a domain that is not captured in Active DNS. For example, determining the relative popularity of two domains is not possible using only Active DNS but can be estimated using passive DNS, at least within the capturing network. A security researcher might estimate the impact of a potentially malicious domain by counting the number of unique IPs querying for that domain in a given passive DNS dataset. The analysts need to account for factors such as mobility, IP churn, pollution, and biases in the dataset, but this type is mainly possible in passive DNS and not Active DNS. For that reason, tools designed to help network administrators understand the state of their specific networks will continue to require passive datasets.

Overall, when both Active and passive DNS datases are sufficient to perform a particular experiment, we assert that researchers should choose to use Active DNS. This choice dramatically increases the work's repeatability and removes ethical considerations surrounding client privacy.

3.5 The impact of Active DNS data on security research

As we have established, DNS datasets have been utilized in security research and creating security applications [13, 6, 7, 14, 5], with passive DNS being the most popular intelli-

Table 3.1: Taxonomy of works that utilized Active DNS data

		DNS Features		Infrastructure		Additional	
Category	Reference	Domain Name Features	Non-A/AAAA Records	Mapping	Expansion	Temporal	Modeling/Evaluation
Cyber and Privacy	Antonakakis2016[50]			✓	✓	✓	
	Alrawi2021 [51]			✓	✓	✓	
	Kintis2017 [52]	✓		✓		✓	
	Tian2018 [53]	✓					✓
	Mi2019 [54]						
	Avgetidis2023 [55]			✓		✓	
	Hoang2020 [56]			✓			
	Khalil2018 [57]			✓	✓	✓	✓
	Bushart2018 [58]		✓				
	Lee2021 [59]				✓		
	Becker2019 [60]						✓
Other	Hoang2020 [61]			✓			
	Portier2019 [62]		✓			✓	
	Romero2017 [63]			✓	✓	✓	
	Zhou2021 [64]			✓	✓		✓
	Romero2016 [65]			✓	✓		

gence source. However, as we have pointed out, passive DNS data is costly to purchase and collect, comes with legal obligations, is only local to the monitored network, and has limited DNS resources. To make access to DNS data easier, free of charge, and available to the security community and researchers, we presented *Thales*. This collection system gathers DNS data by actively submitting DNS resolution requests for domain names. In this section, we will document how the research community has utilized our Active DNS dataset by presenting a number of publications that utilize Active DNS data.

To demonstrate the value of our collection system and the resulting dataset, we highlight the reception that our work received from the academic community members. Since its presentation, our collection system and data-sharing infrastructure have enjoyed a healthy amount of interest from other security community members for academic and internal protection purposes. We have been thrilled to have discovered that our work has enabled new types of research, either due to the simple act of sharing our data or by documenting our system; we have been inviting others to experiment with the value of actively queried and collected DNS data. Among the users of our dataset, the most obvious would be the other members of the academic community that have obtained and used our Active DNS dataset to produce further works in the academic field, some of which are exceptionally influential in the security field.

Table Table 3.1 presents a taxonomy of the published works that have used the Active DNS dataset since it was originally shared. We observe that most of the publications utilize the dataset for cybersecurity and privacy related research, while we also observe applications in network measurements [56, 62] and graphs and visualization [65, 64, 63]. Moving into how the dataset is being utilized into the published works, we witness that infrastructure mapping and expansion are the two most prominent uses. This is to be expected as Active DNS provides a wide coverage of hundreds of millions domain names per day and these studies use it to map out the infrastructure of the domain names of their works but also identify new domain names hosted on the same or nearby infrastructure (infrastructure expansion). Furthermore, we observe that Active DNS is being used as a dataset for feature extraction and evaluation of various detectors and systems [54, 57, 60], as a large seed to study domain name squatting [52, 53] and finally as a dataset that provides wide access to non-A or AAAA DNS responses for measuring the operational use of TXT records [62] and for identifying benign `CNAME` chains [58]. Lastly we observe that many studies utilize multiple days, month or even years of Active DNS data to observe the temporal changes in domain name to IP mappings.

Issa Khallil et al. [66] were some of the first researchers that utilized Active DNS data that we shared using our program to achieve malicious domain detection using only Active DNS datasets, which was then validated by external ground truth. Their work shows that despite the limitations, *"Active DNS data can be utilized to construct strong associations among domains and subsequently detect malicious domains with high accuracy."* Their arguments for using Active DNS data instead of other datasets echo our motivations for developing and building a data-sharing system. They note that using Actively collected DNS data instead of passive DNS prevents any misplacement of user data and bypasses the difficulty of obtaining a quality passive DNS dataset with comprehensive coverage of the entire domain space. Their work demonstrated that even though active data contain more limited information than other types of DNS data, active DNS data "is an important and

easily accessible information source for malicious domain detection."

However, due to its nature as both a proposed system that anyone can implement and a dataset that we offer freely to the research community, Active DNS required some time for researchers to become familiar with the potential of the dataset and expand its potential uses. One of the more influential works that utilized our data, along with other types of actively collected Internet data, was the very widely cited work by Antonakakis et al. [50] in "Understanding the Mirai Botnet." As is documented in their work, the researchers utilized several Actively collected datasets, such as Censys [67] data and our own Active DNS, which was a significant contributor to DNS Resource Records. Using both Active DNS and passive DNS records, the researchers expanded their ability to identify shared DNS infrastructure by linking related historical domain names (RHDN) and related historic IPs (RHIPs). In conjunction with the other datasets, Active DNS was used to research the ownership of C2C infrastructure. With the help of clustering and data from other DNS sources, researchers could identify various independent C2 clusters that shared no infrastructure, which lent credence to the idea that there were multiple active bot operators during their study period. This effort highlights the use case for Active datasets in assisting malware and resource attribution efforts in the security community. We also utilized and documented the process of using Active DNS to link related historical domain names and related historic IPs in our work on Technical Support Scams in section subsection 4.2.1.

Then we saw the work by Kintis et al. [68] on "Hiding in Plain Sight: A Longitudinal Study of Combosquatting Abuse," where the authors introduced the concept of *combosquatting*, in which attackers register domains that combine a popular trademark with one or more phrases to confuse the potential victim. Their work relied heavily on access to DNS and domain name data, and as such, they utilized Active DNS along with passive datasets to measure the impact of this newly documented phenomenon. One of the critical findings in this work is that the number of combosquatting domains is steadily rising in active and passive DNS datasets while traditional abuse domains remain stable over time.

The most important finding, though, was that due to the way combosquatting domains are formed, there is no practical upper bound to the number of combosquatting domains that can be generated per domain, a feature in contrast with other types of domain squatting. They could also provide a picture of the infrastructure hosting these combosquatting domains thanks to the extensive breadth of coverage provided by Active DNS.

Furthermore, our work has also been utilized by researchers without a traditional security background. Active DNS data was instrumental in the work by Rosa Romero-Gomez et al. [63] with the title "Towards Designing Effective Visualizations for DNS-based Network Threat Analysis." In this work, the authors design an interactive threat analysis visualization that relies on DNS data to assist security researchers in threat detection. Active DNS was used as the dataset on which the visualization platform was built. Active DNS offered a wide breadth of data without the need for considerations such as user consent and was provided for free in a parsed format. Not only was their work successful, but they could demonstrate the usefulness of their visualization platform by conducting a user study of IT security practitioners and having them perform various tasks, all of which were built and performed on top of our Active DNS dataset.

Our work and data collected for Active DNS have also been utilized in research into user privacy and how new protocols might affect user privacy regarding DNS communication. The work of Nguyen Phong Hoang et al. [56] aims to assess the privacy benefits of proposals, such as DNS over HTTPS/TLS (DoH/DoT) and Encrypted SNI (ESNI), by considering the relationship between host names and IP addresses, the latter of which are still exposed. For the authors to be able to quantify the privacy implications of the various protocols, they had to conduct active DNS measurements to obtain a large number of samples. So they turned to Active DNS as a source of active DNS data as well as set up their own infrastructure with vantage points in different geographic areas to investigate how much a network observer can learn about the domains visited by a user based solely on the IP address information captured from encrypted traffic.

Another work by Nguyen et al. [61] tried to present the first study of co-located websites since 2007. Their study relied on Active DNS measurements to conduct an empirical study that revisited web co-location using active DNS datasets collected from the Active DNS project. Their results show that the web is still small and centralized to a handful of hosting providers, with more than 60% of websites co-located with at least ten others in a group comprising less popular websites. In contrast, 17.5% of most popular websites are served from their own servers, the study found. Their work also mentions that this study was conducted using Active DNS datasets, primarily the already mentioned and documented extensive coverage of the domain space provided by the large seed that Active DNS utilizes. It also mentions some limitations due to the decision to use Active measurements, such as the potentially limited amount of regional IPs if all the queries are conducted from a single location.

We also identify interest in producing datasets that offer some of the advantages Active DNS offers, such as no recorded personal user data. One such work is by Pavlos Papadopoulos et al. [69] in which the authors try to create a Privacy-preserving Passive DNS dataset. Their solution relies on existing dataset manipulation regarding user privacy and tries to evaluate how effective it would be for a typical Passive DNS dataset to be converted into a privacy-preserving DNS dataset. The authors in their work mention Active DNS as a different approach that inspired their work but with different research goals. So we can see that Active DNS is not only mentioned as a dataset that was utilized in the project but also as a research idea and a solution to an engaging, thought-provoking problem that the community has still not reached a consensus about the best approach to user data privacy or the user privacy implications of passive DNS data usage in research.

Another interesting use of our Active DNS data comes from the work of Jonas Bushart and Christian Rossow [58]. They study DNS amplification attacks against the core DNS infrastructure and then develop and present a new application-layer DoS attack against core DNS infrastructure that uses amplification. The authors utilized Active DNS data to

identify legitimate use cases for long CNAME chains in their work. They identified that the longest valid CNAME chain in our data in 2016 was eight, while resolvers such as Unbound have a typical limit to CNAME chases of nine. So we can see that active DNS can also be used as a simple snapshot of the typical behavior of DNS responses for a large percentage of the Internet, and in this work was utilized to identify a typical CNAME chase length and then make suggestions about limiting the chase length to prevent DoS attacks, such as the one they propose, while also allowing for legitimate and benign CNAME uses.

Aside from the use of the Active DNS dataset, Active DNS is also one of the featured works in a study by Yuri Zhaniarovich et al. [70] in which the authors present an extensive survey of malicious domain detection systems that utilize a variety of DNS datasets. Their work documents many of the instrumental detection systems that inspired our work and include a discussion of the performance of the various detection systems utilizing many different datasets. The work further contains a discussion about the impact of the data collection aspect of the work. Their findings concur with what we will present, that Active DNS datasets can assist in discovering malicious domains even before their actual malicious use if the active DNS collector has a *broad coverage of the domain name space*.

Another use-case for our Active DNS dataset is mentioned in the work by Ryan Curtin et al. [71] in which the authors propose a DGA domain detection system by utilizing recurrent neural networks and side information. The authors mention that due to the lack of easily obtainable WHOIS information and especially after the passing of the European GDPR regulation, WHOIS becoming less and less effective at providing side-information about domains for classification. Hence, the authors point out systems such as Active DNS as a different channel of information that can be used to enrich their system along with implementation such as Alembic [72], which tries to identify domain name residual trust. We present an implementation of Alembic in section subsection 4.3.1 that is based solely on Active DNS data, mainly non-frequent RTYPEs such as SOA, to provide residual trust results without the use of any WHOIS data.

Altogether, it is evident that the *Thales* collection system and the Active DNS dataset it generates have received significant interest and attention from the academic community, with researchers using it to produce new works mainly in the security field but also in the measurement and visualization fields. From active threat detection, to measuring the Mirai botnet and how big the web is, *Thales* and the Active DNS dataset have been valuable resources for the academic community, providing new insights, enabling new types of research and making access to large DNS datasets easier.

3.6 Summary

As mentioned throughout this work, DNS is vital to the operation of the Internet. Users, systems, and services rely on its operation for most network communication—often without even realizing it. Malware is no different. It makes use of DNS to locate C&C servers and provide network agility. Despite all its uses and applications in network security, it is incredibly difficult to gain access to large, open, and freely available DNS datasets. Such data is often burdened with privacy regulations or access restrictions even when available. This severely limits the pool of security researchers that can leverage DNS datasets in their work. Furthermore, it limits the repeatability of existing DNS-based research. There was a need in the research community for access to large, open, and freely available DNS data. To that end, this work shows how we built a new system, Thales, to query and collect massive quantities of DNS data starting from publicly available lists of domains (e.g., zone files, Alexa, Common Crawl, etc.). We are releasing this system's resulting active DNS data to the public. Since this data is derived from public sources, it can be easily incorporated into new or existing research without having to worry about privacy regulations or access restrictions.

To prove its merit, we provide an in-depth comparison between active DNS and a passive DNS dataset collected on a large university network. Overall, this analysis showed that active DNS data provides a greater breadth of coverage (i.e., a greater quantity and greater

variety of records). Still, passive DNS data provides a denser, more tightly connected graph. Due to these differences, we provided some case studies demonstrating how active DNS can be used to facilitate new research or even re-implement existing DNS-related research (thus promoting repeatability). We sincerely hope that by opening up Active DNS to the security community, we can spur more and better research around DNS.

CHAPTER 4

INFRASTRUCTURE EXPANSION THROUGH FREE ACTIVE DNS DATASETS

4.1 Motivation

Our second contribution builds on top of the work in Active DNS by utilizing the dataset to expand our observational reach and document novel enrichment techniques while at the same time dealing with a real-world problem affecting millions of users. The first published demonstration of Active DNS enrichment techniques and utility is the study of Technical Support Scams (TSS), which was a joint research project with Bharat Srinivasan [8] and published at the World Wide Web Conference in 2018 (WWW18). In this work, Bharat designed and executed a search-engine-based system for discovering TSS. After identifying more than 3,996 TSS-related FQDNs operated by TSS scammers, we utilized Active DNS to identify 5,225 new FQDNs totaling to 9,221 FQDNs. Since TSS is a cross channel abuse observing a more complete picture of the infrastructure of the scam operators would require passive DNS visibility from multiple different geo-distributed carriers and ISPs. Active DNS is a prime alternative of a free and without privacy restrictions dataset that we utilized in this work to identify new malicious infrastructure that was not visible through the search-engine-based system, as Thales is querying daily the majority of the gTLDs where most of the TSS related domains reside [8].

In this chapter, we will quickly document the experimental setup and demonstrate the methodology and module that led to significant expansion in visibility, as well as show how well-known network enrichment techniques allow us to discover more previously unknown TSS domains. Furthermore, we present how similar data enrichment techniques can allow us to utilize our Active DNS dataset to study a wide variety of security-related issues. The results of these case studies are presented in section 4.3.

4.2 Active DNS Network Enrichment of the Infrastructure of Technical Support Scammers

The Technical Support Scam (TSS) is a prevalent cybersecurity issue wherein malicious actors deceive their targets into transferring significant sums of money for counterfeit technical assistance services. This deceptive practice has persisted since the Internet became a routine aspect of daily life for the general populace. Initially, TSS involved scammers conducting unsolicited calls to individuals while impersonating legitimate technology vendors. Over time, it has evolved to encompass advanced online manipulative techniques, aimed at coercing potential victims into contacting phone numbers under the malefactors' control. Prior research, such as that by Miramirkhani et al. [21], has documented the diverse array of methods employed by TSS websites to reach unsuspecting individuals, particularly by exploiting domain parking and ad-based URL shortening services to serve advertisements.

However, TSS scams utilize a number of other social engineering and technical methods to spread and avoid detection. To that end, in this work, we aim to provide a more comprehensive study of the TSS environment. The initial contribution of this study was the ability to scan and identify TSS pages by actively crawling for them, which was performed by Bharat [8], and we will quickly summarize before we expound on the second contribution of this work. The second contribution focuses on utilizing Active DNS data to expand our observational reach and document novel enrichment techniques while at the same time dealing with a real-world problem affecting millions of users. We will document the theory and proof of concept experiments we performed to verify the validity of our data enrichment methodology; then, we will document the experiment that allowed us to expand the initially identified scam domains by 130%. We show that well-known network enrichment techniques allow us to discover many more previously unknown TSS domains.

In this work, the authors utilized a data-driven methodology to explore TSS tactics and infrastructure used to support search-and-ad abuse. To achieve this, Srinivasan built a

search and crawl web engine to collect various data about TSS websites. Then we assisted by helping design, implement, verify, and provide data for the Network Amplification Module NAM). In this section, we will briefly discuss the methodology used by Bharat [8], to provide context on the study and then move on to describe the NAM implementation, its impact on the study and the infrastructure analysis of the TSS domain names.

4.2.1 Methodology

The system, in total, implements TSS data collection and analysis functions and consists of the following six modules:

(1) The Seed Generator module creates search query phrases likely to be employed by individuals seeking technical support resources. This module leverages a known corpus of TSS webpages from Malwarebytes and a probabilistic language modeling approach to generate these search phrases.

(2) Utilizing the generated search phrases, the Search Engine Crawler (SEC) module extracts technical support-related content from widely used search engines, such as Google, Bing, and Yahoo!, through search results (SRs) and sponsored advertisements (ADs). Additionally, the SEC mines a few lesser-known search engines, including goentry.com and search.1and1.com, which we have identified as being exploited by technical support scammers.

(3) The Active Crawler Module (ACM) proceeds to monitor and document URI redirection events, HTML content, and DNS data associated with the URIs/domains present in the ADs and SRs gathered by the SEC module.

(4) The Categorization Module, which incorporates a proficient TSS website classifier, distinguishes TSS SRs and ADs based on the acquired content.

(5) Employing DNS information, the Network Amplification Module (NAM) enhances

signals derived from the labeled TSS domains, such as host IP addresses, to broaden the set of domains delivering TSS through an enrichment algorithm.

(6) Finally, the Clustering Module utilizes the collected data on TSS domains to group domains exhibiting similar attributes at both the network and application levels.

Our direct contribution to this project was the *DNS Crawler* component of the Active Crawler Module. It offers a portion of the Active DNS dataset we have already documented in chapter 3 and was provided as an already deduplicated and processed dataset.

More information about the other components of the analysis system can be found on the originally published work [8], along with more details about the operation of each module. Next, we focus on the development and implementation of the Network Amplification Module, which was our other contribution to this work.

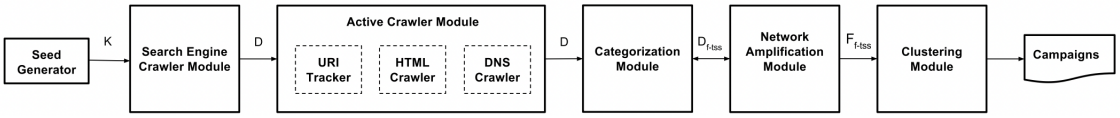


Figure 4.1: X-TSS threat collection and analysis system.

4.2.2 Network Amplification Module

In this research, we build upon the documented methodology for utilizing search listings to identify active TSS websites, which provides a foundational level of understanding surrounding these scams. It is conceivable that incorporating network enrichment techniques on our Active DNS dataset can further this intelligence, revealing additional TSS-supporting domains that the search engine crawler might have missed.

One indication of these supplementary TSS domains, that we exploit in this work, comes from the sharing of network-level infrastructure with already recognized TSS domains. In this context, a DNS request resolves a domain name, d, to an IP address, IP, at a specific time, *t*, generating a *(d,ip,t)* tuple. Let D_{f-tss} represent a set of labeled final-landing TSS domains. For each domain $d \in D_{f-tss}$, we calculate two sets: (i) *RHIP(d)*,

the set of all IPs mapped to domain d as documented by the DNS Crawler within the time window *T*, and (ii) *RHDN(ip)*, the set of domains historically connected with the *ip* or the *ip /24* IP subnet in the RHIP set within the time window $T \pm \Delta$, where Δ is also a time unit (typically one week in our case). Subsequently, we compute $D_{rhip-rhdn}$(d), representing all domains related to *d* at the network level as discovered by the *RHIP-RHDN* expansion (network enrichment method).

For each domain $d' \in D_{rhip-rhdn}(d)$, we evaluate if the associated webpage $w_{d'}$ is a TSS webpage using the classifier module (subsection 4.2.2). Only if this condition is met, we add d' to an enrichment set, $D'f - tss(d)$, linked with *d*, as co-location can occasionally be misleading [73]. The cardinality of the final enrichment set provides the enrichment factor, $\mathcal{A}(d)$. We define the expanded set of TSS domains, $\mathcal{E}f - tss$, as the union of all enrichment sets. Combining the initial domain set, D_{f-tss}, with the expanded set, $\mathcal{E}f - tss$, yields the final set of fraudulent technical support domains $\mathcal{F}f - tss$. The data relating to historic DNS resolutions originate from our prior work on the Active DNS project section 3.2.

4.2.3 Network Enrichment Efficacy

The network-level enrichment process or NAM was instrumental in uncovering additional TSS domains. After removing domains with enrichment factor $\mathcal{A}(d) < 1$, we were left with a conservative estimate of 2,623 domains in the D_{f-tss} set, contributing to the expanded *RHIP-RHDN* set, $\mathcal{E}_{f-tss}$. Figure 4.2 displays the cumulative distribution of these domains' enrichment factors. Approximately 60% of the domains had $\mathcal{A}(d) \leq 50$, while the remaining 40% had $\mathcal{A}(d) > 50$, with a maximum value of 275.

Altogether, the total number of unique FQDNs hosting TSS content was 9,221, comprising 3,996 TSS FQDNs from final-landing websites in search listings and an additional 5,225 TSS FQDNs identified through the network-level enrichment process using the Active DNS dataset. These extra domains were not visible in search engine results or visited by anyone on the university network. Without the Active DNS data querying new domains

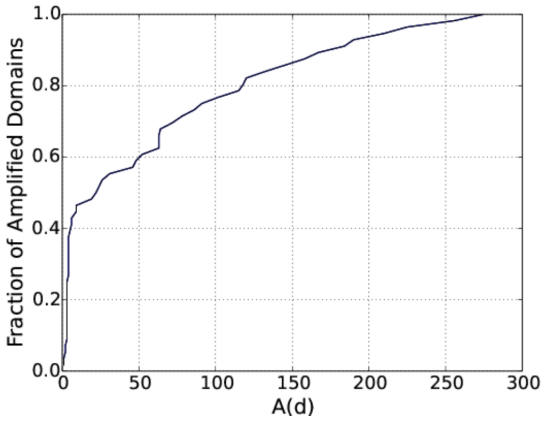


Figure 4.2: CDF of the network enrichment factor, A, of final-landing TSS domains discovered using search listings.

upon registration, we would not have access to these. The 9,221 FQDNs correspond to 8,104 TLD+1 domains, demonstrating the substantial impact of enrichment in discovering domains not visible through search listings alone. The network enrichment process also revealed 840 passive-type TSS domains co-located with one or more aggressive TSS domains, suggesting that some passive scams are operated by the same individuals behind the aggressive ones—a finding that diverges from the original non-amplified set where aggressive and passive TSS domains appeared uncorrelated.

4.2.4 TSS Study Enrichment Conclusion

In this section, we have showcased how a free dataset, Active DNS, can help enhance the visibility of a prominent work by identifying new TSS domain names through infrastructure enrichment thus increasing the domain visibility by 130%. Future works can utilize the freely available Active DNS dataset and implement a similar enrichment process as NAM ,demonstrated in subsection 4.2.2, to expand the visibility of their studies.

4.3 Abuse Detection using Active DNS Case Studies

With these considerations in mind and the information provided about the operation of our system, we have exposed several of the dataset properties from the active DNS data

and have further shown how information enrichment techniques can be utilized in order to take advantage of these new Active DNS datasets. We should clarify that our goal is not to claim any of the following abuse detection processes as a novel contribution. All of them have been discussed in previous works in the field. Instead, our goal is to practically demonstrate, using the actual Active DNS datasets, the security merit that Active DNS data can offer to the research and operational security communities, by simply adopting existing research and network security techniques.

4.3.1 Enhancing The Detection Of Domain's Residual Trust Change

As we've established in the TSS case study the associations between the organizations and infrastructure that support and host a domain name can assist us in identifying malicious domains, even before they are utilized. This leads us to the topic of domain reputation and how that reputation changes based on infrastructure observations about each domain, a topic that we studied in the following case study.

On the Internet, domain names serve as trust anchors for numerous systems and services, and for many, ownership of a domain is enough to prove one's identity. Work by Lever et al. [74] opened a discussion about the problems caused by the use of domains as trust anchors. It showed that residual trust, implicitly inherited by domains after an ownership change, is a root cause of many seemingly disparate security problems. Therefore, identifying changes in ownership due to expiration or any other cause is an important problem in protecting against the potential abuse of residual trust. WHOIS [75] is typically used to discover more information about the owner of a particular domain, and thus, it would appear to be a natural fit for creating a remedy to this problem. However, collecting WHOIS data at scale is outside the grasp of most organizations due to rate limitations imposed on the automated collection of WHOIS records by domain registrars. It is also less helpful as time goes on due to privacy policies that registrars offer automatically to new clients as well as regulations that limit the amount of personally identifiable information that can be in-

cluded in the WHOIS information. To make matters worse, any potential limits frequently vary by registrar, further adding to the complexity of reliably collecting WHOIS data at scale. To circumvent this problem, Lever et al. proposed Alembic [72], a lightweight algorithm for locating potential ownership changes that rely solely on passive DNS data. This algorithm relied upon three different components: changes in infrastructure, changes in lookup volume distribution, and changes in SOA records.

While passive DNS might be simpler to collect for any large network, it is also very sparse as we discussed on section 3.4, resulting in two limitations with respect to Alembic. Scores can only be computed for domains observed in passive DNS and that have sufficient historical resolutions. Active DNS would thus seem like a natural fit as a way to enhance the existing datasets assisting in alleviating these limitations. First, as Figure 3.6a has shown, Active DNS is capable of capturing many more effective second-level domains than passive DNS. Given that the passive DNS dataset used for comparison was generated from a large university network, this result is significant. It demonstrates that even large networks have difficulty matching the breadth of domains that can be collected using Active DNS querying and effective seed management. Next, active DNS querying can consistently gather specified DNS record types over time. In particular, Figure 3.5b and Figure 3.5a show that active DNS results in substantially more SOA records than passive DNS each day. Since one of the key components of the Alembic scoring is SOA records, Active DNS should be able to enhance the performance of the Alembic scoring algorithm. While active DNS provides many benefits, it is important to note that the one component Active DNS cannot enhance is the lookup volume distribution of domains. This component is derived from user behavior observed in passive DNS, and therefore, there is no analog in the Active DNS dataset, as we have previously mentioned.

To evaluate whether Alembic could work using only active DNS, we implemented a modified version of the algorithm, with the assistance of the original author, that excluded lookup volume distribution as a component and used a fixed window size of two weeks.

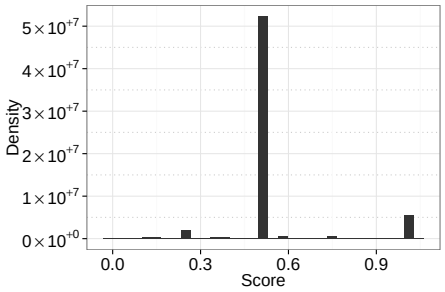


Figure 4.3: Histogram showing the distribution of Alembic scores for March 27, 2016.

Then we computed scores for March 27, 2016, using our modified algorithm. In total, this resulted in 63,332,836 domains with non-zero scores, where larger scores indicate higher confidence in an ownership change. The distribution of those scores can be seen in Figure 4.3.

4.3.2 Enhancing Public Blocklists

Another documented use for Active DNS is the system's ability to enhance blocklists and other datasets containing malicious resources or targets. In the case of Public Blocklists, that information is part of Open Source Intelligence (OSINT) and is often used by the security community for various reasons, both operationally (e.g., as a blocklist of domains or IPs a device should not contact), and in research and development (e.g., as a means to train and evaluate a detection system).

As we showed previously, section 3.4 Active DNS data collected by Active DNS has had visibility into malicious domain names for several days, even months, before they are identified and released in a malicious domains list. To further demonstrate the advantages that Active DNS data collection compared with static blocklists, we performed an experiment when the system was first documented and our work was published. We collect malicious domains that appear in public blocklists and then identify them in the Active DNS dataset.

Due to the nature of Active DNS, we can use the collected data to reveal abuse signals

about domains before they are identified as domains participating in actual malicious use. Blocklisted domains, for example, are an interesting category of candidate indicators of abuse that can be registered, set up, and pointed to an IP location well before they are used in malicious activities. Thus, Active DNS could be used as a potential source of raw datasets that can be utilized for timely domain abuse detection.

As we have already discussed, we could gather a plethora of public domain name blocklists alongside the active DNS data collection. As expected, domain names in these blocklists also appeared in the Active DNS traces we collected while operating the Active DNS project. We identified two important dates for all the domain names seen in both the public blocklists and Active DNS data. The first date denotes the day Active DNS initially probed the domain name. This behavior is driven by the addition of the domain in our seed list that can be caused by a change in any of the zone files collected daily from any top-level domain authorities. The second important date we identified is the first day one of the many blocklists we collect (on a daily basis) actually listed this domain name as part of a particular abusive activity.

We compared the first-seen dates of blocklisted domains and the first-seen date of a domain resolved by Active DNS. We plotted the results in a cumulative distribution function (CDF) that depicts the time difference in days between a resolution in our passive collection or a resolution in Active DNS data and the appearance of the domain in a public blocklist. Negative values denote the number of domains that have first appeared in our Active or passive DNS data before eventually getting blocklisted. On the other hand, positive values represent domains that had been blocklisted before they had a resolution in our data.

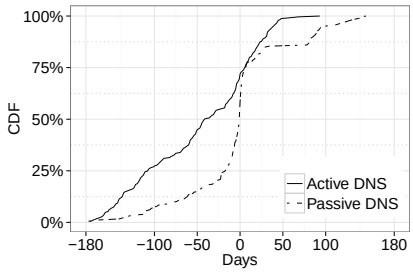
It is worth pointing out that not all the public domain name blocklists were used as a seed domain source for Active DNS, rather the ones described in section 3.2. We point that out because we should expect a fair amount of both positive and negative values in these CDFs. Positive values indicate that a domain name was first seen in a blocklist and then in either the active or passive DNS data that we present in Figure 4.4, while negative

values indicate that the domain was first seen in any of our DNS collections before being blocklisted.

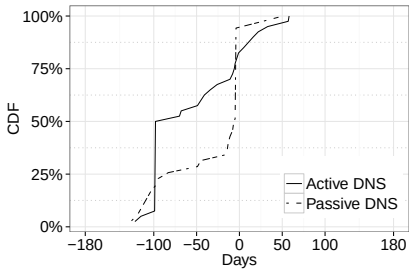
It is readily apparent by the plots presented at Figure 4.4 that DNS datasets in general and Active DNS, in particular, tend to have early visibility when it comes to domains compared with public blocklists.

One of the reasons for this performance is that our system, Active DNS, resolves domains that came in part from zone files for major top-level domains. It queries any domain registered in that zone within a day after it was registered and added to the zone file. This behavior creates a temporal history of the DNS activity capable of describing the IP infrastructure history that supported the domain name, before blocklisting, at the time, and after it was blocklisted. This is a new and very useful property that Active DNS datasets can freely offer to the security community, and it is a property that is rarely seen in passive DNS data. Active DNS exhibits this behavior for a simple reason compared to traditional passive DNS; infections get remediated, and hosts are mobile, thus making it hard for the network operator to passively observe the network evolution of the infrastructure that supports a domain. On the other hand, Active DNS effectively creates a timeline of resolutions for a specific domain, accompanied not only by IP information but also by other record types and any other data we collect and store through our active queries. Thus, Active DNS is able to offer a strong signal augmenting existing passive DNS data to which researchers and network operators have access. This point has already been demonstrated in chapter 4 where we apply the network enrichment techniques using a variety of datasets. The efficacy of this process in the context of our Technical support scam (TSS) work can be found at subsection 4.2.3, and it shows how the technique was able to expand our effective visibility for the TSS project.

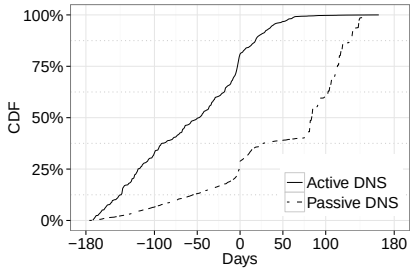
Figure 4.4 shows the CDF plots for different classes of malicious domain names (Figure 4.4a to Figure 4.4d). The values plotted include the domains in our Active and passive DNS datasets that have been blocklisted. Several instances of these domains are found



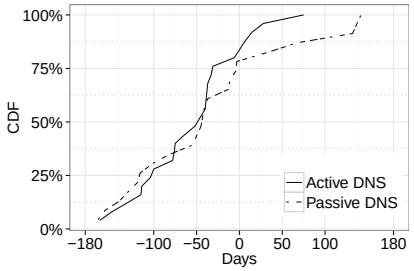
(a) Zeus malware.



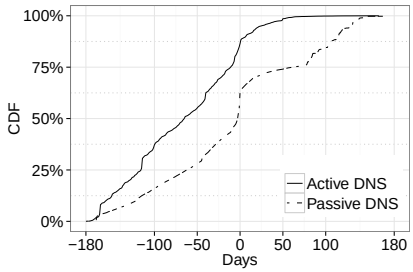
(b) Spam.



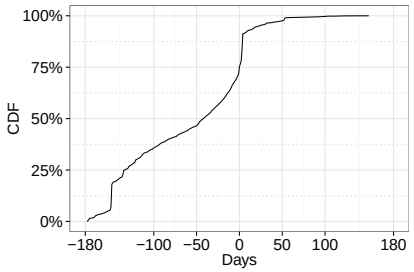
(c) Phishing.



(d) Exploit.



(e) Difference of days from the first time a domain name was seen in Active and passive DNS before it appeared in a PBL.



(f) The difference between the first date a blocklisted domain was seen in Active DNS versus the passive DNS dataset for the domains that were seen before they were blocklisted. Approximately 70% of the 17,000 domains that exist in both datasets and were blocklisted were seen first in the Active DNS dataset.

Figure 4.4: Cumulative distribution of the first seen date in active and passive DNS, subtracting the first seen date of the same domain in a PBL for Zeus, Spam, Phishing, and Exploit domains.

in our dataset long before they were blocklisted; for example, 50% of domain names associated with spam was queried approximately 2.5 months before they were added to a blocklist. On the other hand, we do not have the same visibility for ephemeral types of attacks, such as phishing or exploit kits. In the latter two cases, approximately 75% of the domain names are queried by Active DNS at least one day earlier than blocklisting, with the 50% mark being approximately 50 days earlier.

In total, 42,000 domain names have been blocklisted while also appearing in our Active DNS dataset. From this set of domains, 30% were queried, and DNS data were collected for approximately 100 days before the blocklisting instance (Figure 4.4(e)). For 75% of the block-listed domain names, Active DNS has been collecting and storing data for more than a week before they appeared on a PBL. Considering that PBLs have been used as ground truth for various security systems [15, 76, 77, 78], anyone with access to such a dataset would have the ability to utilize this data over time to model the behavior of these domains and identify the threats long before current systems, or even dare we say, before the adversaries utilize them.

On the other hand, we could identify 20,000 domain names in our passive DNS dataset that also appear in blocklists. The dashed line in Figure 4.4 plots represents these domain names. Approximately 50% of the domain names that are blocklisted appear in the passive DNS data feed, with only 25% revealing themselves 50 days earlier than the blocklisting event, as shown in Figure 4.4e. In this case, we have to note that there are only 20,000 domain names that were blocklisted, and the visibility that the passive DNS dataset provides us is approximately 15% for the 100 days mark. About 50% of all the domain names were seen roughly two days before they were blocklisted, likely due to the domains being actively utilized by the malicious operators. This clearly supports our claim about the merit of Active DNS datasets and how well they complement existing passive DNS repositories. The early linkage between domain names and IP infrastructure witnessed by the active DNS data will be able to enrich the signal that passive DNS data contains, potentially making

local DNS modeling efforts easier for researchers and operators.

In most cases, as shown, the active DNS dataset contains domain names far before they appear in either the passive DNS or the blocklist dataset. Note that the intersection between Active and passive DNS records that have been blocklisted is approximately 19,000 domain names. Almost half of the domains in the active DNS dataset and 95% of the domain names in the passive DNS dataset. Passive DNS seems to show better results in the early days for the spam domain names case (Figure 4.4b), but active DNS catches up very fast (within 15 days) and then loses the advantage again at the time of the block listing events (0 points in the plot). This is expected as spam is a high churn, wide breadth, and low-yield malicious activity.

Lastly, Figure 4.4f depicts the difference between the day a blocklisted domain name was first seen in our Active DNS dataset and the day it was seen in our passive DNS dataset. This includes only the domain names seen before the PBLs included them. Approximately 17,000 domain names were found in active and passive DNS before they were blocklisted. The vast majority of them were first resolved by Active DNS at least one day before a system in our university visited it, thus capturing it in our passive dataset. Approximately 40% of the domain names were already resolved by Active DNS for more than 100 days before they appeared in the passive DNS dataset. An observation that supports our claims that Active DNS can provide early data and a timeline of activity for domain names that might be utilized for malicious purposes in the future.

The majority fell in the range between 0.4 and 0.5, and further inspection revealed that the SOA component contributed the most to these scores. In short, most of the scores in this range resulted from changes in the domain's SOA record. Since we saw very little difference in hosting infrastructure, these scores could be the result of minor changes within the SOA record. The following most extensive range was between 0.9 and 1.0 and consisted of 5,652,910 domains. According to the algorithm, domains with a score in this range are most likely to have undergone a change in ownership. 5,625,397 (99.5%) of these domains

had a score of 1.0, indicating that both infrastructure and SOA records had undergone complete changes. Indeed, we found 10,885 of these domains on a public service's list [79] of expired domains for March 27, 2016. The remainder of these domains provides interesting cases for further study. Our modified version of the Alembic algorithm, initially proposed by Lever et al., provides an interesting example of how active DNS can be used to enhance or extend existing research. Without Active DNS, deploying an algorithm like Alembic would require access to a large-scale passive DNS dataset (e.g., university, enterprise, Internet service provider). However, using openly available Active DNS data, as offered by this research, can help remove the barriers to using or deploying existing DNS research.

4.3.3 Tracking Malicious Domain Names In Non-Routable IP Space

The final case study around the applications of Active DNS data that we will demonstrate in this chapter focuses on the study of Bogons. *Bogons* are private, reserved, or otherwise unallocated network blocks [80, 81, 82]. Bogons, for any average internet user, should be boring since they should not be hosting anything in the context of the global Internet. But occasionally, a domain name, like `messisux[.]bix`(on March 2016), resolved to a bogon like `0.0.0.0` despite the fact this IP can not host anything. The presence of a domain name, however, indicates a service that should be globally reachable exists. These "nonsense" resolutions are at times caused by misconfigurations, brand protection services, and occasionally malicious actors. To investigate further, we don our threat researcher hats and analyze domain names that resolved to bogon IP space during our analysis. Here we focus on malicious infrastructure as it is a primary interest of the security community. However, we also note that Active DNS data that resolves to bogons would be useful in other contexts, such as identifying potential trademark infringements.

We identified two known malicious campaigns in the subset of bogon data: "Operation Hangover" and "CopyKittens." The former is the infrastructure of a cyber espionage threat targeting government, military, and private sector networks with some ties to India [83].

Table 4.1: Operation Hangover and CopyKittens Attack Group Infrastructure and Domain Names.

Operation Hangover	CopyKittens
alertmymailsnotify[dot]com	alhadath[dot]mobi
cloudone-opsource[dot]com	big-windowss[dot]com
download-mgrwin[dot]com	cacheupdate14[dot]com
necessaries-documentation[dot]com	fbstatic-akamaihd[dot]com
newsfairprocessing[dot]com	fbstatic-a[dot]space
onestop-shops[dot]com	fbstatic-a[dot]xyz
servicesloginmail-process[dot]com	gmailtagmanager[dot]com
servicesprocessing[dot]com	haaretz[dot]link
websourceing[dot]com	haaretz-news[dot]com
worldvoicetrip[dot]com	mswordupdate15[dot]com
	mswordupdate16[dot]com
	mswordupdate17[dot]com
	patch7-windows[dot]com
	patch8-windows[dot]com
	patchthiswindows[dot]com
	walla[dot]link
	wethearservice[dot]com
	wheatherserviceapi[dot]info
	windowkernel[dot]com
	windows-drive20[dot]com
	windowskernel14[dot]com
	windows-my50[dot]com
	windowsupup[dot]com

Domain names are seen in Active DNS data for this threat are shown on the left-hand side in Table 4.1. The latter is infrastructure for threats targeting "high ranking diplomats at Israel's Ministry of Foreign Affairs and some well-known Israeli academic researchers specializing in Middle East Studies" [84] and its active DNS domains are shown on the right column in Table 4.1.

These are helpful indicators despite the fact these attacks are known and likely inactive. Neutered yet unidentified infections are likely still operating in networks today, which should lead to incidence responses and damage assessments. For example, knowing the specific internal machine that was infected with targeted malware is valid even after an attack has taken place. An end-user device on a company's corporate network has different

implications than a locked-down server in a data center or the CEO's laptop. Interestingly, some targeted threats resolve to bogon space, while active, to reduce their network footprint [85]. This suggests a signal for malicious detection in the non-routable IPs that can be found in Active DNS data.

4.4 Summary

In this section, we demonstrate actual applications of how the freely available Active DNS dataset can be utilized to expand the network visibility of security relevant studies and tasks and how it can be used to complement passive DNS datasets and aid in abuse detection efforts. More specifically, in section 4.2 we showcase how the Network Amplification Module has expanded the visibility of Technical Support Scam domain names by 130%. Furthermore, in section 4.3, we demonstrate how Active DNS can be utilized for other research and operational activities in network security settings ranging from being used as an alternative to passive DNS data to identify changes in the residual trust of domain names, to enhancing public blocklists and the tracking of malicious domain names.

CHAPTER 5

STUDYING THE OPERATIONAL IMPACT OF CHANGES TO THE GLOBAL DNS INFRASTRUCTURE THROUGH DISTRIBUTED ACTIVE PROBING

5.1 Motivation

The work in this chapter presents a study of the deployment of a DNS extension called EDNS Client Subnet (ECS) [9], which was initially proposed as an experimental Internet draft in 2011 [86]. The ECS extension changes how client IP information is shared with upstream DNS infrastructure in order to optimize Content Delivery Network (CDN) selection. It provides more granular client information to authoritative DNS servers that reveal information about the underlying client making a request. This entire process is transparent to end users who receive no indication of whether ECS will be used by their recursive DNS server. ECS is currently ”on by default” for all traffic through many of the largest open DNS recursive servers. To date, this extension has been adopted by many of the largest open DNS providers on the Internet —including Google Public DNS, OpenDNS, Quad9, and NextDNS [87, 88, 89, 90].

The resulting widespread adoption of the ECS extension by commercial open recursive resolvers means that information is now shared across many networks on the Internet that previously did not have access or visibility to such information. This is potentially problematic because DNS sometimes leaks information about user behavior. For example, the automated DNS behavior of some applications (e.g., web browsers) can reveal limited, indirect information about local users [91]; that is the reason for recommending that DNS prefetching is disabled when dealing with sensitive applications. In addition, DNS attacks on anonymity networks, such as Tor, have been demonstrated. Still, they are either patched or are based on the non-trivial capability of monitoring the recursive footprint used by most

Tor nodes [92].

Furthermore, information leaks through anonymity networks have long been addressed by SOCKS tunneling of UDP queries [93]. Beyond these potential security issues, ECS fundamentally alters the privacy expectations traditionally associated with using a shared recursive DNS server. Therefore, the potential to exploit these negative repercussions of ECS must be studied and better understood.

Although ECS is only an extension to the DNS system, with potential implications for user privacy, we were further interested in studying it in order to figure out how it would affect *Thales*, our Active DNS collection system, and what the impact, if any, of the newly introduced data fields would be to our system capabilities and the value of the resulting dataset. As we studied the history, discussions, and implementation of ECS, we identified that it might offer new, valuable information to DNS collection systems. Seeing as we were already in the process of finding new use cases for our *Thales* infrastructure and also laying out the potential updates to the system that we mentioned in chapter 3, we decided to try and study the effects of this new DNS extension and also try to utilize these new data fields in the DNS packet to enrich our DNS based datasets and more specifically if we should add and store ECS data fields in our existing Active DNS collection.

Beyond the potential adverse side effects, we note that ECS introduces benefits to end users and security researchers. Numerous open DNS providers offer extra features built on top of DNS, such as content filtering, ad blocking, malware protection, and more [88, 90]. These features may be attractive to end users looking to protect their networks. However, before ECS, such users may have incurred a performance penalty for switching DNS providers due to CDN optimization based on proximity to a client's recursive DNS server rather than the client's actual location. Thus, ECS allows users a greater choice of DNS providers without a performance penalty.

Additionally, as we also mentioned, we started work on this project in order to explore whether the ECS extension can allow security researchers to gain more significant insights

into potential infections above the recursive DNS server than they would with a traditional passive DNS collection. This would be particularly valuable to security researchers when monitoring DNS queries at the authoritative DNS server, which sees all non-cached queries for a zone, as mentioned in subsection 2.1.3. Additionally, due to the way it is implemented and deployed, ECS provides a potential added benefit for organizations running DNS sinkholes. By providing wide-reaching client-level visibility, DNS sinkholes can be used to estimate infected populations and provide different responses to clients in different networks if ECS information is appropriately collected and stored. This extra level of visibility can be used to generate previously impossible insights before ECS's introduction.

This work aims, primarily, to understand the real-world adoption of ECS since its inception by utilizing existing datasets and exploring the operational applications that a system such as *Thales* can provide to researchers. It further seeks to provide a discussion of the potential security benefits and pitfalls introduced by ECS, especially concerning the topics of user privacy, forensic investigations, and further expanding operational security measurements. It then tries to identify whether these new data fields in the DNS packet can offer further information and enrich our Active DNS collection system and, in general, DNS datasets. It accomplishes these goals by providing a longitudinal study of ECS deployment using DNS data collected from several vantage points before and after its official adoption in 2016. The outcomes of our investigation can be summarized in the following contributions:

- We measure the ECS adoption from the perspective of three different DNS authoritative name servers to show how the protocol grew before and after its official adoption in 2016. We show that, despite being an optional extension, ECS has seen steady adoption over the years, with numerous DNS providers now supporting it.

- We show how ECS reveals more information about the clients making DNS requests and discuss the effects of this increased visibility. We discuss how it may provide

more freedom to end users and how it can aid security practitioners. At the same time, we discuss how it potentially exacerbates existing threats and offers researchers better forensic investigation capabilities through user location information.

- We examine the practical benefit provided by ECS to end users by using a combination of Alexa domains and our active collection infrastructure to construct a novel experiment that identifies one interesting operational application of *Thales* in assisting us in studying the DNS infrastructure itself. We show that the vast majority of highly ranked ECS-enabled domains *do not* benefit from the use of ECS. Thus, most ECS-enabled domains appear to exacerbate existing privacy problems related to DNS without any benefit to the end user.

5.1.1 Evolution of DNS with ECS

The typical resolution process, as we have mentioned already in subsection 2.1.1 can be conceptually split into two parts. The first is the communication between the stub resolver (client) and the recursive, seen in steps (1) and (8), which is said to occur *below the recursive*. The second is the iterative resolution process shown in steps (2) to (7), which is said to occur *above the recursive*. Below, we discuss how the adoption of ECS enables entities above the recursive to acquire client-specific information that was not available to them before.

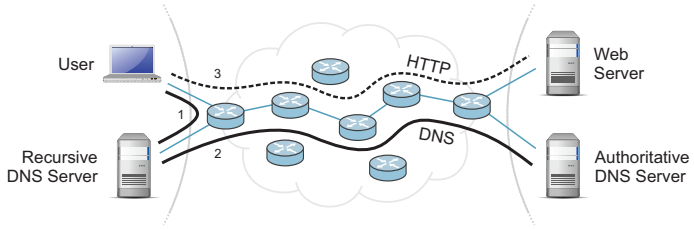


Figure 5.1: Legacy DNS network topology. Typically, recursion took place in the user's own autonomous system, and authorities were often situated in the same AS as the web server. Both DNS and HTTP traffic followed the same network path.

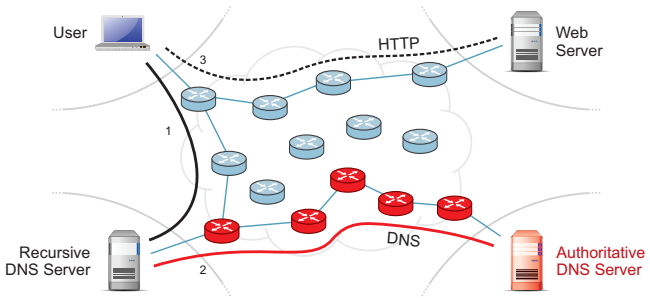


Figure 5.2: Modern DNS network topology. Increasingly, clients query a "cloud DNS" host or open resolver situated at a different autonomous system, and modern websites frequently outsource DNS management to third parties. Due to the inclusion of ECS information in DNS requests, a fraction of autonomous systems that would otherwise be unrelated to the path between the user and the actual web server are now in a position to gather client-specific information about browsing (or other) activity.

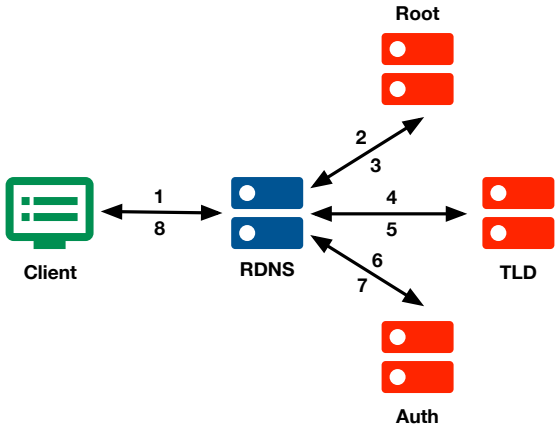


Figure 5.3: Illustration of the iterative name resolution process. In the diagram, the recursive is labeled as RDNS, and the authority is referred to as Auth.

EDNS Client Subnet (ECS) [9], has been adopted by most large open recursives [94, 95], which does not change the resolution process below the recursive but augments the information exchanged between recursives and authorities. Without ECS, only communication below the recursive (step 1) reveals the IP address of the clients. Thus, authorities receive no information about who is performing a query other than the IP address of the recursive. For example, as shown in Figure 5.1, only the IP address of the recursive DNS (RDNS) server is revealed to entities outside the local autonomous system (AS).

What primarily changes with the adoption of ECS is the information contained in the

communications above the recursive, which is shown in steps (2) to (7) in Figure 5.3. The steps are still the same; the main difference is that when the recursive resolver and the authoritative DNS servers support ECS, the DNS packet contains the extra information to help the authority identify the user's general geographic location. This change has come about due to the changing landscape in how DNS resolutions are performed nowadays. The rise of open recursive DNS servers, which are typically situated in separated ASes than the users (as shown in Figure 5.2), disrupts the optimal delivery of content (e.g., as performed by CDNs), which previously assumed users were proximate to their resolvers. When a user resolves the name of a CDN-enabled website, the authority DNS server would respond with a web server address close to the recursive instead of the actual user. Thus, a North American user relying on a European DNS server could be directed to a non-local CDN mirror, slowing the resulting TCP connection.

ECS attempts to address this issue by including a truncated portion of the client's IP address, referred to as the *source netmask*, in all subsequent requests made by the recursive to an authority supporting ECS. An authority usually indicates that it supports ECS by including a scope netmask in reply to an ECS-enabled query. On the other hand, some recursive resolvers send ECS-enabled queries to all authorities. This added user information allows selecting a mirror that is in close proximity to the actual user, not just their cloud recursive. According to the ECS protocol [9], the source netmask should be determined using the most detailed network information available to the recursive, but by default and what we have observed in practice, it includes the first three octets of a client's IP address. An authority may include in its response a *scope netmask* that can guide a recursive's future choice of source netmask. Including a scope netmask by the authority is one way to signal that the authority supports ECS. The scope netmask indicates the authority's desired source netmask length, which should correspond to the minimum length that will allow for an optimal answer with respect to network performance. The recursive resolver also uses the scope netmask to help with caching an answer; based on the documentation, the an-

swer from an ECS query with a scope netmask indicates the scope under which the answer is valid, and the recursive can proceed with caching the answer for the clients under the specified netmask. The caching behavior and the potential issues that can arise from it are further discussed in [35].

Finally, the discovery process of ECS authorities by the resolvers varies but usually relies on the recursive resolver sending ECS-enabled queries and observing if the authority responds with a scope netmask in most cases. Some operators repeat the discovery process over time, while other recursives do not keep a list of authorities that support ECS and instead send ECS-enabled queries to all the authorities.

5.1.2 Implications of ECS Misuse

One might think that the information that ECS introduces to the DNS packet does not present any additional privacy leakage since the actual HTTP traffic will eventually reveal a user's IP address to the web server (and all entities along that path). However, this is not true on the modern web for two main reasons:

1. The recursive DNS server is often situated in a different AS than the user.

2. The authoritative DNS server is often situated in a different AS than the website.

Consequently, when resolving a domain name, there is no guarantee (and should be no assumption) that the same organization will manage both the DNS server and the web server. ECS introduces new ways to expose the added user information to parties that would typically not have visibility in the traditional DNS resolution case. When using an ECS-enabled cloud-based recursive, the DNS resolution request might have to follow a different path between the recursive and the authority (red line Figure 5.2). For example, a user located in a European country using an open cloud-based DNS resolver could have their DNS packet information leave the confines of their ISP and traverse outside third-party networks on the way to the DNS resolver. This is a case of below the recursive information

leakage and applies to all DNS queries before and after ECS adoption where the DNS recursive resolvers are outside the user's ISP network. On the other hand, after the cloud-based DNS recursive accepts the query, if the recursive support ECS, all subsequent hops from the recursive to ECS supporting authorities would include the client's ECS-related IP prefix. When, in this case, the authority is on a different path, signified by the red line in Figure 5.2, all the hops will receive the unencrypted ECS IP prefix of the client. This ties back to point (1), where even when the authoritative is in the same AS as the web server (e.g., as shown in Figure 5.1), the network path from the user to the web server will be different than the path from the third-party open DNS resolver (e.g., Google's 8.8.8.8) to the web server. More importantly, regarding (2), due to the increasing reliance on third-party DNS hosting services (e.g., No-IP, EveryDNS, EasyDNS, Afraid, Zoneedit, Cloudflare), the path between the recursive and the authoritative may be completely different than the path between the user and the web server, as shown in Figure 5.2.

As we show in section 5.3, it appears there is significant misuse of ECS on the Internet. While, in many cases, this may pose no privacy concerns, in others, the users' anonymity may be seriously jeopardized. For example, Kintis et al. [50] discussed how this information could enable highly stealthy and targeted man-in-the-middle and surveillance attacks against dissidents, minorities, and even entire industry sectors. This information leak potential is even more sharply debated at the beginning of the new decade after the impact of the ever-shifting legal landscape was realized due to a number of legal changes and international awareness about minority populations. This awareness brings renewed focus to the potential privacy impact and the actual freedoms of populations under mass surveillance systems, especially considering that ECS monitoring can be enabled at any point of transit along the global network path, and utilized as a coarse sieve for the selection of populations for further investigation, due to their Internet activity.

In any case, it cannot be denied that users can indeed benefit from ECS; however, its correct and absolutely necessary deployment is paramount in order to minimize the

possibility of privacy leakage.

5.2 Methodology

In this section, we discuss how we study the adoption of the ECS protocol by recursives from three different vantage points and investigate the client information sharing due to ECS from the perspective of the authority (i.e., what *additional* client-related information ECS shares with authorities) and its applications. We first describe the datasets and provide statistics about the observed legacy and ECS-enabled requests throughout our different collection sources. Then, we demonstrate that the ECS protocol is already widely adopted across our sources and constitutes a significant percentage of the DNS traffic. Next, we demonstrate how the ECS-enabled captured traffic can be utilized to provide a view of the clients behind the ECS-enabled recursives that make the DNS queries to the authoritative servers through a sinkhole authority case study.

5.2.1 Datasets

Top Level Domain (TLD): This historical dataset consists of queries to popular Top-Level Domain (TLD) zones. The DNS queries for this dataset span one year, from July 2014 to July 2015, beginning just before the wide adoption of ECS. This source gives us a coherent view of the first years of the ECS adoption and shortly before the release of the RFC (RFC 7871 [9]) in 2016.

DNS Zones: This DNS dataset consists of DNS traffic to authoritative DNS servers for several popular zones. The data from this authority ranges from March 2017 to June 2019. It contains DNS traffic for 9.8 Million unique IPs. We are utilizing this dataset to get a coherent view of ECS adoption after the official release of the RFC.

Sinkhole: Our sinkhole passive DNS data consists of a total of 24 sinkhole domain names related to targeted attacks from Advanced Persistent Threats (APTs) and typosquatting [96, 97] and combosquatting [52]. When users visit these sinkholed domains due to social

engineering or a typographical error when typing the domain name in the browser, our domain names get resolved, and we record the resolution process.

ISP DNS dataset: This dataset was collected by a large ISP (top 10) in North America over the first five days of April 2019. This ISP provides services over the entire North American region and provides us with real-world information about the state of ECS and its usage. We use this dataset in section section 5.4 to provide us with more insight into the benefits that ECS-enabled domains obtain by adopting the protocol.

Alexa: As part of the study we use the list of the most popular domains compiled by Amazon's Alexa. We use this dataset to help identify popular domains on the Internet and examine their support for ECS and how they might benefit, or not, from supporting ECS.

***Thales* generated random client queries:** Finally, we use the already documented *Thales* infrastructure, modified to query selectively with either pre-selected ECS client prefixes or random ECS client prefixes, and also collect, process and store the new incoming information. The resulting datasets enabled us to study whether popular domains on the Internet that support ECS actually benefit from enabling support for the extension, as well as the change in behavior throughout the years.

Table 5.1 provides a detailed view of the first three datasets mentioned above, the size for each one, and the time period they cover. At this point, we should note that the TLD authority data is approximately 1.5 years older than the Zones and Sinkhole datasets. Even though this could seem inconsistent at first, our results will demonstrate the statistical significance of our measurements, even though time periods might not overlap. Moreover, obtaining contiguous datasets of such large volumes and different time periods is particularly difficult. However, we chose to use all three datasets in our study to paint a clear and longitudinal picture of the different ECS uses and changes from the very early adoption days until recently. The fourth dataset, the ISP DNS dataset, is used to provide our study insights into the beneficial aspects of the adoption of ECS, such as its utilization from CDN providers.

Table 5.1: The four types of passive DNS datasets that we utilize in our study. For the first three datasets, the dates span from July 2014 before the official adoption of ECS and then follow the evolution and growth of its deployment using popular DNS Zone authority data that we collected.

Dataset Type	**Size**	**Time Period**
TLD Authority	141.9T	2014/07/01—2015/07/09
Zones Authority	50.9T	2017/03/10—2019/07/17
Sinkhole Authority	455.6G	2017/09/10—2019/06/20
ISP DNS dataset	4.2T	2019/04/01—2019/04/06

5.2.2 Identifying ECS in Our Datasets

Figure 5.4 shows the volume of ECS-enabled and legacy DNS resolution requests for the TLD and the DNS Zones authorities. We observe that in both authorities the vast majority of the DNS requests are non-ECS-enabled. In the TLD authority dataset, which goes back to July 2014, we observe no ECS-enabled queries until mid-August 2014. We spot the first noticeable volume of ECS-enabled queries on August 20, 2014, with a total of 2.6 million queries from 95 different recursives, all of which can be traced back to Google by using the Route Views [98] BGP announcement project database. ECS-enabled traffic constitutes about 0.2% of the daily TLD traffic in 2014-2015, featuring an increasing trend over time. After the adoption of the ECS RFC, we noticed that the ECS-enabled requests make up 30% of the daily DNS traffic, with a mean of 295 million requests per day. This clearly indicates the significant growth after ECS was adopted as an RFC.

Our sinkhole dataset contains observations between September 10, 2017, and June 20, 2019. On the first day of our experiment, we had 11 domain names sinkholed. We kept incorporating more and more domain names in our sinkhole, reaching a maximum of 24 domains. Figure 5.6 shows the volume of ECS-enabled and normal DNS resolution requests. Contrary to the previous two datasets, we observe that the ECS-enabled requests constitute the majority of the traffic to our sinkhole authority, while the daily query volume is unsurprisingly orders of magnitude smaller than that of the other two authorities. In total, we saw 11.5 billion DNS requests from which 69% contained ECS information.

By looking at the IPs of the recursives making the DNS requests at the TLD and DNS Zones authority in Figure 5.5, we initially observe that the vast majority of the recursives that query the authorities do not use ECS. The ECS requests come from a small number of recursives that increased from less than 100 in the first months of ECS-enabled traffic in 2014 to more than 1,000 in 2017-2019. Figure 5.7 shows a similar trend for the recursives at the sinkhole authority. The ASes that host ECS-enabled recursives are predominately owned by Google to the tune of 1,579 (85%), 2,444 (46%), and 500 (78%) of the recursives for the sinkhole, DNS Zones, and the TLD authorities. Likewise, Google's recursives handle most of the ECS-enabled traffic, as shown in Table 5.2, which shows that in all of the authorities, Google handles the vast majority of the ECS traffic.

Considering this large data aggregation on Google's recursives and the overall sensitivity of the DNS lookup data, ECS can be a significant privacy concern in the event of any large recursive operator deciding to monetize this data, or during the case of a breach in one of the open recursives. Similar concerns on data aggregation on third-party recursives have been raised regarding other recent proposals concerning DNS [99, 100]. While the recursives used for the ECS requests were located in at least 21 countries, with the majority of them (54%) stationed in the US, based on the ECS information, we were able to observe more than 76,000 "/24" networks from 202 countries, according to the MAXMIND GeoLite2 City [101] database and the Public DNS recursive locations as publicly disclosed by Google. Some of those networks that we were able to observe based on the clients behind the recursives, link back to governmental IP space, or IP addresses tied with illicit activities (according to various blocklists).

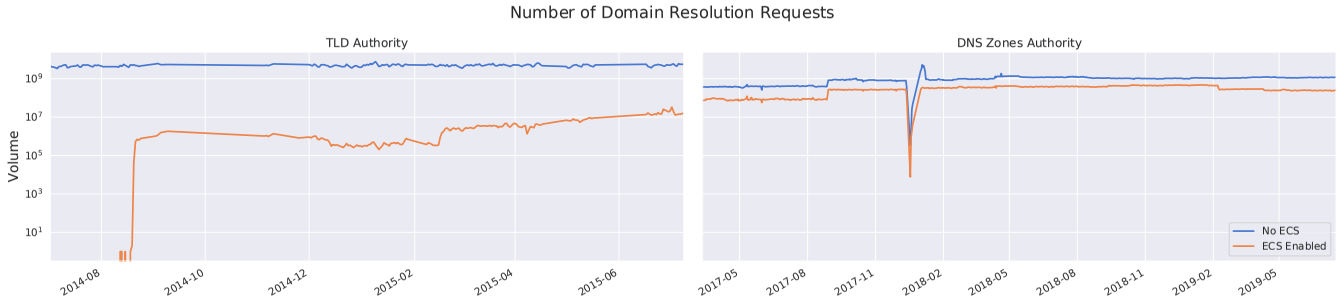


Figure 5.4: The number of daily legacy and ECS-enabled DNS requests to the authorities. The non-ECS-enabled requests constitute the majority of the DNS requests. The dip in December 2017 in the DNS Zones authorities is a result of collection issues (missing data) during that period.

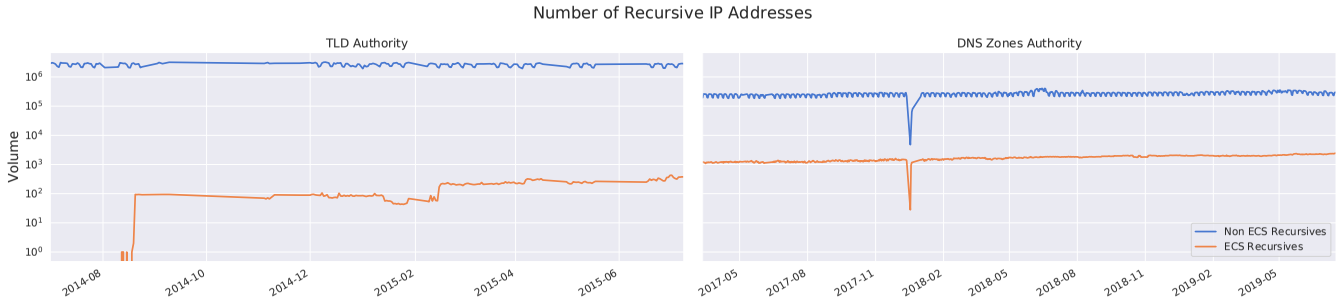


Figure 5.5: The number of different legacy and ECS-enabled recursives that resolved domain names in the authorities. Most of the recursives do not utilize the ECS protocol, while ECS traffic emanates from a small number of recursives. The dip in December 2017 resulted from collection issues (missing data) during that period for the DNS Zones authority.

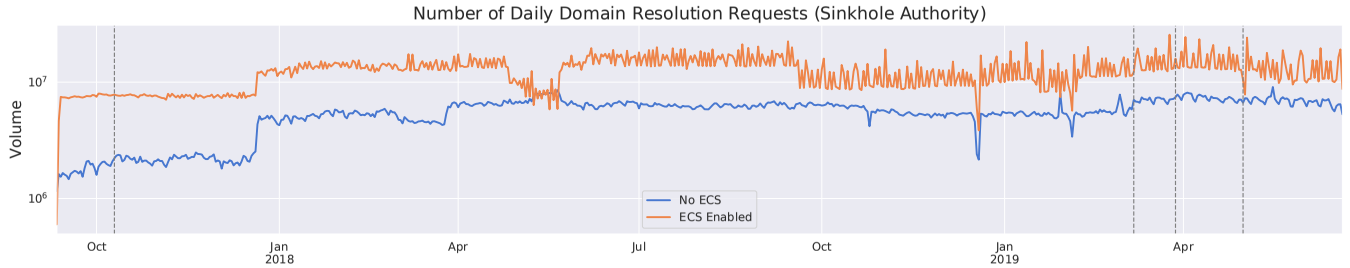


Figure 5.6: The number of daily legacy and ECS-enabled DNS requests to the sinkhole authority. The dashed lines represent the event of the addition of new domain names to the authority. Contrary to the global authority data, the ECS-enabled requests constitute the majority of the overall traffic.

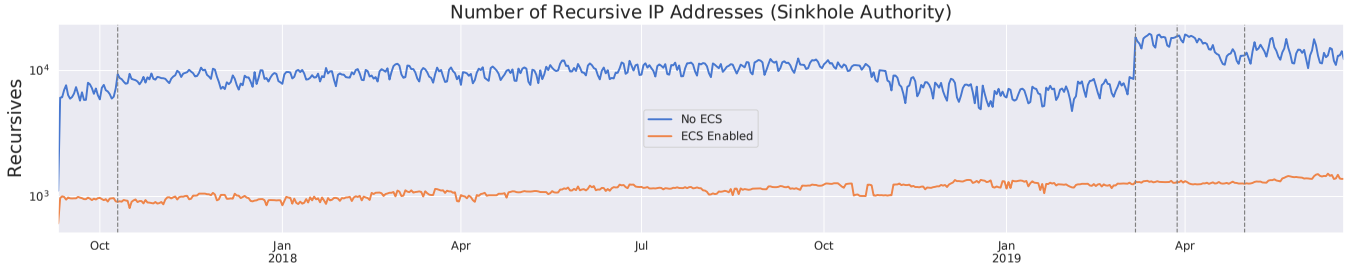


Figure 5.7: The number of different legacy and ECS-enabled recursives that resolved the sinkholed domains. The vertical dashed lines represent the addition of a new sinkhole domain name to the authority. A large number of legacy recursives have submitted resolution requests, whereas the ECS-enabled requests originate from a very small number of recursives.

Table 5.2: Top 5 Autonomous Systems where the ECS-enabled recursives reside. Clearly, the vast majority of the ECS-enabled requests to all of the authorities come from Google's recursives.

TLD Authority		DNS Zones		Sinkhole Authority	
Recursive IP AS Owner	Queries	Recursive IP AS Owner	Queries	Recursive IP AS Owner	Queries
GOOGLE - Google LLC, US	743M	GOOGLE - Google LLC, US	212B	GOOGLE - Google LLC, US	8B
AS-APPRIVER - APPRIVER LLC, US	3,546	OVH, FR	19B	AS-CHOOPA - Choopa, LLC, US	155,307
COMCAST-7922 - Comcast Cable Communications, LL	2,649	OPENDNS - Cisco OpenDNS, LLC, US	11B	DYNDNS - Oracle Corporation, US	40,754
CHINANET-BACKBONE No.31,Jin-rong Street, CN	365	IPV6 Internet Ltda, BR	844M	CHINANET-IDC-GD China Telecom (Group), CN	38,270
DETEQUE - Deteque LLC, US	142	DYNDNS - Oracle Corporation, US	148M	SKYTEL-AS, GE	36,140

5.3 Measuring ECS in the Real world

In the previous section, we presented the state of ECS adoption, as can be observed through passive measurements. In this section, we will present a measurement study of the real-world deployment of ECS among popular websites to examine whether its use is justified by exploring the operational applications of *Thales*. We will also present a study of sinkholed domains and how the use of ECS can provide us with information about the clients connecting to our sinkhole, potentially enabling better and more readily available data for forensic investigations.

To begin, we investigate the privacy preservation claims in the ECS RFC [9] with respect to the length of the source netmask (Section subsection 5.3.1) and show that the prefixes suggested for ECS do not necessarily reflect the reality in terms of routing on the Internet, where the vast majority (50%) of prefixes have the same /24 that ECS recommends. Second, in Section subsection 5.3.2, we revisit prior work by measuring the deployment and distribution of ECS-enabled resolvers and identify steady growth in the adoption rate of ECS across both popular and less popular sites. Third, we examine various properties of the observed ECS speakers. We show that over the years, more and more domains have opted to support ECS; even though many ECS speakers do not appear to represent content delivery networks, the same technology ECS is meant to assist. In fact, the majority of ECS-enabled domains (80%) do not exhibit any kind of CDN behavior. (Section section 5.4). Lastly, in Section subsection 5.4.2, we show that ECS-enabled domains often do not exhibit CDN behavior but utilize outsourced and managed DNS services

by commercial providers. These services reside in different autonomous systems (AS), and anyone else on the DNS path can collect client-specific information (through the ECS client netmask) that would be otherwise unavailable to them if not for ECS.

5.3.1 Revisiting the Default ECS Configuration

In Section 11.1 of RFC 7871 for ECS [9], the authors discuss some privacy considerations due to the use of the proposed extension. The RFC authors' suggestion is for recursive servers to truncate the client's IP address into a source network mask, which typically contains the first 24 bits, in an effort to preserve privacy. The authors also suggest that entities responsible for operating ECS-enabled recursives should adjust the source netmask such that it reveals the least information possible about the client, while still providing beneficial information to ECS-enabled authorities so that they can serve the client the proper IP. To the best of our knowledge, no study demonstrates what portion, if any, of an IP address can be safely revealed while still preserving user privacy. Thus, it is unclear if the RFC's suggestion of using a source netmask is sufficient.

Before evaluating the privacy suggestions in the RFC, we first compared the assumed "/24 default" in the protocol to the general allocation and delegation practices found in IPv4. The RFC suggests that the /24 of the client's IP is sufficient to protect users' privacy while allowing better geolocation identification. However, that brings up the question of whether another choice, for example, /16 or some other mask size, would perform as well while leaking less information. To that end, we show the distribution of the announced prefixes on the Internet for two snapshots, one in April 2015 and one in June 2019. The dataset also includes the organization for each announced prefix. These figures showcase the changes the Internet has undergone over the past few years.

For the 2015 dataset, we downloaded the public delegation files from each of the five Regional Internet Registries (RIR) [102]. Since delegations are updated daily, we picked a single snapshot spanning April 6, 2015, to April 7, 2015, to limit experimental complexity

and account for time zone variations across registries. The delegation files only map prefixes to autonomous systems. Therefore we utilize the Team Cymru IP to ASN tool [103] to associate the prefixes and their associated organizations. On April 9, 2015, there were 325,680 prefixes found for the entire IPv4 address space. While the vast majority of these were /24s, the shortest prefix found was a /8, and the longest was a /32. Figure 5.8 shows the distribution of these prefixes across the different subnet masks. Note that the vast majority (50%) of the prefixes are allocated as /24, and only 12,496 (4%) are less than or equal to a /16 subnet.

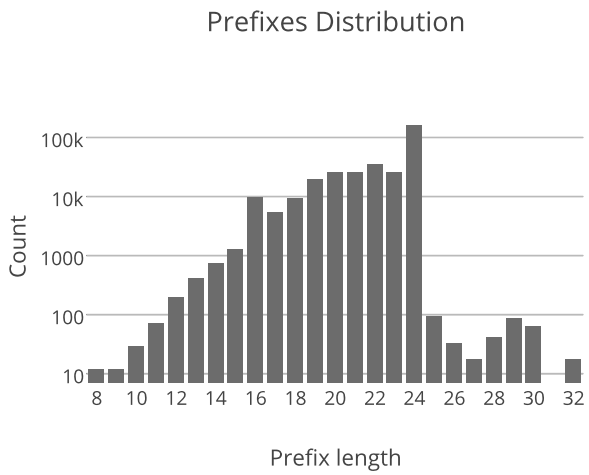


Figure 5.8: The distribution of prefixes (log scale) announced on the Internet for 2015 as reported by Team Cymru's IP to ASN Mapping service.

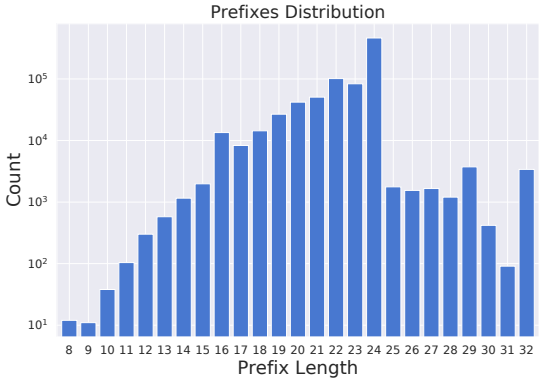


Figure 5.9: The distribution of prefixes (log scale) announced on the Internet as reported by Route Views in 2019.

We examine the same statistics for June 2019, this time through the Route Views [98] dataset (that already has the ASN to organization mappings), in order to measure how this aspect of the Internet has evolved. As indicated by Figure 5.9, the total number of prefixes currently is 732,182, more than double compared to four years ago. This is expected since the Internet landscape has constantly been evolving over time, with more and more companies opting to use it. However, the distribution of the prefixes has remained more or less the same, especially for the lower prefixes (up to /24). Approximately 396,045 (54%) of the announced prefixes are /24, and 17,588 (2.5%) are less or equal to /16. The utilization of prefixes larger than /24 has also increased considerably, but that is not of immediate interest to our study since the RFC has determined that, at most, a /24 can be leaked without compromising the client's privacy. Overall, by comparing the two figures side by side, it is evident that there is a trend of smaller portions of the IP space being delegated to the organizations, resulting in a higher percentage of longer network prefixes.

Considering that ECS reveals the IP address of the stub resolver, and depending on an organization's network infrastructure, ECS might reveal information about the stub's behavior to third parties that might not be necessary for the optimal routing of data back to the client. This applies particularly to cases where the client inside a network uses a DNS provider outside the organization network with ECS enabled as the default. In such cases, ECS should ideally use only the prefix length that would provide enough geographical information for optimal content delivery. With this in mind, we attempt to measure the extent to which this if feasible as we vary the length of the subnet mask. We expect that, by reducing the prefix length, the behavior of clients within an organization will no longer be able to be uniquely identified. This is also a way to theoretically test the ability to set custom ECS network masks as per the RFC. Organizations that want to avoid this type of information leak should augment their IT policies to make sure that any client operating on their network is not using cloud-based DNS providers that might expose their organization's IP

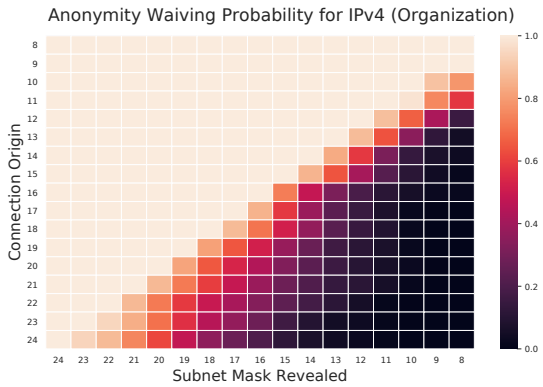


Figure 5.10: The probability that a client's organization can be precisely identified, given its actual network prefix (y-axis) and the revealed source network mask through ECS (x-axis).

to outside parties. Considering the prevalence of the use of cloud-based DNS resolvers and their use in devices like mobile phones, we believe that this part of the study can help shed some light on the potential issues that can arise in such a scenario.

We set up the measurement as follows: Using the Route Views dataset [98] of prefixes and corresponding organizations, we organized the prefixes as a radix tree so that it is easy to collect the organizations covered by a given prefix. Then, we calculated the probability of a network prefix falling into a single organization for a given source network mask. We chose to use network prefixes instead of IP addresses because they align more closely with organizational delegations from RIRs. We limit our computations to prefixes between "/8" and "/24" because ECS suggests that a "/24" prefix is sufficient. For each possible network prefix in the Route Views dataset, we sequentially reduced the length of the source network mask being revealed and measured the number of unique candidate organizations after each reduction. The probability computed for a given network prefix and source network mask equals the percentage of prefixes for which only a single organization was a possible match, i.e., the probability of uniquely identifying an organization for a given prefix and source network mask.

For example, let's assume a stub resolver connects from a given prefix X.Y.Z.W/24

(with subnet mask length 24). We reduce the length of the network mask to 22, and we get all the organizations that are covered by X.Y.Z.W/22 with prefix lengths 22,23 and 24 (there is no point examining lengths more than 24 since we assumed that the origin subnet is a /24). If only one organization appears, then the target organization can be uniquely identified, even after reducing the mask length by 2 bits.

Figure 5.10 shows that changing the source network mask does not always increase the number of candidate organizations a request may originate from. For example, a user connecting to the Internet through a /24 and revealing a /16 source network mask can be linked to a *single* organization about 20% of the time. The likelihood of uniquely identifying the organization jumps to 50% if a user connects from a /16 and reveals a /14 mask. Consequently, it is often quite easy to precisely identify the originating organization, despite changes in the source network mask introduced by ECS. Most organizations, though, have publicly documented network boundaries, so this is only a consideration for not documented organizations.

With these observations in mind, we can observe that the default subnet considerations made by the ECS RFC are proving to have wider implications in terms of identifying behavior from within networks, especially because so much of the Internet infrastructure is based around network masks of /24. This is also the observation that the authors of the RFC had made early during draft-making progress [104] that there should be a selectable mask length flag due to the potential privacy concerns of the initial proposal. Unfortunately, there is still no user-facing option to select the size of the subnet mask.

5.3.2 ECS Adoption Over the Years

In 2013, a group of researchers from the Technical University of Berlin measured ECS adoption among the Alexa top million websites [34]. In their findings, it is stated that approximately 13% of the Alexa top million domains were found to provide some support for ECS.

With the goal of measuring how the adoption has evolved over time, we set up a custom resolver that complies with RFC 7871 and implements ECS. Then we performed similar experiments by sending ECS-enabled requests to authorities and analyzing their responses. The pipeline of the query information is simple and has been a part of *Thales* since it's inception. First, we collect all the Alexa top million domains that we are going to Actively query. Second, as part of the resolution process we query for the nameservers that are authoritative for these domains. As a final step, we pull the domain's `A` record from the corresponding authority using our ECS-enabled resolver with a random client subnet. The detailed results for two randomly selected days, one in April 2015 and one in June 2019, are presented next. These snapshots will reveal useful information regarding the ECS adoption.

In August 2015, using the prototype version of *Thales* we found that there were 5,607 ECS-enabled authorities, which account for approximately 3% of all authorities (187,730), that serve domains in the Alexa top million. Due to network errors and misconfigured authorities, we were able to successfully measure 731,813 (73%) of these domains, out of which 161,302 were ECS-enabled and served by the previously identified ECS-enabled authorities. Almost 22% of the domains are ECS-enabled; this represents a 9% increase in ECS-enabled domains since 2013.

In the June 2019 case, approximately 92% of the total domains were successfully measured, as a result of the upgraded collection infrastructure, and the number of ECS-enabled authorities is 19,133 (out of 173,905 authorities), a huge increase compared to 2015 data. This accounts for 11% of the unique authorities that serve the top domains in the Alexa dataset. However, ECS adoption over the years has been strong, and 418,314 out of the measured 922,139 (45%) domains support ECS. This is a significant increase compared to 2015 as well.

Given that ECS aims to improve network performance, intuition suggests that popular sites are more likely to benefit from its use, and therefore, they should be more likely to use ECS-enabled domains. To test this hypothesis, we aggregated the ECS-enabled domains

in the Alexa top million by their Alexa rank. Figure 5.11 and Figure 5.12 present the distribution of ECS-enabled domains and authorities, respectively, according to their rank for our measurements in 2019 and 2015. The authority rank is defined as the average Alexa rank of all domains for which the given DNS server is authoritative.

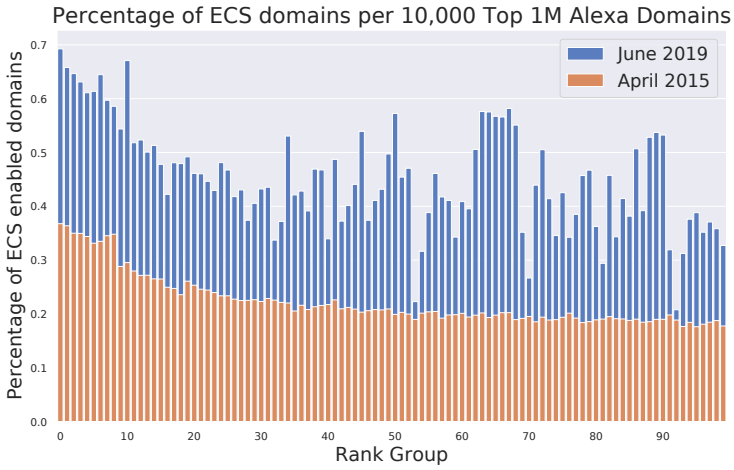


Figure 5.11: The percentage of ECS-enabled domains from the domains that responded, aggregated into buckets of 10,000 elements, for the Alexa top million websites for 2019 and 2015. As expected, the most popular domain names are also ECS-enabled. In total, we identified 161,302 ECS-enabled domains in April 2015 and 418,314 in June 2019.

By closely examining Figure 5.11, we see that there is indeed a recognizable trend that a larger percentage of the highly ranked domains tend to support ECS compared to the lower ranks, with some notable exceptions. Especially for 2015, this is very clear, even though the ECS adoption between ranks does not vary as greatly as in 2019. The top ranks of the Alexa dataset traditionally do not change drastically over time compared to the lower ranks. Additionally, these domains are associated with sites visited by millions of clients daily. Consequently, these sites often have multiple servers located all around the globe and may use CDNs to help improve network performance. Such sites represent the intended beneficiaries of ECS. Compared to 2015, it is apparent that many more domains support ECS today, even in the lower ranks. Websites are increasingly relying on CDNs to enhance their customer. This, combined with the introduction of new domains in the dataset that

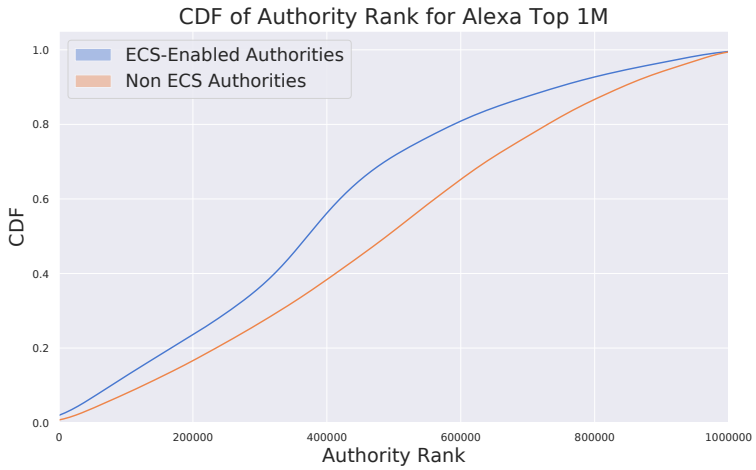


Figure 5.12: CDF of the authority rank for ECS and non-ECS enabled authorities. The authority rank is the average Alexa rank of the domains that this authority is authoritative for. The ECS-enabled domains are served by 19,133 authorities in June 2019, compared to 5,607 authorities in April 2015.

were not there in the past and serve different content, which justifies the difference over the years.

The majority of authorities Figure 5.12) in the 2019 dataset have ranks falling around the middle of the top million. In the 2015 dataset, the landscape is more balanced, with ECS-enabled authorities having almost a linear distribution over the possible ranks. This is reasonable since the authority ranks are averages, suggesting that many authorities today are shared by domains spanning multiple ranks in the Alexa top million. The prevalence of shared hosting and DNS services likely explains much of this behavior and is another example of the operational visibility capabilities that a large and consistent collection system such as *Thales* can provide with its depth of data over a few weeks of passive DNS collection information from a specific network.

This result highlights that there has been a clear trend in ECS adoption over the years. Even though ECS is presented as an optional standard designed to solve a particular issue, we have observed a steady increase in the adoption of this extension over time by a variety of websites without an apparent consideration as to whether the adoption of ECS would

offer a performance improvement to the website.

5.3.3 Client IP Subnet Information

We could observe clients' geographic and network locality using DNS logs generated from the authorities we discussed above. Since the authorities enabled ECS, recursives submitted ECS-enabled domain resolution requests, leaking the first three octets of the clients' IP addresses. Using this leaked client prefix, we could identify in greater detail the potential geographic location of the DNS requests rather than relying solely on the recursive IP. Furthermore, we could identify specific organizations and networks, many of the research institutes and government networks that were interested in the domain names in our sinkhole authority. This information was collected by solely operating an authority and would have been unavailable to us if it were not for ECS.

Client Geolocation and Network prefixes

In the absence of ECS, the visibility of the authorities would have been limited to the recursive IPs. However, when ECS is considered, we can observe a significantly better picture of the geolocation of the clients "behind" the recursives. Figure 5.13 shows the geographic distribution of the ECS-enabled recursives and the clients that resolved domain names in each of the authorities. More specifically, we were able to identify 180, 231, and 204 more countries in the sinkhole, in TLD and the DNS Zones authorities, respectively, when we considered the geolocation of the client prefixes in the ECS enabled requests compared to only taking into account the location of the recursive IPs. The source of the DNS requests can be traced back in greater detail when ECS is enabled.

When looking into the origin network prefixes of the requests, we identified some noteworthy cases of networks that queried our domain names in the sinkhole Authority. Among the requests received by our sinkhole domains, the prefixes 180.94.82.0/24 and 180.94.94.0/24 (two networks in Afghanistan and delegated to "AFGHANTELECOM

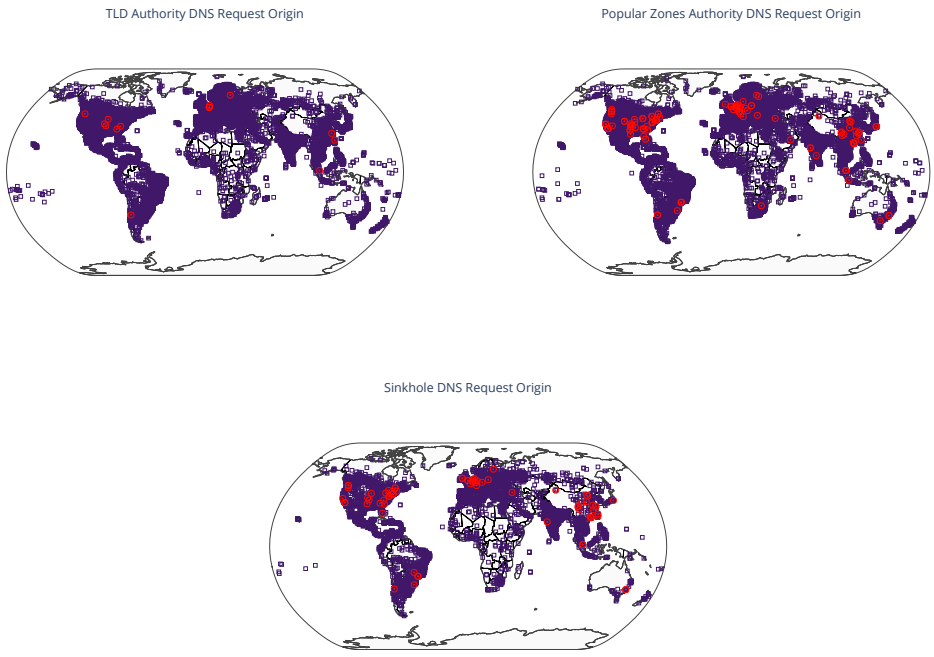


Figure 5.13: The distribution and density of the geographic location of the recursives and clients making ECS-enabled DNS requests to the authorities. The red dots show the location of the ECS recursives, while the location of the clients behind the requests are in purple. We can see that by considering the geolocation of the client prefixes, which is only available in the ECS-enabled requests, an authority is getting a much more granular view of the source of the DNS requests. For the TLD and DNS Zones, we calculate the distribution for a random day in June 2015 and June 2019, respectively, while in the sinkhole authority, we use the full dataset.

Government Communication Network.") appear to resolve two domain names related to APT activity in the past. Additionally, we also see frequent DNS queries to our APT domains from Academic networks, with 128.237.28.0/24 delegated to Carnegie Mellon University, 147.46.121.0/24 delegated to Seoul National University, and 171.67.70.0/24 delegated to Stanford University as some prominent examples. These DNS queries could be research related (e.g., by dynamically running malware that communicated with our sinkholed domain names) rather than infections. Finally, we also observe requests coming from Security vendors with 155.64.38.0/24 delegated to Symantec Corporation and 103.245.47.0/24 delegated to McAfee, making requests to both our APT-related domain

Table 5.3: Number of unique "/24" prefixes for the clients of ECS enabled requests and the recursives of legacy DNS requests for a random day in each dataset. We can see that in the TLD and the DNS Zones, the ECS-enabled traffic comes from more "/24s" than the traffic of the legacy DNS requests, even though the legacy DNS requests constitute the majority of the daily DNS requests.

	TLD Authority	**DNS Zones**	**Sinkhole Authority**
ECS client subnets	660,073	771,052	1,319
Recursive client subnets	218,944	166,374	4,151

names and typosquatting domains.

Prior to the introduction of ECS, we would have only seen that "someone is using GoogleDNS" to resolve these domains. With ECS, we could identify specific networks engaged in research, security vendors, and governmental activities. We remind the reader that all this information comes from DNS alone and is off-path of any TCP analysis. However, it could be argued that even with legacy DNS requests, authorities can have the same level of subnet visibility as clients that would not use ECS-enabled open recursives, such as Google's public DNS, would resort to a local recursive solution, thus still revealing their subnet to the authorities. In order to test this argument, we randomly chose one day of DNS requests from each authority. We looked at the number of ECS-client "/24" prefixes for ECS-enabled requests and the number of "/24" from legacy DNS requests after removing bogon IP prefixes from both datasets. Table Table 5.3 shows that we can see more client "/24s" in the ECS-enabled traffic for the TLD and the DNS Zones authorities than in the legacy DNS traffic, even though, as we have shown above, legacy DNS traffic makes up the majority of the daily DNS traffic for these authorities. While ECS only reveals up to 24 bits of the IP address of the clients, authorities can see a broader range of client subnets than legacy DNS. We can see that ECS can be used for extracting more granular information about the nature of the clients than in legacy DNS queries.

ECS Scope Size

Another interesting observation with respect to ECS-enabled resolution requests has to do with the size of the network the recursive is reporting to our authorities. We correlated the prefix reported by the recursives with the prefix containing the host addresses. From the eight billion ECS-enabled requests in our sinkhole authority, we could identify 75,138 unique prefixes for 99.8% of the requests. We could not identify 3,186 networks for 0.2% of the ECS-enabled DNS requests that correspond to IPv6 networks that are out of the scope of the current experiment and subnets that were not available on the Route Views [98] project database. From the available networks, 7,030 were ”/24” delegations, while 68,074 had a smaller prefixes. We also note 34 cases of networks with a bigger prefix than a ”/24”. Figure 5.14 shows the distribution of the prefixes to which IP addresses reported by ECS are delegated.

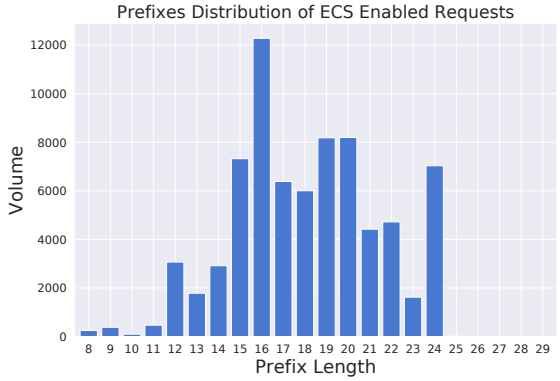


Figure 5.14: The distribution of the DNS resolution requests compared to the CIDR prefix length from where they originated. ECS could have provided the same level of service with the respective announced CIDR we see in the plot. Thus, the client’s /24 was submitted with no value for the client.

Regarding the distribution of the client prefixes that recursives forward to our sinkhole authority, we see that the significant majority of the ECS client prefixes (99.8%) are ”/24s,” and we do not observe any prefix less specific than a ”/24” exchanged. Although only the “/24” portion was forwarded for most of the clients, by considering Figure 5.14, the origin

of a request can be attributed to a *single* organization or ISP. On the other hand, we observe 130,261 queries in which the recursives respond with the full "/32" IPv4 address of the clients. Looking more closely at these queries, we see that the announced prefix for the corresponding clients is smaller or equal to a "/24". Thus there is no point in forwarding the full "/32" IPv4 address, and the last octet from all the "/32s" forwarded was "1". The vast majority of the recursives (118 out of the 122) exhibiting this behavior were attributed to different organizations in China and did not forward any prefix smaller than a full IPv4 address. We observe queries for all of the domain names corresponding to our authority from 432 different clients. Considering that these recursives do not forward any prefix smaller than a "/32" for all the IPv4 client addresses that they serve and the fact that the last octet is "1" for all the clients served, there is a high probability that these recursives are misconfigured.

To sum up, using passive DNS datasets, we showed that ECS-enabled traffic has made up a considerable portion of the daily DNS traffic in the past years. Contrary to legacy DNS requests, ECS-enabled queries provide more granular client information to authorities. This can be a valuable tool for researchers using DNS lookup data (e.g., running a sinkhole, as we have illustrated) to better understand the nature of the clients that are querying a domain name. Due to these reasons, we decided to add an option to include the relevant ECS flags in our Thales *collection during the recent updates. Note that the ECS fields in Active DNS do not identify clients, as they are under our control, but rather indicate if a recursive is set up to receive ECS data. Also,* Thales *can "spoof" ECS fields when querying, and this ability was utilized in our active probe insubsection 5.4.1.*

5.4 ECS speakers and CDNs

We have seen that ECS has a high adoption rate among domains in the Alexa top million. Given that the goal of ECS is to improve CDN performance by enabling more accurate identification of a user's location [9, 105], this obviously raises the question of whether

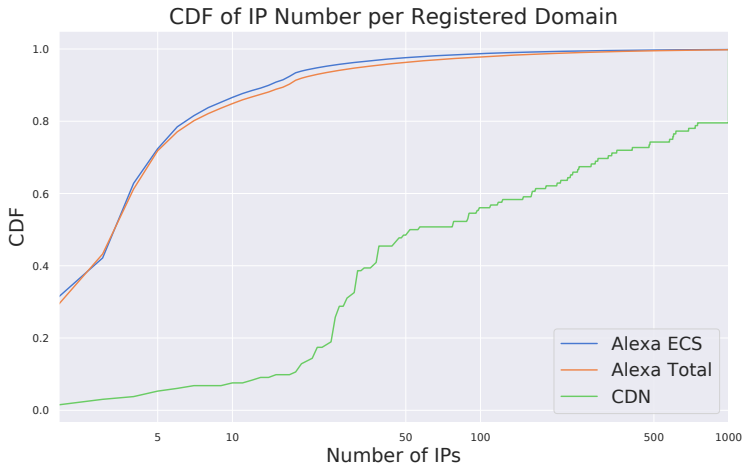


Figure 5.15: CDF of the number of IP addresses per domain name (log-scale) in the three datasets. The majority of CDNs have a much higher number of IP addresses than ECS-enabled domains and the average Alexa domain.

these domains actually use ECS to facilitate content delivery. As CDNs rely on servers in multiple locations around the world, we expect resolutions of ECS-enabled domains from different vantage points result in different IP addresses, exhibiting this way, a consistent CDN behavior. In this subsection, we will show that only a few ECS-enabled domains appear to resolve to more than a single IP address. Consequently, there is no real performance benefit for the vast majority of domains that currently support ECS in the Alexa top million.

In order to perform the experiments, we compiled a list of 133 verified CDNs by starting with a list of known, popular CDN domains and supplementing it with additional domains discovered in real-world network data. We used this set of CDN domains to make observations about the operation of the respective networks. Using the ISP DNS dataset collected by a large ISP in North America over the first five days of April 2019, we counted the number of IP addresses each domain (and CDN subdomains) resolves to, both in the CDN and the Alexa list. We observed that 50% of the verified CDN domains resolved to more than 50 distinct IP addresses in our passive DNS dataset. In sharp contrast, 80% of all ECS-enabled domain names from the Alexa top million resolved to less than seven distinct

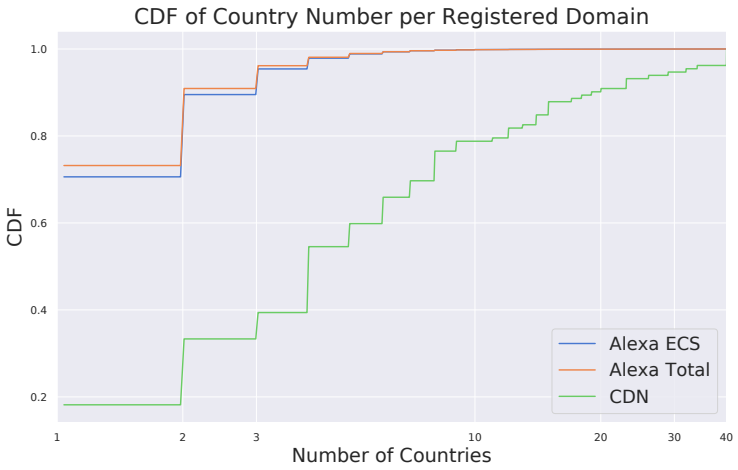


Figure 5.16: CDF of the number of distinct countries for IP addresses per domain name (log-scale) in the three datasets. CDN domains are distributed in multiple countries around the globe to better deliver their content, whereas ECS-enabled domains are mostly contained in the same country.

IP addresses, and less than 5% resolved to more than 50 IP addresses in the same passive dataset. Figure 5.15 shows the cumulative distribution function (CDF) of the number of IP addresses that a domain name resolves for each data set. Since the number of IP addresses for some domain names exceeded 400,000, we set an upper bound of 1000 IPs for each domain name for visualization purposes. Every domain with at least 1000 associated IPs was aggregated into a single group to make the resulting plots easier to read and understand.

We further correlate each IP address with its associated country of origin in order to provide a better understanding of the geographical diversity of the IP infrastructure that hosts each domain. For this purpose, we used the MAXMIND GeoIP2 [101] country database. For each domain, we counted the number of different countries it resolved to and presented the results in Figure 5.16. Considering the geographic location, less than 20% of the known CDN domains and the vast majority (70%) of the Alexa and ECS-enabled domains resolve to only a single country. Again for visualization purposes, we set the maximum number of distinct countries to be 40.

It is clear from Figure 5.16 and Figure 5.15 that most of the Alexa domains do not share

many of the CDN characteristics (the networks that ECS was originally proposed for), even though they eagerly support ECS. Also, behavior-wise, the generic Alexa domains, and the ECS-enabled Alexa domains display very similar attributes (although not exactly the same).

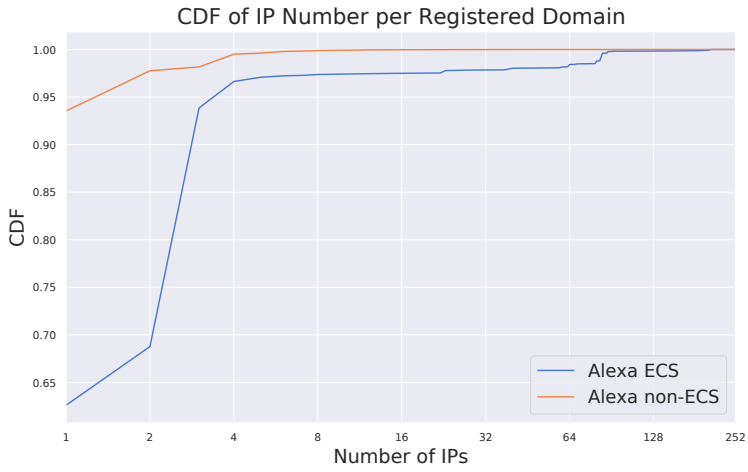


Figure 5.17: CDF of the number of IP addresses per domain name (log-scale) in our active querying experiment, notice y-axis starts at 0.625. The majority of Alexa domains have a very small number of IPs that they resolve to even when using ECS; in fact the majority, over 62%, only resolves in one IP, observing no benefit from the use of ECS.

In order to further examine the behavior of popular domains from Alexa that support ECS and to examine the utility of ECS for these domains, we conducted a further active data collection experiment with the purpose of demonstrating the variety of `RDATA` when ECS is used. For this experiment, we used some portions of the Active DNS infrastructure in order to collect the large amount of data we needed for this operational experiment. We take the entire Alexa 1M list of domains and submit queries using a modified resolver that allows us to specify the ECS client prefix we will send to the authority nameserver. For a list of geographically diverse IP addresses to use as ECS prefixes, we utilized the publicly available AWS IP ranges, which provide us with actual IPs that are geographically diverse, as is the AWS infrastructure subsection 5.4.1. We repeat the experiment for 26 different IP ranges and present them in Figure 5.17 and Figure 5.18.

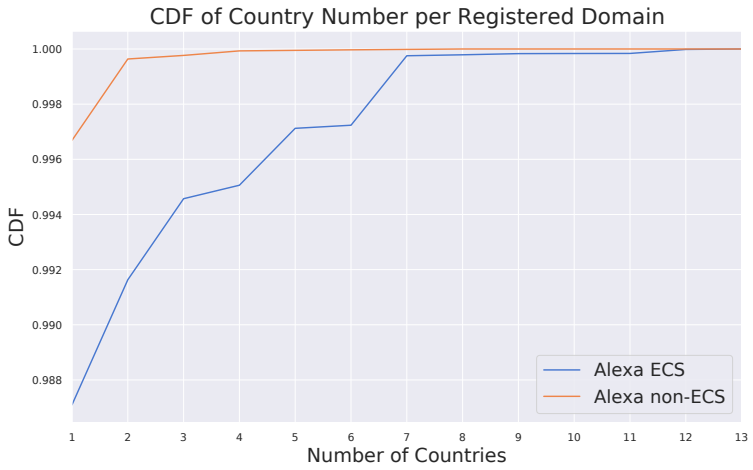


Figure 5.18: CDF of the number of distinct countries for IP addresses per domain name (log-scale) in our active querying experiment, notice y-axis starts at 0.98. In terms of variability in the country that's hosting the domain, Alexa domains exhibit even less variability and are in line with our passive measurements.

5.4.1 Active probing subnets

It is very clear from Figure 5.18 that the overwhelming majority (over 98%) of Alexa domains are hosted in only one country. This means that there are small, actual geographical benefits from using ECS, even when we query the domains from client subnets that correspond to locations all around the world. Similarly, Figure 5.17 verifies that our passive DNS measurements are consistent with our Active global measurements and shows that the hosting infrastructure of popular Alexa domains is not particularly diverse, especially compared with the CDN diversity we observe in Figure 5.15. Another observation we can make here is that due to the proactive nature of our Active DNS collection, we can ensure that we have varied geographical ECS coverage without potential bias when compared to the large but geographically concentrated passive DNS dataset. The vast majority of Alexa domains, even those that support ECS, only utilize one IP address, and 95% of them use less than four IP addresses. It also shows an interesting use case for Active DNS domain data, as it can be adapted to query for specific types of records, proving it's operational research potential. Due to its size and customizability, it can provide global query coverage

for millions of domains. An experiment of this kind can only be performed with active data in order to be able to control the queries and compare the domain names under investigation while also making sure that we have complete coverage for all geographic regions under investigation.

Based on the observed behavior, it appears that the benefit from using ECS is not significant (or apparent) for many of these domains. This reinforces our intuition that ECS is sometimes misused. It is also apparent that even in the case of the limited number of Alexa domains that point to many IP addresses, these IPs are not necessarily located in different places around the world. On the contrary, most of them can be found in the same region. In these cases, the users' anonymity could be waived without benefitting them. Given that, in the following subsection, we will measure the diversity of the infrastructure of these domains to understand how users' information travel during a DNS resolution request and in which cases other entities can obtain this information.

5.4.2 Infrastructure Diversity

To present how different entities are involved in the resolution of a domain, we analyze the infrastructure that hosts ECS-enabled domain names. Since routing on the Internet is based on Autonomous Systems (AS), BGP announcements, and peering agreements between ASes, we focus on the distance of the ASes that host ECS-enabled authorities from the ASes that host the respective services for those domains (i.e., `RDATA`). Ideally, we would want to know the different hops a packet will make before reaching the authority and the respective service. However, network packets are expected to take several different paths, depending on factors like peering agreements, congestion, load balancing, etc., which make it particularly hard to predict [106, 107, 108, 109].

In order to demonstrate that the DNS packets traveling to an authority are likely to take a different routing path from consecutive communication with the actual service the domain offers (e.g., web server that serves HTML context), and therefore reveal information about

Table 5.4: CIDRs and their respective countries and regions selected for the active probing of the Alexa 1M domains for ECS optimized responses. The CIDRS are networks belonging to Amazon AWS based on publicly available data. The countries are geolocation of the CIDRs based on Amazon's published network information, available at: https://docs.aws.amazon.com/general/latest/gr/aws-ip-ranges.html

CIDR	**Country**	**Region**
150.222.81.0/24	Ireland	West Europe
64.252.84.0/24	United Kingdom	West Europe
52.95.224.0/24	Spain	South Europe
52.94.18.0/24	Spain	South Europe
52.219.168.0/24	Germany	Central Europe
64.252.88.0/24	Germany	Central Europe
13.248.100.0/24	Sweden	North Europe
15.177.72.0/24	Sweden	North Europe
150.222.78.0/24	Singapore	Southeast Asia
64.252.104.0/24	Singapore	Southeast Asia
13.248.117.0/24	India	South Asia
150.222.235.0/24	India	South Asia
52.95.226.0/24	Hong Kong	East Asia
54.240.241.0/24	Hong Kong	East Asia
15.221.34.0/24	Japan	Northeast Asia
150.222.116.0/24	South Korea	Northeast Asia
15.230.137.0/24	United States	North America East
13.248.103.0/24	United States	North America East
99.77.132.0/24	United States	North America West
52.95.247.0/24	United States	North America West
15.230.138.0/24	South Africa	South Africa
52.95.180.0/24	South Africa	South Africa
99.77.147.0/24	Bahrain	Middle East
13.248.106.0/24	Bahrain	Middle East
64.252.78.0/24	Brazil	South America
150.222.12.0/24	Brazil	South America

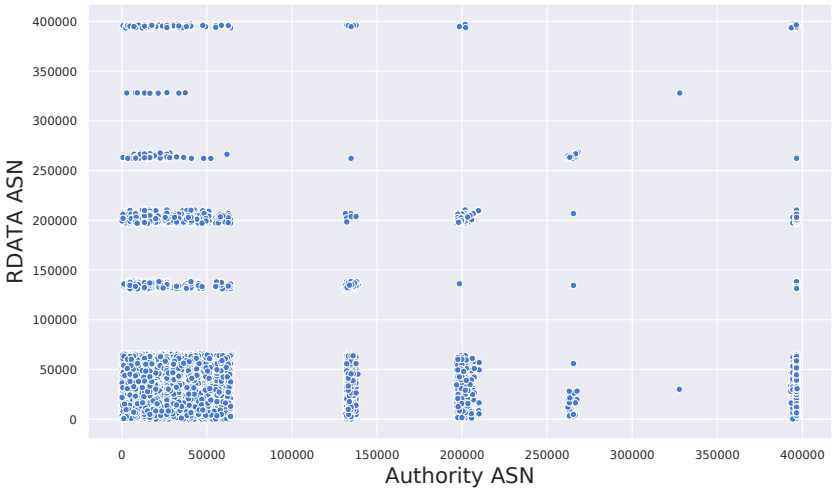


Figure 5.19: Scatterplot of the Autonomous System Number (ASN) where the authority's IP address is being announced from and the ASN where the `RDATA` for a domain name resides into. The diagonal corresponds to authority-domain pairs that reside in the same Autonomous System.

the client to multiple other entities, we base our analysis on the ASes that host the authority and the returned IP address for an ECS-enabled domain. We also show that there are entities positioned on the path between a global recursive and the authority of multiple domain names, which are in a position to collect all clients' information just from DNS resolution requests.

For a given ECS-enabled domain name, if both the authority and the respective `RDATA` [110, 111] — referred to hereon as the *service* — are hosted within the same AS, then there is a probability that DNS leaks will be limited to the same path as the TCP connection that will follow. Currently, however, it is often the case that the DNS packets will take a completely different routing path than the subsequent service connection (as shown in Figure 5.1). In that case, the ECS subnet information is leaked to all ASes between the resolver and the authority.

When a given ECS-enabled domain is served by an authority in a different AS than the service, then inevitably the DNS packet will take a different route and pass through

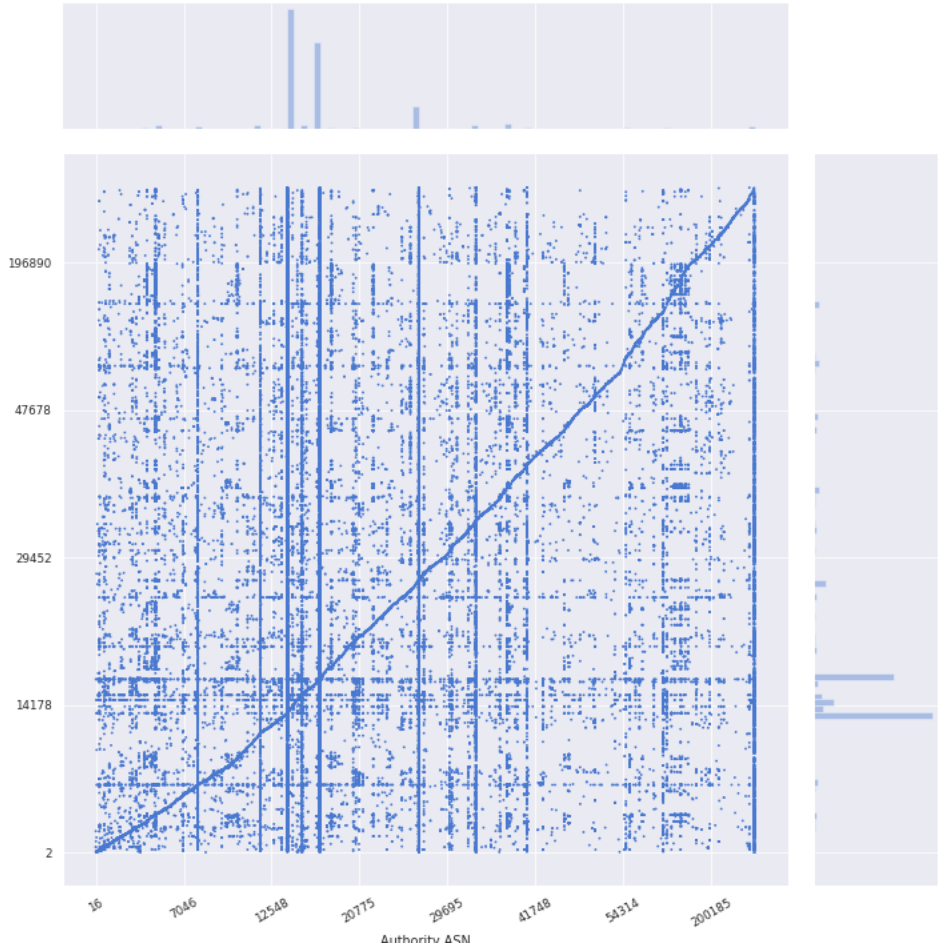


Figure 5.20: A different visualization of Figure 5.19 showing the joint distribution and collapsing the empty space. This distorts the diagonal because different ASNs are present on each axis. The diagonal is now a crooked line.

at least one different AS (that of the authority). Thus, in the best case, one more AS will have information about the client (in reality, more ASes are likely to exist on the path). Figure 5.19 shows the relation between the AS of the authority (x-axis) and the AS of the service (y-axis). The diagonal on the plot depicts cases where the authority and the service are located within the same AS. To generate Figure 5.19 and Figure 5.20, we used the 2019 Alexa data from subsection 5.3.2. We associated each authority and domain IP with the ASN according to the Route Views dataset [98].

One thing that stands out from Figure 5.19 is that this kind of visualization is prob-

lematic because there is a lot of empty space (i.e., ASN numbers that are not used either in the authority or the RDATA axis). For that reason, we also created Figure 5.20, which is practical Figure 5.19 without the empty space (collapsing the plot to only include valid data points). Since each of the axes has different ASes (because some of the authority ASN may not have complete overlap with the RDATA ASN and vice versa), the actual diagonal is comprised of different ASNs per axis. However, if both the ASN for RDATA and authority are the same, it will be a dot near the diagonal. The ideal diagonal now looks like a crooked line but still stands out.

For reference, we examine some of the top 10 authority ASNs (that are related to the most RDATA ASN). Namely, AS 13335 belongs to CloudFlare, AS 26496 to GoDaddy, AS 16509 to Amazon, and AS 396576 to VeriSign. Obviously, these organizations are affiliated with CDNs, cloud services, and domain name registrations (and thus parking), and that behavior is expected. In any case, every point in the plot apart from the diagonal in Figure 5.19 and the "crooked" diagonal in Figure 5.20 corresponds to cases where the RDATA (webserver) and the authority (DNS Server) ASN are separate. Therefore there is potential information leakage to a different entity. There is no arguing that in the above cases, this occurs predominantly. The outsourcing phenomenon is a characteristic of the modern web.

Finally, to estimate the potential of a leak when a DNS resolution request arrives at any of the authorities, we measured the number of ASes that the authority's AS peers with. We use the Shadowserver [112] API to identify peers for the CIDRs that announce the IP addresses of the authorities for ECS-enabled domains. Figure 5.21 shows the distribution of peers for each domain name. The majority of the domains are served by authorities that are located in ASes with three, four, or eight peers. Any of those peers, along with other ASes until a packet reaches them, is a potential collector of activity from ECS-enabled DNS packets.

Essentially, we find that a large number of the domain names that utilize ECS use third-

party DNS providers. This means that the DNS infrastructure of these domains resides in a separate network with a different AS and administrator. Thus, the IP information included in the new ECS-enabled DNS packets is shared with third parties unknown to the client for no immediately discernible reason. When considering the lack of a diverse hosting infrastructure for these domains, there is no benefit from enabling ECS. Similarly, ECS-enabled domains provide IP information to third parties on-path during the resolution process. This partial information (e.g., a /24) would otherwise be unavailable to anyone other than the recursive itself.

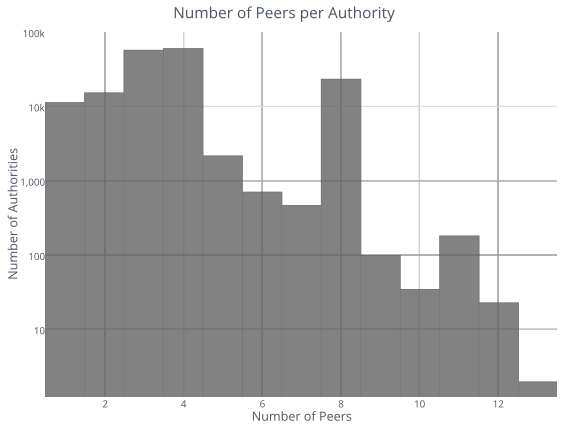


Figure 5.21: The distribution of the number of peers per Autonomous System that hosts an ECS-enabled authority. The vast majority of the authorities reside in ASs that have three, four, or eight peers, which can be potential alternative paths for a DNS resolution request and one more collection point for entities involved.

5.5 Discussion and Summary

5.5.1 Discussion

Considering that currently, years after its initial introduction, ECS is still enabled by default depending on the recursive used, the user has limited ability to control the amount of information shared using ECS, and so we would like to discuss the options available to the users. The RFC mentions that the user can signal the maximum resolution of the scope netmask that can be used by setting it in the initial request to the recursive, and the recursive

should follow the resolution that the user's stub set. By setting a scope netmask of /0, the user can effectively opt out of using ECS while also not taking advantage of the benefits that ECS provides. Another option that the user has is to set a netmask more coarse than the default used /24 resolution. That will balance privacy and allow for more content delivery optimization by services that benefit from ECS. The issue with this approach is that no user-facing stub resolver currently allows for this setting level. Support for ECS scope netmask setting needs to be added to stub resolvers. Another potential issue is that currently, not all the recursives implement the RFC correctly but default to a different netmask, disobeying the netmask set by the user, similar to cases mentioned in subsubsection 5.3.3.

From the side of website operators, we can only comment that they should only enable ECS responses when they perform some form of traffic optimization. Considering the number of domains that seem to support ECS but do not benefit from the protocol, we believe that a large number of managed domain hosting enables ECS by default. Another interesting approach is the discussion around more privacy-minded ECS implementation that was presented in a publication by NextDNS's Olivier Poitrey [113]. This approach relies on the geographical awareness of the Autonomous Systems that the recursive resolver serves and depends on providing a geographically relevant IP portion to the authority instead of the user's IP address portion. As for the more privacy-conscious user, the standard privacy-preserving methods of browsing the web, such as VPNs and the Tor network, will still provide the user the ability to hide their IP from an ECS-enabled authority. A more straightforward solution would be to manually set the DNS servers that the user prefers and thus choose a set of recursive resolvers that do not send ECS information. On the other hand, with any of these solutions, the user will not be able to take advantage of ECS's benefits.

5.5.2 Conclusions

In this chapter, we presented a longitudinal study measuring the adoption of a DNS extension called ECS. Given the widespread usage of DNS in IP-based networks, our work aimed to identify how changes introduced by this extension affect network communications that rely on DNS. This analysis serves as a case study that explores the good and bad unintended consequences of introducing small changes to fundamental network protocols.

The primary goal of ECS was to optimize CDN selection through DNS, but our analysis found that most sites in the Alexa top million do not benefit from ECS (Section section 5.4). This result demonstrates how new functionality may not always get used as intended, so it is essential to consider potential unintended consequences. For example, we identified that most authoritative DNS servers using ECS adhere to the proposed defaults and set an IP subnet mask of /24 (Section subsection 5.3.1). The use of small subnet masks results in sharing of fine-grained client information with DNS nameservers above the recursive DNS server. We found that most ECS-enabled domain names outsource their DNS infrastructure (Section subsection 5.4.2). As a result, more networks now have fine-grained client information for DNS on-path communication. Thus, we find ECS potentially exacerbates the effects of existing threats such as DNS leaks.

These consequences raise questions about the scope of impact. Our analysis finds that, despite being optional, ECS has seen steady adoption over time (Section subsection 5.3.2). Thus, the unintended consequences of ECS are not limited to a small subset of Internet communications. As a result, authoritative DNS servers—and all DNS nameservers above the recursive, for that matter—now have visibility of the client networks querying them. This client information enables DNS operators to track client networks and user behaviors in ways that were impossible before ECS (Section subsection 5.3.3). At the same time, this same information can also help security practitioners track new threats or aid remediation efforts when local network visibility is limited. That is why we decided to include the relevant fields in our Active DNS infrastructure capabilities. We also showed that, especially

when combined with other datasets, such as Active DNS data, ECS has the ability to unlock further types of security research, such as tracking of security threats based on infrastructure and providing another dataset that can be used to perform forensic investigations of incidents.

Ultimately, we find that ECS has impacted a large volume of DNS traffic on the Internet. It is widely deployed and used by domains all across the Alexa top 1 million. As a result, security practitioners should be aware of its pitfalls and potential uses for good.

CHAPTER 6
CONCLUSION

In this thesis, we presented *Thales*, our scalable active collection infrastructure system, alongside documenting the collected dataset, which we call Active DNS. We then show how any researcher can utilize Active DNS data as well as information enrichment methods that enable us to expand our dataset's visibility. We have demonstrated that an active DNS query system provides wide coverage of the domain space and can operate reliably for many years while also reliably capturing a DNS dataset that allows us to conduct a wide variety of research projects. We have already documented our work on similar projects, such as the TSS enrichment, and the various case studies, such as studying the adoption of ECS. These are all examples of projects that were only possible with the vast breadth of data points and global visibility that our Active DNS dataset provides. Our findings confirm our original hypothesis that we can design a system that can deliver a DNS dataset that enables reliable and repeatable research while at the same time removing many of the restrictions that come when obtaining Passive DNS datasets —monetary expense, privacy implications— to conduct network security research. We hope this work inspires other researchers to explore and utilize our dataset while discovering new ways to utilize DNS data and encourage further experiment repeatability within the network security community.

6.1 Overall Contribution

This work focuses on documenting the system's creation, operation, evolution, usage, value, and the datasets derived from what we call Active DNS. DNS data have been used in the network security community for years, but various restrictions around passive DNS datasets have spurred us to develop a new solution. Our solution, Active DNS, presented here thought this thesis focuses on providing a wide-reaching DNS dataset without any

legal or other restrictions in the hope that it will assist in future research using DNS data and will also bring in a new way to verify the results of already published research using a consistent and well documented open dataset. We have shown how actively collected DNS data has various applications in security research. Some of those applications overlap with passive DNS datasets, while other uses rely on the strengths of Active DNS to yield reproducible results by anyone with access to similar data.

6.2 Considerations and limitations

A number of limitations govern any system constructed in the world. In our case, we are bound by the limitations of the collection system itself, as well as limitations, considerations, and policies around active scanning and probing of Internet resources. While we try to present the value of our system and its various uses, we also have to document the limitations we had to face while working on assembling this thesis. The following subsections explain the limitations of the works we presented.

6.2.1 Limitations of Active DNS collection

This study implements an active querying system capable of generating a dataset that makes many existing internet research measurements more accessible and repeatable while enabling new paths of exploration. However, the limitations of our system must be understood by those wishing to incorporate its results into their research.

System Architecture (Thales): Seed generation is a primary limiting factor in generating an Active DNS dataset. *Thales* will only issue queries for domains included in its seed list. For example, our seed list is composed primarily of e2LDs. As a result Active DNS dataset contains a very small amount of child labels. The limited amount of child labels found in Active DNS data comes from our inclusion of OSINT and partner security data. Further, some zones, for example, many country-coded TLDs, do not make their zonefiles public, preventing us from enumerating all second-level domains under those zones. While

we can add additional seed lists to feed *Thales*, we must be careful not to cause unnecessary network congestion or overwhelm servers in the upper DNS hierarchy.

Another limitation to our dataset comes from the operation of *Thales*, which limits each domain's observations to once or twice a day. That limits the number of data points per day and prevents a more timely observation of changes in the domain and its resolution. This limitation can be engineered to be less impactful for lists of domains that we know change rapidly by assigning them to multiple queues per day but that would rely on a specific target.

Thales also operates in a single physical location, limiting its ability to capture divergent behavior related to geography. Passive DNS collected at a network such as a university campus would be similarly limited. Operating multiple instances of *Thales* in different regions could mitigate this in the future.

Finally, many upgrades have been made to *Thales* since we initially deployed the system. These changes have helped to make the system more portable, scalable, and reliable. However, the automatic collection of health statistics for *Thales* and corresponding alerts have yet to be implemented. Including these in the system could increase the system's reliability by more quickly alerting operators in the event of an outage.

Active DNS dataset: Even though we have demonstrated that our Active DNS dataset has some comparable qualities to existing DNS datasets, we have also documented that it is not a complete collection of all domain name associations. The first and most crucial point is one we have made sure to point out throughout this document, and that is that our dataset does not contain any client data or information, as is the case with traditional passive DNS, because all domain resolutions originate from our Active query system, *Thales*. As is the case with any DNS dataset, there may be many more threats and indicators of abuse that are not detected by any of the methods described in this work. Furthermore, another important area where Active DNS data differs from passive DNS datasets and mainly the significant lack of domain child labels. As we have explained in chapter 2, and due to the nature

of how the system actively queries for domains, we have no way to obtain a complete view of a DNS zone. In traditional Passive DNS data, any DNS name that traverses the network will be captured and recorded; thus it will contain a number of child labels that were utilized in serving the user. Since Active DNS does not interface with any other portion of the server infrastructure other than the DNS, it can only obtain information for the domains that *Thales* already contains in the seed or were the result of a reply from the DNS authority.

Any attempts to work around this limitation would require that the collection system behaves in one of three ways: Either have pre-existing awareness of all the labels in a zone, interface in an effort to expose the zone, or randomly attempt to guess potential domain labels. Obviously, having knowledge of all the zone contents would be ideal, but unfortunately, there is no current way to reliably obtain that information. Interfacing with the DNS server in an effort to obtain a full picture of the zones that it serves would be a limited solution. DNS servers do have a query type to request zone information through a zone transfer (`SOA`) type query or `AXFR` type query. Unfortunately, as we have noticed, very few operators reply to these types of queries, and even fewer provide actual zone information to third parties outside their network. Finally, attempting to guess child label zones would be an easy-to-implement solution but one that will have to be based on some type of dictionary system to try and "predict" potential child labels. This would obviously make the system inefficient and would potentially cause issues for DNS operators due to the large number of random queries that they have to reply to. Effectively, if not implemented correctly, this approach can lead to a DDoS attack against DNS authorities.

6.2.2 Limitations of exposing TSS scammers

Like all real-world systems, the work we performed in chapter 4 Other than the technical limitations that are described in the original work by Srinivasan [8], such as the choice of using PhantomJS for the active web crawling search results and ads can, in principle, be

detected by scammers who can use this knowledge to evade our monitors.

Furthermore, for the enrichment performance of the system as well as the performance in the case studies presented in section 4.3 the main limitation is the potential of a similar system to miss data points that would prevent it from being connected to the existing data from datasets from which we already have knowledge. This situation can be mitigated by increasing the size and reach of our initial domain seed. Still, it is possible, due to the nature of Active DNS data to miss data points when they are connected with domains that have several levels of delegation and child labels. Of course, a Passive DNS dataset would also be impacted by similar limitations if the activity on the network does not contain sufficient data points for the amplification and eventual detection of threats.

6.2.3 Considerations about ECS and the impact

Given how large the use-case of streaming large videos and other bandwidth-intensive types of media over the Internet has gotten in the past five years, we attempted to examine the potential impact of one of the proposed protocol changes to DNS operation that would primarily benefit Content Distribution Networks, which are essential for the delivery of all these streaming-data. Our study focused on the ECS extension of the DNS system, which was designed as a method to allow CDN networks, and content delivery providers to more effectively identify the approximate geographic location of a user, even if the user is employing a large open recursive, by providing the DNS authority with a portion of the requesting user's IP address. We presented a longitudinal study measuring the adoption of ECS in chapter 5 over the years up to the point where ECS was accepted as an RFC. Given the widespread usage of DNS in IP-based networks, our work aimed to identify how changes introduced by this extension affect network communications that rely on DNS. This analysis serves as a case study that explores the good and bad unintended consequences of introducing small changes to fundamental network protocols.

In order to be able to conduct this research and verify the behavior of the various re-

solvers that reply to queries for the most popular websites, which we would expect to direct users to different infrastructures based on user IP, we generated and sent millions of queries with spoofed User IP sections in the ECS extension. The results of these experiments are discussed in more detail in section 5.4 and subsubsection 5.3.3.

As we mentioned, the stated primary goal of ECS was to optimize CDN selection through the use of DNS. Still, our analysis found that most sites in the Alexa top million do not receive any measurable benefit from ECS (Section section 5.4) or even attempt to answer differently to our specialized queries. This result demonstrates how new functionality may not always get utilized as intended, and therefore, it is essential to consider potential unintended consequences that the new functionality might uncover. For example, we identified that most authoritative DNS servers using ECS adhere to the proposed defaults and set an IP subnet mask of /24 (Section subsection 5.3.1). The use of small subnet masks results in the sharing of fine-grained client information with the DNS nameservers above the recursive DNS server. We further found that most ECS-enabled domain name owners outsource their DNS infrastructure (Section subsection 5.4.2). As a result, more networks can now provide fine-grained client information for DNS on-path communication. Thus, we find that ECS potentially exacerbates the effects of threats such as DNS leaks.

These consequences raise questions about the scope of the impact. Our analysis finds that, despite being optional, ECS has seen steady adoption over time (Section subsection 5.3.2). Thus, the unintended consequences of ECS are not limited to a small subset of Internet communications. As a result, authoritative DNS servers—and all DNS nameservers above the recursive, for that matter—now have visibility of the client networks querying them. This client information enables DNS operators to track client networks and user behaviors in ways that were not possible before ECS (Section subsection 5.3.3). At the same time, though, this same information can also assist security practitioners in tracking new threats or aid remediation efforts when local network visibility is limited.

Ultimately, we find that ECS has impacted a large volume of DNS traffic on the Internet.

It is widely deployed and used by domains across the Alexa top million. As a result, security practitioners should be aware of its pitfalls and potential uses for good.

6.3 Future Work and Improvements

Considering the operational scope and expansive capabilities of *Thales* as well as the quality and wealth of our dataset Active DNS, we believe that there are countless improvements and adaptations that can be made to further expand on the capabilities of the infrastructure and of the depth and wealth of data collected. Because we believe that the work we presented in this thesis and the ideas and lessons we learned from the development of *Thales* and the study of Active DNS data can be an important branch in our field of academic study and so we want to assist anyone that wants to continue the work we started by providing some ideas for future work. These ideas can be split conceptually into two broad categories, that of further study based on Active DNS data and the other category based on improving the *Thales* collection infrastructure.

We will first present some of our future work ideas that concern the collection infrastructure and how it can be improved or adapted for more specific uses in future works. One simple approach would be to keep expanding the resources and the domain seed of the existing infrastructure, yielding a more representative view of the global web. Such a simple approach would require a more expansive domain seed as input and an expanded worker infrastructure but is well within the scope of the original system design for *Thales.* Another potential future extension of *Thales,* is that it can be utilized to collect an almost complete list of PTR records for the IPv4 space. In a way, similar to other active scanning projects such as *zmap* [114], and Censys [67] *Thales* can be configured to dynamically send PTR record requests for every IP in the IPv4 network range or a predetermined IP range. The resulting dataset would include all reverse network delegations that are publicly resolvable in the IP range. Another way to optimize this type of collection would be to collect AS information from public lists and only send PTR requests to the AS delegation point. That

would limit the number of requests needed to be sent as well as the number of requests the servers would have to reply to, thus making the operation much more sustainable. The main issue with these approaches is that they would necessitate a large increase in the number of queries that *Thales* can send and process per day. Furthermore, such an extension could also operate as its own collection system that focuses on collecting network mapping information, and the dataset could be separated that the traditional, forward DNS resolutions to make the data it collects more easily searchable.

Another option for the expansion of the collection system would be to modify it in a capacity that would provide a more customized dataset as a result. A use case such as this would be useful for researchers that operate honeypots or even researchers that operate malware analysis systems. Such operations generate a number of DNS queries, but they are only captured as part of the analysis process. We propose that a system such as that can benefit from curating a list of domains "seen" during the dynamic execution of a piece of malware. Then an instance of Active DNS can operate and generate a dataset that will contain active queries for these domains and create a dataset that includes long-term longitudinal observations of these interesting domains and their behavior through a large period of time.

A future study could potentially utilize a modified collection infrastructure that distributes different portions of the DNS delegation tree as seed input for each worker, with the goal of mapping the impact of seed changes on the resulting dataset. This would involve changes to the current collection infrastructure, as the current system is optimized for storage efficiency and does not keep redundant records. Such a study would be novel in that it would allow for the exploration of unknown delegations and connections between the highest levels of DNS authority, providing insights into the impact of seed input on the resulting dataset. It would also allow for the mapping of DNS resolution delegations and non-typical resolutions resulting from the use of `CNAME` or similar out-of-tree delegations. Overall, this study would provide valuable insights into the DNS system and allow for a

better understanding of the impact of seed input on the resulting dataset.

On the other hand, one could focus on improving the performance of the system without adding a number of new sources in an effort to build an even more complete map of the IP/domain associations. That could be achieved by further optimizing the seed list and the query instructions for each worker by building feedback systems into *Thales* that can curate the seed list based on the responses that the system has received previously for the same domain. As an example, simply the removal of NXDOMAIN-associated domain names from the seed list of *Thales* and an improved queue system improved the number of RRs collected by the system two-fold. Similar optimization can be offered for the various `QTYPEs` that the system queries for, where based on previous results, the system does not make daily queries to domains that do not offer a response to that type of query. That would enable an even faster and more optimized query queue, that would result in likely more daily observations, more than the current two, for the active domains in the seed.

Another approach would be to utilize *Thales* as a way to probe resources that show activity through a network. Extracting a list of domain names and query types from security appliances such as IDS and inputting this dynamic and changing list as part of the domain seed of *Thales* would yield potentially interesting results as well as include a user activity component into the Active DNS dataset. After operating such a system, the operators would then have to choose how long the observed domains will be retained in the system and when to retire entries from the seed. Of course, there would be a need for some filtering of the new dynamic input into the system in order to avoid sending excessive amounts of requests to the same system operators, but our current implementation of Thales can already support that kind of dynamic queue management.

The logical progression of such features leads to the implementation of a query system that would allow researchers to quickly query the Active DNS dataset in real time through a web portal. An application such as this would not change the underlying dataset, but in our experience, it would enable researchers and other security professionals to quickly monitor

and identify potential new threats, or just examine the health of their network.

These some examples of how expandable the concept of Active DNS data collection is. Now we will present some ideas for future work that primarily focuses on the analysis of our Active DNS dataset or even some of the resulting datasets from the proposed expansions.

As, we have pointed out in chapter 3, one of the unique characteristics of Active DNS datasets, is that they include consistent daily records of various "uncommon" `QTYPEs` that still encode and contain a large amount of potential information that is yet to be studied. We are referring to the record types of `SOA`, `TXT` and `MX`, which are not normal products of a recursive resolution for an IP address, and thus the associations with the domains do not usually exist in a passive DNS dataset. As we've shown in subsection 4.3.1, these types of records have great potential in providing yet another signal for activity detection of any considerable kind, from phishing campaign activation and domain ownership change to studies of cryptographic practices and policies that utilize DNS records. Effectively, these types of records that are found in abundance and recorded consistently by *Thales* can provide network forensics investigations and studies with yet another signal to enhance other datasets and make more accurate and confident observations. As this is a unique, novel dataset that is still being propagated in the research community, we can only anticipate the new types of information extraction that these often overlooked `QTYPEs` might provide.

Based on the points made in the previous future work proposal, we can give some more specific examples of future work that can use the already available Active DNS dataset to investigate the case of distributed denial-of-service (DDoS) attacks using DNS. Based on the conclusions of existing works in the space [115, 116, 117] we know that a large number of DDoS attacks on the Internet today come from what we call DDoS amplification attacks. In a DDoS amplification attack, the attacker exploits vulnerabilities in the design of certain Internet protocols to amplify the amount of traffic that is sent to the target. In the case of "DNS amplification," by sending a small number of requests with a spoofed source address

to a DNS server, the attacker can cause the server to send a much larger number of responses to the target, overwhelming it with traffic. Because our Active DNS infrastructure contains a large number of special `QTYPEs` such as `TXT` that are more likely to be implicated in a DDoS attack, it is possible for researchers to study the phenomenon and sizes of special `QTYPEs` in the DNS hierarchy. This capability can be further expanded by including more special `QTYPEs` for *Thales* to query, thus increasing the potential visibility for these packet types. Some of the special `QTYPEs` that are candidates for such an expanded collection include the `DNSSEC` extension types, as they have large sizes and have been previously implicated in DNS amplification attacks.

A further adaptation to the dataset would be to operate in a way that produces multiple output datasets. That can be as a consequence of each dataset using a different seed input or as a result of different levels of post-processing and deduplication upon the commonly generated Active DNS dataset. In effect, an operator can utilize multiple input seed lists that would generate separate output datasets, with different data points contained. The other approach would be to offer datasets with variable detail in some of the DNS records. An example of such an approach would be to generate a condensed dataset that only contains the differences between observations of various domains, with first-seen and last-seen data fields informing the time periods between changes. Such an approach would lead to a smaller, more portable dataset that can be utilized as a first step to identify if a resource is contained in the larger dataset before committing to spending the time and resources of downloading the full dataset.

But our Active DNS dataset is not limited to only research topics around security but can support more broad network measurements and even assist in drawing observations that will improve our understanding or such distributed systems.

One novel idea, for future research using the Active DNS dataset would be to investigate the use of DNS for distributed storage systems. By analyzing the record of DNS associations recorded in Active DNS, a researcher could study the use of DNS for future

or already proposed distributed storage systems. This could involve identifying patterns in DNS traffic and use by existing storage systems such as Amazon's S3, or that indicate the presence of distributed storage systems, and developing algorithms for detecting and tracking such systems.

Similarly, Active DNS datasets, can also be utilized in Investigating the role of DNS in the operation of the internet of things (IoT). As we know IoT devices have exploded in popularity in the past years and have already had a profound impact in our space, making possible one of the largest DDoS attacks in Internet history by utilizing IoT devices infected by the Mirai [117] Botnet. By analyzing the record of DNS associations in our dataset, a researcher could study the use of DNS in the operation of IoT devices. This future work could involve identifying patterns in DNS traffic that indicate the presence of IoT devices or developing algorithms for detecting and tracking IoT-related DNS activity. In an effort to improve the community's understanding of the topic.

Finally, one other type of future work is to more broadly Investigate the use of DNS for load balancing in large networks. By analyzing the record of DNS associations in our dataset, a researcher could methodically study the use of DNS for load balancing in large networks. This could involve identifying patterns in DNS traffic that indicate load balancing and developing algorithms for optimizing the distribution of DNS queries across multiple servers.

In conclusion, the potential for further research in the fields of Active DNS data and further development for the *Thales* collection infrastructure is vast. By continuing to build upon the foundation laid out in this thesis, we can make significant strides in understanding and improving the field of academic study. We hope that the ideas and suggestions presented in this section serve as a helpful starting point for those interested in pursuing further work in these areas, and we stand ready to offer assistance and support to anyone seeking to continue the work we have begun.

6.4 Closing Remarks

Working on the projects that culminated in this thesis taught me how to utilize distributed systems in order to actively query millions of domains names daily, with the goal of creating a dataset that is capable of assisting researchers in identifying, studying, and better understanding security threats while at the same time enabling a number of other uses for a variety of Internet researchers. The first study of this thesis shows how we can provide an alternative to Passive DNS datasets for many areas of network security research. We show how we can achieve this at scale and provides the result as a free dataset to the research community, with many outside researchers already utilizing the dataset. One of many specific strengths of our system is its ability to map infrastructure associated and can often provide more detail than Passive DNS for this task. As just one example, the second study in this thesis utilizes the dataset from Active DNS, along with active search engine queries, to identify the infrastructure of Technical Support Scam schemes. Using the Active DNS dataset enables us to enhance our information by revealing other infrastructure that the scammers have under their control but is not revealed by search engine queries. This study was able to provide a systematic and comprehensive analysis of the TSS ecosystem while also demonstrating the real-world value of Active DNS. Finally, the last study we present here also takes advantage of our Active DNS dataset and aims to examine the adoption and potential effects that a relatively new DNS extension known as ECS can have for the users of the Internet as well as for security researchers. Due to its nature as an extension, we show that ECS operates mainly on Authority and Recursive Nameservers, so end users are usually entirely unaware of it. At the same time, it still generates data that can be captured on additional networks that include a sizable portion of the end user's IP address, usually a 24. We find out that this information can pose an issue for end-user privacy and how network operators can also exploit it to diagnose infection rates or mitigation efforts on their network. We also found out that the world's most popular domain names activate ECS

even though they do not serve geographically distinct IP addresses, making the use of ECS erroneous.

In summary, this thesis described the design and implementation of *Thales*, a system for actively probing DNS data and collecting it for free access by the information security community. Thales's scalability and distributed nature make it an effective tool for forensic investigation, allowing researchers to study the operational changes within the DNS system and their impact on users. We further believe we were able to achieve our primary motivation for this work, which was an effort to generate and share with the community a DNS dataset that addresses the limitations of existing passive DNS datasets and introduces an important element of standardization and consistency, while also being free and easy to obtain for any researcher. We have additionally showcased several examples of the exciting research we have made more accessible through the availability of Active DNS datasets. In short, we hope this work will continue to increase the security community's ability to prevent, detect, and mitigate online abuse.

REFERENCES

[1] F. Weimer, "Passive DNS Replication," in *In Proceedings of the 17th FIRST Conference on Computer Security Incident Handling*, Hand ling, Singapore, Jun. 2005.

[2] Z. Hu, L. Zhu, J. Heidemann, A. Mankin, D. Wessels, and P. E. Hoffman, *Specification for DNS over Transport Layer Security (TLS)*, RFC 7858, May 2016.

[3] S. Dickinson, D. K. Gillmor, and T. Reddy.K, *Usage Profiles for DNS over TLS and DNS over DTLS*, RFC 8310, Mar. 2018.

[4] M. Antonakakis, D. Dagon, X. Luo, R. Perdisci, W. Lee, and J. Bellmor, "A Centralized Monitoring Infrastructure for Improving DNS Security," in *Recent Advances in Intrusion Detection*, Springer, 2010, pp. 18–37.

[5] L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi, "EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis," in *Proceedings of NDSS*, 2011.

[6] M. Antonakakis, R. Perdisci, W. Lee, N. Vasiloglou, and D. Dagon, "Detecting Malware Domains in the Upper DNS Hierarchy," in *Proceedings of the 20th USENIX Conference on Security (USENIX Security)*, 2011.

[7] M. Antonakakis *et al.*, "From throw-away traffic to bots: Detecting the rise of dga-based malware," in *Proceedings of the 21st USENIX Conference on Security Symposium*, ser. Security'12, Bellevue, WA: USENIX Association, 2012, pp. 24–24.

[8] B. Srinivasan *et al.*, "Exposing search and advertisement abuse tactics and infrastructure of technical support scammers," in *Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18*, ACM Press, 2018.

[9] C. Contavalli, W. V. D. Gaast, D. Lawrence, and W. Kumari, *Client Subnet in DNS Queries*, RFC 7871, Internet Engineering Task Force, May 2016.

[10] P. Mockapetris, *Domain names: Concepts and facilities*, RFC 882, Obsoleted by RFCs 1034, 1035, updated by RFC 973, Internet Engineering Task Force, Nov. 1983.

[11] P. Mockapetris, *Domain names: Implementation specification*, RFC 883, Obsoleted by RFCs 1034, 1035, updated by RFC 973, Internet Engineering Task Force, Nov. 1983.

[12] B. Zdrnja, N. Brownlee, and D. Wessels, "Passive monitoring of DNS anomalies," in *Proceedings of DIMVA Conference*, 2007.

[13] M. Antonakakis, R. Perdisci, D. Dagon, W. Lee, and N. Feamster, "Building a dynamic reputation system for dns.," in *USENIX security symposium*, 2010, pp. 273–290.

[14] Y. Chen, M. Antonakakis, R. Perdisci, Y. Nadji, D. Dagon, and W. Lee, "DNS Noise: Measuring the Pervasiveness of Disposable Domains in Modern DNS Traffic," in *Dependable Systems and Networks (DSN), 2014 44th Annual IEEE/IFIP International Conference on*, Jun. 2014, pp. 598–609.

[15] J. Ma, Saul, L. K, S. Savage, and G. M. Voelker, "Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, Jun. 2009.

[16] B. Rahbarinia, R. Perdisci, and M. Antonakakis, "Segugio: Efficient Behavior-Based Tracking of Malware-Control Domains in Large ISP Networks," in *Dependable Systems and Networks (DSN), 2015 45th Annual IEEE/IFIP International Conference on*, Jun. 2015, pp. 403–414.

[17] S. Krishnan and F. Monrose, "An empirical study of the performance, security and privacy implications of domain name prefetching," in *Dependable Systems Networks (DSN), 2011 IEEE/IFIP 41st International Conference on*, Jun. 2011, pp. 61–72.

[18] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, "A search engine backed by internet-wide scanning," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '15, Denver, Colorado, USA: Association for Computing Machinery, 2015, pp. 542–553, ISBN: 9781450338325.

[19] R. van Rijswijk-Deij, M. Jonker, A. Sperotto, and A. Pras, "A high-performance, scalable infrastructure for large-scale active DNS measurements," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 6, pp. 1877–1888, Jun. 2016.

[20] "A light-weighted data collection method for DNS simulation on the cyber range," *KSII Transactions on Internet and Information Systems*, vol. 14, no. 8, Aug. 2020.

[21] N. Miramirkhani, O. Starov, and N. Nikiforakis, "Dial one for scam: A large-scale analysis of technical support scams," *Proceedings 2017 Network and Distributed System Security Symposium*, 2017.

[22] K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song, "Design and evaluation of a real-time url spam filtering service," in *2011 IEEE Symposium on Security and Privacy*, 2011, pp. 447–462.

[23] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker, "Spamscatter: Characterizing internet scam hosting infrastructure," Ph.D. dissertation.

[24] S. Sharma, S. Kapoor, B. R. Srinivasan, and M. S. Narula, "Hicho: Attributes based classification of ubiquitous devices," in *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, Springer, 2011, pp. 113–125.

[25] B. Srinivasan, P. Gupta, M. Antonakakis, and M. Ahamad, "Understanding cross-channel abuse with sms-spam support infrastructure attribution," in *Computer Security – ESORICS 2016*, I. Askoxylakis, S. Ioannidis, S. Katsikas, and C. Meadows, Eds., Cham: Springer International Publishing, 2016, pp. 3–26, ISBN: 978-3-319-45744-4.

[26] B. R. Srinivasan, "Exposing and mitigating cross-channel abuse that exploits the converged communications infrastructure," Ph.D. dissertation, Georgia Institute of Technology, 2017.

[27] S. Krishnan and F. Monrose, "DNS prefetching and its privacy implications: When good things go bad," in *Proceedings of the 3rd USENIX conference on Large-scale exploits and emergent threats: botnets, spyware, worms, and more*, USENIX Association, 2010, pp. 10–10.

[28] F. Zhao, Y. Hori, and K. Sakurai, "Analysis of Privacy Disclosure in DNS Query," in *Multimedia and Ubiquitous Engineering, 2007. MUE'07. International Conference on*, IEEE, 2007, pp. 952–957.

[29] S. Guha and P. Francis, "Identity trail: Covert surveillance using DNS," in *Privacy Enhancing Technologies*, Springer, 2007, pp. 153–166.

[30] S. Bortzmeyer, *DNS Privacy Considerations*, https://tools.ietf.org/id/draft-bortzmeyer-dnsop-dns-privacy-02.txt, Apr. 2014.

[31] S. Bortzmeyer, *DNS Query Name Minimisation to Improve Privacy*, RFC 7816, Mar. 2016.

[32] D. Dagon, N. Provos, C. P. Lee, and W. Lee, "Corrupted DNS Resolution Paths: The Rise of a Malicious Resolution Authority.," in *NDSS*, 2008.

[33] C. Huang, D. A. Maltz, J. Li, and A. Greenberg, "Public DNS system and global traffic management," in *INFOCOM, 2011 Proceedings IEEE*, IEEE, 2011, pp. 2615–2623.

[34] F. Streibelt, J. Böttger, N. Chatzis, G. Smaragdakis, and A. Feldmann, "Exploring EDNS-client-subnet adopters in your free time," in *Proceedings of the 2013 conference on Internet measurement conference*, ACM, 2013, pp. 305–312.

[35] R. Al-Dalky, M. Rabinovich, and K. Schomp, "A look at the ecs behavior of dns resolvers," in *Proceedings of the Internet Measurement Conference*, ser. IMC '19, Amsterdam, Netherlands: Association for Computing Machinery, 2019, pp. 116–129, ISBN: 9781450369480.

[36] J. S. Otto, M. A. Sánchez, J. P. Rula, and F. E. Bustamante, "Content delivery and the natural evolution of DNS: remote DNS trends, performance issues and alternative solutions," in *Proceedings of the 2012 ACM conference on Internet measurement conference*, ACM, 2012, pp. 523–536.

[37] *LinuxContainers.org*, https://linuxcontainers.org/, 2016.

[38] *Actionable analytics.* https://www.alexa.com, 2016.

[39] *Common Crawl*, https://commoncrawl.org/, 2016.

[40] *Abuse.ch domain blacklist*, http://www.abuse.ch/, 2016.

[41] *Malware Domain List*, https://www.malwaredomainlist.com/, 2016.

[42] *Malc0de Database*, http://malc0de.com/bl/BOOT, 2016.

[43] *Sagadc.org list*, http://dns-bh.sagadc.org/, 2016.

[44] *Hphosts feed*, http://hosts-file.net/?s=Download, 2016.

[45] *SANS ISC Feeds*, https://isc.sans.edu/feeds/, 2016.

[46] *I.T. Mate List*, http://vurldissect.co.uk/daily.asp/, 2016.

[47] P. Kintis, "Characterizing network infrastructure using the domain name system," Ph.D. dissertation, Georgia Institute of Technology, 2020.

[48] Y. Nadji, M. Antonakakis, R. Perdisci, and W. Lee, "Connected colors: Unveiling the structure of criminal networks," in *Research in attacks, intrusions, and defenses*, Springer, 2013, pp. 390–410.

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

[50] P. Kintis, Y. Nadji, D. Dagon, and M. Antonakakis, "Understanding the privacy implications of ecs," in *Detection of Intrusions and Malware, and Vulnerability Assessment: 13th International Conference, DIMVA 2016, San Sebastián, Spain, July 7-8, 2016, Proceedings*, Springer, vol. 9721, 2016, p. 343.

[51] O. Alrawi *et al.*, "The circle of life: A large-scale study of the iot malware lifecycle.," in *USENIX Security Symposium*, 2021, pp. 3505–3522.

[52] Kintis, Panagiotis and Miramirkhani, Najmeh and Lever, Charles and Chen, Yizheng and Romero-Gómez, Rosa and Pitropakis, Nikolaos and Nikiforakis, Nick and Antonakakis, Manos, "Hiding in plain sight: A longitudinal study of combosquatting abuse," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 569–586.

[53] K. Tian, S. T. Jan, H. Hu, D. Yao, and G. Wang, "Needle in a haystack: Tracking down elite phishing domains in the wild," in *Proceedings of the Internet Measurement Conference 2018*, 2018, pp. 429–442.

[54] X. Mi *et al.*, "Resident evil: Understanding residential ip proxy as a dark service," in *2019 IEEE symposium on security and privacy (SP)*, IEEE, 2019, pp. 1185–1201.

[55] A. Avgetidis *et al.*, "Beyond the gates: An empirical analysis of http-managed password stealers and operators," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023.

[56] N. P. Hoang, A. Akhavan Niaki, N. Borisov, P. Gill, and M. Polychronakis, "Assessing the privacy benefits of domain name encryption," in *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security*, 2020, pp. 290–304.

[57] I. M. Khalil, B. Guan, M. Nabeel, and T. Yu, "A domain is only as good as its buddies: Detecting stealthy malicious domains via graph inference," in *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy*, 2018, pp. 330–341.

[58] J. Bushart and C. Rossow, "DNS unchained: Amplified application-layer DoS attacks against DNS authoritatives," in *Research in Attacks, Intrusions, and Defenses*, Springer International Publishing, 2018, pp. 139–160.

[59] J. Lee, H. Lee, J. Jeong, D. Kim, and T. T. Kwon, "Analyzing spatial differences in the tls security of delegated web services," in *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, 2021, pp. 475–487.

[60] N. Becker *et al.*, "Streamlined and accelerated cyber analyst workflows with clx and rapids," in *2019 IEEE International Conference on Big Data (Big Data)*, IEEE, 2019, pp. 2011–2015.

[61] N. P. Hoang, A. A. Niaki, M. Polychronakis, and P. Gill, "The web is still small after more than a decade," *SIGCOMM Comput. Commun. Rev.*, vol. 50, no. 2, pp. 24–31, May 2020.

[62] A. Portier, H. Carter, and C. Lever, "Security in plain txt: Observing the use of dns txt records in the wild," in *Detection of Intrusions and Malware, and Vulnerability Assessment: 16th International Conference, DIMVA 2019, Gothenburg, Sweden, June 19–20, 2019, Proceedings 16*, Springer, 2019, pp. 374–395.

[63] R. Romero-Gomez, Y. Nadji, and M. Antonakakis, "Towards designing effective visualizations for dns-based network threat analysis," in *2017 IEEE Symposium on Visualization for Cyber Security (VizSec)*, 2017, pp. 1–8.

[64] Y. Zhou, A. Rathore, E. Purvine, and B. Wang, "Topological simplifications of hypergraphs," *arXiv preprint arXiv:2104.11214*, 2021.

[65] R. Romero-Gomez, Y. Nadji, P. Kintis, and M. Antonakakis, "Visualizing dns datasets for alert-driven threat analysis,"

[66] I. Khalil, B. Guan, M. Nabeel, and T. Yu, *Killing two birds with one stone: Malicious domain detection with high accuracy and coverage*, 2017.

[67] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, "A search engine backed by Internet-wide scanning," in *Proceedings of the 22nd ACM Conference on Computer and Communications Security*, Oct. 2015.

[68] P. Kintis *et al.*, "Hiding in plain sight," *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, Oct. 2017.

[69] P. Papadopoulos, N. Pitropakis, W. J. Buchanan, O. Lo, and S. Katsikas, "Privacy-preserving passive DNS," *Computers*, vol. 9, no. 3, p. 64, Aug. 2020.

[70] Y. Zhauniarovich, I. Khalil, T. Yu, and M. Dacier, "A survey on malicious domains detection through DNS data analysis," *ACM Computing Surveys*, vol. 51, no. 4, pp. 1–36, Jul. 2019.

[71] R. R. Curtin, A. B. Gardner, S. Grzonkowski, A. Kleymenov, and A. Mosquera, "Detecting dga domains with recurrent neural networks and side information," in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, ser. ARES '19, Canterbury, CA, United Kingdom: Association for Computing Machinery, 2019, ISBN: 9781450371643.

[72] C. Lever, R. Walls, Y. Nadji, D. Dagon, P. McDaniel, and M. Antonakakis, "Domain-z: 28 registrations later measuring the exploitation of residual trust in domains," in *2016 IEEE Symposium on Security and Privacy (SP)*, 2016, pp. 691–706.

[73] C. A. Shue, A. J. Kalafut, and M. Gupta, "The web is smaller than it seems," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '07, San Diego, California, USA: Association for Computing Machinery, 2007, pp. 123–128, ISBN: 9781595939081.

[74] C. Lever, R. Walls, Y. Nadji, D. Dagon, P. McDaniel, and M. Antonakakis, "Domain-Z: 28 Registrations Later Measuring the Exploitation of Residual Trust in Domains," in *37th IEEE International Symposium on Security and privacy*, May 2016.

[75] L. Daigle, *WHOIS Protocol Specification*, RFC 3912 (Draft Standard), Internet Engineering Task Force, Sep. 2004.

[76] P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta, "Phishnet: Predictive blacklisting to detect phishing attacks," in *INFOCOM, 2010 Proceedings IEEE*, IEEE, 2010, pp. 1–5.

[77] K. Ishibashi, T. Toyono, H. Hasegawa, and H. Yoshino, "Extending black domain name list by using co-occurrence relation between dns queries," *IEICE transactions on communications*, vol. 95, no. 3, pp. 794–802, 2012.

[78] M. Felegyhazi, C. Kreibich, and V. Paxson, "On the Potential of Proactive Domain Blacklisting," in *Proceedings of the 3rd USENIX Conference on Large-scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More (LEET)*, Apr. 2010.

[79] *Domain Graveyard*, http://domaingraveyard.com/, 2016.

[80] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, *Address Allocation for Private Internets*, RFC 1918 (Best Current Practice), Updated by RFC 6761, Internet Engineering Task Force, Feb. 1996.

[81] M. Cotton and L. Vegoda, *Special Use IPv4 Addresses*, RFC 5735 (Best Current Practice), Obsoleted by RFC 6890, updated by RFC 6598, Internet Engineering Task Force, Jan. 2010.

[82] J. Weil, V. Kuarsingh, C. Donley, C. Liljenstolpe, and M. Azinger, *IANA-Reserved IPv4 Prefix for Shared Address Space*, RFC 6598 (Best Current Practice), Internet Engineering Task Force, Apr. 2012.

[83] B. Coat, *Snake In The Grass: Python-based Malware Used For Targeted Attacks*, https://www2.bluecoat.com/security-blog/2014-06-10/snake-grass-python-based-malware-used-targeted-attacks, 2014.

[84] M. L.
bibinitperiod C. C. Security, *CopyKittens Attack Group*, https://eforensicsmag.com/copykittens/, 2015.

[85] Mandiant, "APT1," Tech. Rep., 2013, http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf.

[86] C. Contavalli, W. V. D. Gaast, D. Lawrence, and W. Kumari, *Client Subnet in DNS Requests (draft-ietf-dnsop-edns-client-subnet-00)*, 2015.

[87] *A free, global DNS resolution service that you can use as an alternative to your current DNS provider.* https://developers.google.com/speed/public-dns, 2020.

[88] *IMPROVE YOUR INTERNET*, https://www.opendns.com/, 2020.

[89] *An open DNS recursive service for free security and high privacy*, https://www.quad9.net, 2021.

[90] *The new firewall for the modern Internet*, https://nextdns.io, 2021.

[91] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, p. 3, 2007.

[92] B. Greschbach, T. Pulls, L. M. Roberts, P. Winter, and N. Feamster, "The effect of dns on tor's anonymity," *arXiv preprint arXiv:1609.08187*, 2016.

[93] M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas, and L. Jones, *SOCKS Protocol Version 5*, RFC 1928 (Proposed Standard), Internet Engineering Task Force, Mar. 1996.

[94] OpenDNS, *The OpenDNS Global Network Delivers a Secure Connection Every Time. Everywhere.* http://info.opendns.com/rs/opendns/images/TD-Umbrella-Delivery-Platform.pdf, 2010.

[95] Google, *Introduction to Google Public DNS*, https://developers.google.com/speed/public-dns/docs/intro, Accessed: 2021-08-07.

[96] P. Agten, W. Joosen, F. Piessens, and N. Nikiforakis, "Seven months' worth of mistakes: A longitudinal study of typosquatting abuse.," in *NDSS*, 2015, pp. 156–168.

[97] J. Szurdi, B. Kocso, G. Cseh, J. Spring, M. Felegyhazi, and C. Kanich, "The long "taile" of typosquatting domain names," in *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, 2014, pp. 191–206.

[98] R. Views, *University of oregon route views project*, 2000.

[99] K. Borgolte *et al.*, "How DNS over HTTPS is reshaping privacy, performance, and policy in the internet ecosystem," *SSRN Electronic Journal*, 2019.

[100] Hoffman, P and McManus, P, *DNS Queries over HTTPS (DoH)*, RFC 8484, Internet Engineering Task Force, 2018.

[101] MAXMIND, *GeoIP2: Industry Leading IP Intelligence*, 2015.

[102] R. Housley, J. Curran, G. Huston, and D. R. Conrad, *The Internet Numbers Registry System*, RFC 7020, Aug. 2013.

[103] *IP to ASN Mapping - Team Cymru*, http://www.team-cymru.org/IP-ASN-mapping.html, 2016.

[104] C. Contavalli, W. V. D. Gaast, D. Lawrence, and W. Kumari, *Client Subnet in DNS Requests*, RFC 7871, Internet Engineering Task Force, May 2015.

[105] OpenDNS, *A Faster Internet: http://www.afasterinternet.com*, 2011.

[106] Feamster, Nick and Winick, Jared and Rexford, Jennifer, "A model of BGP routing for network engineering," in *ACM SIGMETRICS Performance Evaluation Review*, ACM, vol. 32, 2004, pp. 331–342.

[107] Feamster, Nick and Rexford, Jennifer, "Network-wide prediction of BGP routes," *IEEE/ACM Transactions on Networking (TON)*, vol. 15, no. 2, pp. 253–266, 2007.

[108] Madhyastha, Harsha V and Anderson, Thomas and Krishnamurthy, Arvind and Spring, Neil and Venkataramani, Arun, "A structural approach to latency prediction," in *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, ACM, 2006, pp. 99–104.

[109] Mühlbauer, Wolfgang and Feldmann, Anja and Maennel, Olaf and Roughan, Matthew and Uhlig, Steve, "Building an AS-topology model that captures route diversity," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 4, pp. 195–206, 2006.

[110] P. Mockapetris, *Domain names - concepts and facilities*, RFC 1034 (INTERNET STANDARD), Updated by RFCs 1101, 1183, 1348, 1876, 1982, 2065, 2181, 2308,

2535, 4033, 4034, 4035, 4343, 4035, 4592, 5936, Internet Engineering Task Force, Nov. 1987.

[111] P. Mockapetris, *Domain names - implementation and specification*, RFC 1035 (INTERNET STANDARD), Internet Engineering Task Force, Nov. 1987.

[112] Shadowserver, *Shadowserver Foundation*, https://www.shadowserver.org/, 2016.

[113] O. Poitrey, *How we made DNS both fast and private with ECS*, https://medium.com/nextdns/how-we-made-dns-both-fast-and-private-with-ecs-4970d70401e5, 2019.

[114] Z. Durumeric, E. Wustrow, and J. A. Halderman, "Zmap: Fast internet-wide scanning and its security applications," in *Proceedings of the 22nd USENIX Conference on Security*, ser. SEC'13, Washington, D.C.: USENIX Association, 2013, pp. 605–620, ISBN: 9781931971034.

[115] G. Kambourakis, T. Moschos, D. Geneiatakis, and S. Gritzalis, "Detecting DNS amplification attacks," in *Critical Information Infrastructures Security*, Springer Berlin Heidelberg, 2008, pp. 185–196.

[116] K. Alieyan, M. M. Kadhum, M. Anbar, S. U. Rehman, and N. K. A. Alajmi, "An overview of DDoS attacks based on DNS," in *2016 International Conference on Information and Communication Technology Convergence (ICTC)*, IEEE, Oct. 2016.

[117] M. Antonakakis *et al.*, "Understanding the mirai botnet," in *26th USENIX Security Symposium (USENIX Security 17)*, Vancouver, BC: USENIX Association, Aug. 2017, pp. 1093–1110, ISBN: 978-1-931971-40-9.