University of Twente

Master's Thesis

On Services Exposed by DNS Infrastructure: A KINDNS Investigation

Georgia Christou

June, 2023

Committee
dr. A. Sperotto (Anna)
dr.ing. F.W. Hahn (Florian)
R. Sommese MSc (Raffaele)

Faculty
Electrical Engineering, Mathematics and Computer Science (EEMCS)

Programme
Computer Science: Cyber Security Specialization

Group
Design and Analysis of Communication Systems (DACS)

Abstract

Despite the increased community efforts to improve DNS hygiene, DNS operators seldom live up to industry standards. ICANN acknowledged this with the introduction of the KINDNS framework, intended to offer better focus and incentives to DNS operators via a purposefully compact ruleset. This work stands as an initial attempt to investigate KINDNS readiness with regard to the services offered on DNS infrastructure. The findings reveal only a few DNS hosting providers being ready for KINDNS adoption. When configuration lies in the hands of individuals, the practices of virtual private server providers show security at its weakest. DNS insecurity is further supported by 2.5% of authoritative servers, most of which appearing in the wild for over 2 years, that increase their attack surface by offering recursion. Recursive servers are more guilty of weak configurations, with 99% of them neglecting DNS-over-Encryption in their communication with clients. 70% of authoritative and 24% of recursive servers are further guilty of acting beyond their DNS functionalities, though the practices of more popular and shared zones are better. It hereby remains to be seen if KINDNS does eventually align everyone's priorities so as to have security at their center.

Acknowledgements

This thesis marks the end of my academic years with a project I genuinely enjoyed working on. This is the first thing I am grateful for, to be given a chance to be part of the DACS research group as well as the freedom to choose to work on what I enjoy. For that, I would like to thank Anna, firstly for agreeing to be my supervisor, but also for her valuable insight and guidance on every occasion needed. Of course, I also owe a special thank you to Raffaele, for being my weekly compass during the last months, providing me with paramount feedback and condoning my ever-changing schedule. A last but certainly not least thank you belongs to my family and my friends who travelled this journey alongside me, acting as a constant source of support, love and inspiration. At times, I certainly wouldn't be able to look at the bright side of things if not without you all. So thank you.

Contents

List of Figures

List of Tables

1 Introduction

Contents

1.1 Research Motivation

As the Internet integrates more and more persistently into everyday life, it becomes inherently difficult to imagine a world in which Internet navigation is not as convenient as we've become accustomed to. The Domain Name System (DNS) played a vital part in providing this convenience, serving as a distributed phonebook of mappings between the numerical IP identifiers of Internet entities to their human-friendly domain names.

Though DNS was not designed with the intention to support the billion number of entities it supports today [40], its distributed nature allowed it to scale nonetheless. At the same time, however, this same distributed nature became a weakness. This is not only due to the innate complexities that come with distributed systems [80] but also due to the highly dynamic nature of the cyberspace [35].

Back when the original DNS specifications were proposed, the cyberspace was significantly different than that of today, a trusted ecosystem in which the possibility of attacks was negligible. This allowed DNS to operate based on assumptions that no longer hold [83]. The quickly unsafe nature of the Internet became most alarming for DNS in 2008, when Dan Kaminsky revealed a major flaw in DNS design, that is, the possibility to perform arbitrary cache poisoning to direct Internet users to malicious websites [43]. As the cyberspace grew more unsafe, the Internet standards organization (IETF) proposed several practices on protecting DNS, yet the proposed practices were never mandatory to follow. Up to today, the community's attempts to secure DNS come in the form of guidelines that are up to the DNS operators to decide to follow or ignore. Unfortunately, the latter is very often the case.

Despite the increased community efforts to improve DNS hygiene, as indicated by hundreds of DNS-focused RFCs [66], this very abundance of information proved to be particularly challenging for the average DNS operator to fully implement or even understand. In fact, years of DNS research reveal that DNS operators seldom live up to industry best standards, sometimes because the proposed guidelines come with poor implementation trade-offs, other times because good intentions are not complemented with adequate expertise. The abundance of documentation at the absence of a unified framework certainly doesn't make things any better. The Internet Corporation for

Assigned Names and Numbers (ICANN) acknowledged this problem with the introduction of the Knowledge-Sharing and Instantiating Norms for DNS and Naming Security (KINDNS) [44], the first formal framework of DNS security best practices.

Just like the lot of documentation that came before it, KINDNS too is not an obligation. The new framework is not intended to lecture DNS operators nor to overwhelm them, but rather it is intended as a movement that DNS operators may participate in by choosing to follow the proposed security best practices. The set of practices is purposefully compact, this way drawing attention only to the most important ones so that smaller and bigger operators alike are properly focused and incentivized to operate DNS securely.

1.2 Research Contributions

At the point of performing this research, KINDNS is still a relatively new attempt, launched in the summer of 2022. Research focus on ICANN's new initiative is thus naturally limited. This thesis serves as one of the first attempts to measure DNS security in accordance with a unified framework, that is, KINDNS. But though the proposed set of best practices is compact compared to the totality of DNS-focused RFCs, the entirety of KINDNS guidelines is still too large to cover as part of a Master's dissertation. The focus of this work hereby lies in a set of three types of KINDNS practices that, together, describe the kinds of services that DNS servers should and should not offer, a side of DNS hygiene that has not been researched at scale.

1.3 Research Objectives

The first kind of service is to serve DNS itself but in a way that is strictly relevant to a server's role in the DNS ecosystem. Namely, the requirement is that a server commits to its intended role in the DNS ecosystem by using the IANA-assigned DNS port 53 to serve as either authoritative or recursive infrastructure, the former meant to serve information over a specific part of the global namespace, the latter meant to retrieve such information upon client request. In adhering to this requirement, KINDNS seeks to minimize a server's attack surface, in this way minimizing the attack surface of the DNS ecosystem as a whole. In violating this requirement, servers create a single point of failure within DNS, offering the possibility of abuse either at the client/recursive or the server/authoritative side of DNS. Throughout this work, servers that violate this requirement are referred to as offering *duplex* DNS as a service.

The second kind of service is with relation to recursive servers, and that is to serve encrypted DNS on either port 853 or 443, reserved to run over the the TLS protocol and the HTTPS protocol respectively. In adhering to this requirement, KINDNS seeks to ensure a level of privacy in DNS exchanges, protecting the confidential information communicated from and to Internet users. In violating this requirement, servers handle their communication with clients in the clear, leaving their content uncovered and susceptible to any unsolicited third-party that monitors the network. Throughout this work, servers that violate this requirement are referred to as offering *unencrypted* DNS

as a service.

Finally, the third requirement is in respect to non-DNS services, proposing that DNS servers keep to their intended role in the overall Internet ecosystem, by offering DNS and DNS only. Hereby, serving non-DNS services or allowing access to non-DNS ports should both be denied by default. In adhering to this requirement, here too KINDNS seeks to minimize a server's attack surface and consequently the attack surface of DNS in general. In violating this requirement, servers are susceptible to whatever vulnerability is relevant to their DNS-irrelevant protocol, which in turn makes this type of violation of arbitrary harm potential. Throughout this work, servers that violate this requirement are referred to as offering *beyond* DNS as a service.

This study is founded on the aforementioned three requirements and on how these relate to important and less important servers and networks. Specifically, the research questions guiding the rest of this work are more formally stated as follows:

1. How many DNS operators comply to KINDNS by separating authoritative and recursive services?
   (a) How many authoritative servers offer recursion on the DNS standardized port 53?
2. How many DNS operators comply to KINDNS by serving encrypted DNS?
   (a) How many recursive servers offer encrypted DNS over the TLS protocol on the standardized port 853?
   (b) How many recursive servers offer encrypted DNS over the HTTPS protocol on the standardized port 443?
3. How many DNS operators comply to KINDNS by restricting access to all non-DNS ports and services?
   (a) How many authoritative servers expose non-DNS open ports?
   (b) How many recursive servers expose non-DNS open ports?
   (c) What services are exposed on the identified open ports?
4. Are there any drivers behind certain operational practices?
   (a) Are more important servers following similar operational patterns with less important ones?
   (b) Are specific networks responsible for driving specific operational patterns?
   (c) Are servers that violate or comply to one practice more likely to repeat this pattern to another practice?

1.4 Research Approach

By combining existing measurements with new measurements, this research managed to minimize the time and ethical constrains of performing all measurements from the ground up, while at the same time taking advantage of the computational resources and

DNS intelligence of previous research. For the most part, existing measurements were used to create target lists of authoritative and recursive IPs, whereas new measurements were used to estimate the KINDNS readiness of the target servers.

For discovering duplex DNS violations (Research Question 1), focus was first directed on identifying servers known to appear both in the target list of authoritative servers but also in the target list of recursive servers. As the intersection, however, only held a specific viewpoint of DNS infrastructure at a certain point in time, a second discovery was performed, by actively querying the target list of authoritative servers on port 53 to retrieve DNS information outside their authorization.

For discovering unencrypted DNS violations (Research Question 2), the target list of recursive servers was actively queried on ports 853 and 443 to respond to a certain DNS request. Servers that failed to provide an answer to the query within the specified time frame were then classified as offering unencrypted DNS, though the classification could as well be a false negative if servers were too slow to respond or if they served encrypted DNS on non-standardized ports.

For discovering beyond DNS violations (Research Question 3), the methodology was the same for authoritative and recursive servers. Namely, the servers were first actively prompted on all their non-DNS ports up to 1,023 included. Then, those that responded to the probing were further sent an application-specific packet to check if they served their IANA-assigned protocol. Because of the time and ethical complications of the second phase, only a few popular services were examined.

For discovering any drivers behind operational patterns (Research Question 4), this research examined a number of different factors, focusing on both individual servers but also their networks. As a way of estimating the importance of authoritative servers, two importance metrics were used, one based on domain popularity rankings and one based on shared responsibility of popular domains. For recursive servers also, importance was estimated on the basis of two importance metrics, one based on the persistence of a server through the course of time and one based on public knowledge of known recursive servers. Furthermore, DNS market dominance was used as a metric of importance for networks rather than individual servers.

1.5 Document Structure

The remaining of this document sets forth with some important context regarding DNS and KINDNS. This information is formulated in the **Background** chapter. Previous relevant research and measurements are presented next, as part of the **Related Work** chapter. The next two chapters constitute the measurements themselves, laying down the data, tooling and techniques used to perform the research, as well as its emerging outcomes. These are presented in the **Methodology** and **Results** chapters respectively. Ultimately, the thesis rounds off with a **Conclusions** chapter, summarizing the research and proposing how future work can benefit from this work's findings and limitations.

2 Background

Contents

2.1 DNS

2.1.1 Basic Functionalities

In its simplest form, as first formulated in RFC 1034 [56] and RFC 1035 [57], DNS operates with three main entities. On the client side of DNS, stub resolvers act as software interfaces that connect Internet users to the DNS ecosystem by forwarding DNS queries. Such queries can be as simple as requesting the IP of a domain, say www.utwente.nl, though it is important to note that DNS queries can request information beyond just mappings.

Assuming a typical stub resolver which receives a DNS query to translate a certain domain to its IP and assuming that the stub resolver does not implement internal caching, the stub resolver does not perform the translation itself. Instead, the stub resolver forwards the request to another DNS entity, that is, a recursive resolver, also referred to as a recursive nameserver. Depending on its network location and its accessibility, a recursive resolver can be either private, semi-public, or public, in the first case residing within a local network and accessible only from inside the network, in the second case residing out in the open yet reachable only by a specific set of users, and in the third case being accessible by any Internet user. The latter case of resolvers, also referred to as open resolvers, are naturally easier to measure as a result of their accessibility, but this also makes them an easier attack target. With that said, the behaviour of private and semi-public resolvers is exempted from this research, as they are both harder to measure and abuse.

With the exception of their network location, different types of recursive resolvers are not fundamentally different. Upon receiving a DNS query, a recursive resolver may serve the query response directly from its cache, if the same information is recently queried by some client and stored in the cache. If the desired information is not in the

cache, then the recursive resolver has to perform a process called DNS resolution, that is, to recursively request the desired information from the DNS servers responsible to store this information. These servers are called authoritative nameservers, and each of them is responsible for a specific part of the global namespace. By following a sequence of requests to the relevant authoritative nameservers, a recursive resolver ultimately obtains the desired information and returns it to the stub client.

But to understand how DNS resolution actually works and how exactly the responsibility of authoritative nameservers is reflected in the DNS ecosystem, it is important to take a step back and look at the data that compose DNS. From the perspective of the Internet user, the domain name is the core data unit. A domain name is basically a specially crafted string of ASCII characters structured as a sequence of labels separated by dots. For the case of www.utwente.nl, the domain consists of three labels, as indicated by the two dots. DNS resolution relies upon the domain structure, as different authoritative nameservers are responsible for answering requests for different domain parts. **Figure 2.1** shows a small subset of the tree-like structure of DNS, with every node in the tree representing a different part of the domain namespace, also known as zone. DNS zones essentially depict the DNS hierarchy, with each zone storing unique information that is accessible by one or more authoritative nameservers. As these servers are intended to communicate with arbitrary resolvers, they can only do that if they are public-facing, which in turn makes them easier to measure at scale.

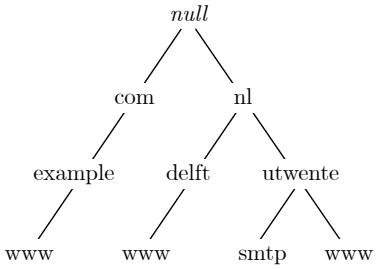

Figure 2.1: Example Delegation Subtree of DNS Hierarchy

As their name suggests, authoritative nameservers are authorized to serve DNS information. This authorization is delegated to a server via its parent zone in the tree. In the example of **Figure 2.1**, the root zone delegates responsibility over the .com and .nl zones to the authoritative nameservers in .com and .nl respectively. As .com and .nl lie right below the root, these zones are referred to as Top Level Domains (TLDs), which are further classified into generic (gTLDs) and country code (ccTLDs). The zones below the root further delegate responsibility over subparts of the global namespace to their own children in the tree, referred to as Second Level Domains (SLDs). In turn, these nodes too delegate responsibility over smaller parts of the global namespace to nodes lower in the tree. This responsibility is reflected in a server's zone file via various types of resource records, each serving as the core data unit from the perspective of DNS servers. The type of information in a record is reflected via a resource type, two of the most common of which being the A record and the NS record, pointing to a domain's IPv4 address and a zone's authoritative nameserver respectively. Though these are hardly the only resource types, the two of them alone suffice to finally break

down the underlying idea behind DNS resolution.

A recursive resolver performs DNS resolution by requesting information from the relevant nameservers in the tree hierarchy, starting from the root. For the example hierarchy in **Figure 2.1**, resolving the IP of www.utwente.nl would require the resolver to first ask a root nameserver if it has the A record of www.utwente.nl. Because the root nameserver is not authorized to serve this information nor has access to it, it would direct the resolver to a nameserver of the .nl zone, as indicated by a corresponding NS record. As the .nl nameserver too is not responsible for the domain, it would then direct the resolver to a nameserver of utwente.nl. Such server is also not authorized to serve or store this information, so it would direct the resolver to a nameserver of www.utwente.nl. After reaching this level of the hierarchy, the resolver would obtain the IP of www.utwente.nl and return it to the DNS client. The complete flow of requests performed to ultimately retrieve this information is summarized in **Figure 2.2**.

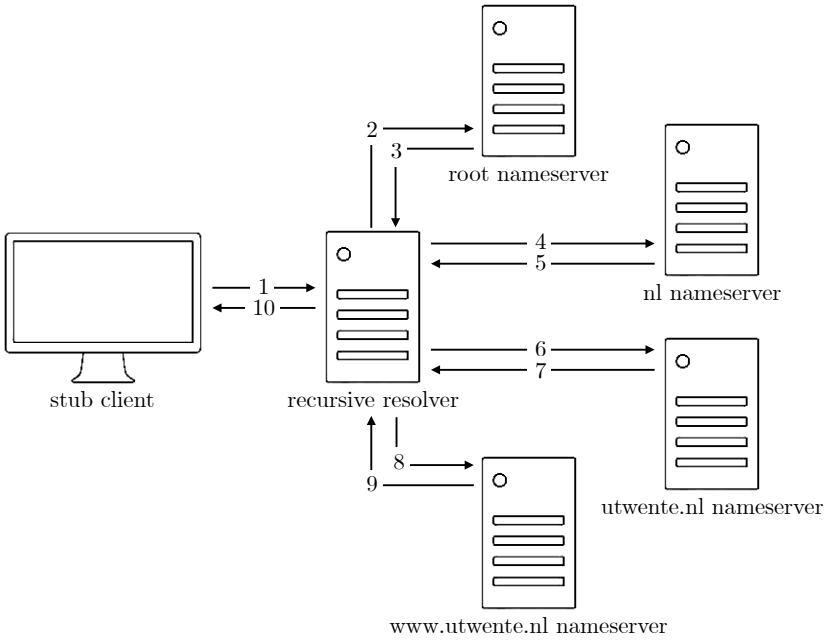

Figure 2.2: Example Information Flow of DNS Resolution

2.1.2 Security Advancements

But the original DNS architecture ultimately proved itself incapable to account for the complexities of the ever-growing cyberspace of the future. Indeed, the original DNS design was in many ways always vulnerable, but its problems were never seriously abused in its earliest years. In those years, DNS hygiene was more so a given than a requirement, a natural guarantee of a less crowded and more secure cyberspace. But as Internet popularity increased, so did the exposure to risks.

DNS has been the target of different types of attacks throughout the years, with some attacks targeting the confidentiality, integrity, and availability of DNS itself, and

others abusing the popularity and complexity of DNS to launch attacks on others [83]. To ensure DNS hygiene despite the unsafe ecosystem, the original DNS specifications needed to be carefully reconsidered and readjusted. One of the most important DNS advancements was the introduction of the Domain Name System Security Extensions (DNSSEC) in 2005, as formulated in RFC 4033 [4], RFC 4034 [6], and RFC 4035 [5].

Because the original DNS design had no mechanism to detect whether DNS data originated from the expected source or whether data were tampered before or during the exchange, attackers were able to manipulate DNS data to direct users to malicious websites [79]. DNSSEC was introduced to provide origin authenticity and data integrity to DNS with the use of public key cryptography and digital signatures as a way of certifying the delegation of responsibilities in the DNS hierarchy. With DNSSEC deployed at the authoritative side, DNS responses could now be complemented with their unique digital signatures as a means to prove their legitimacy. With DNSSEC deployed at the recursive side, data forging could in turn be prevented by checking the validity of signed responses.

But DNSSEC was no remedy to all problems, but a mechanism specifically proposed for a specific kind of threat. In the years following, new security advancements were proposed to account for new security needs. DNS-over-Encryption applications [33, 32, 34], for instance, were proposed to add privacy to DNS in the communication between stub and recursive resolvers. A more recent privacy mechanism is QNAME minimization [8], introduced for protecting the communication with authoritative nameservers by minimizing the information leakage in DNS resolution. Other privacy guidelines aim to protect the authoritative side of DNS, with AXFR restrictions [50, 13], for instance, proposed to prevent against arbitrary zone transfers that could allow one to abuse the value or magnitude of information in a zone.

Despite the increased community efforts to improve DNS hygiene, years of DNS research reveal that IETF guidelines are hardly ever timely or properly deployed. Sometimes, poor security decisions trace back to poor implementation trade-offs, as is the case, for instance, with the increased latencies introduced by QNAME minimization [86]. Other times, good security intentions are not complemented with an adequate level of understanding or expertise to implement security properly. DNSSEC, for instance, though one of the oldest security advancements, is still not adequately deployed even today, while for those that do implement it, they do not always implement it properly, with misconfigured resource records hindering a zone's availability [22, 1] and vulnerable cryptographic choices hindering its security [84, 77, 15].

2.2 KINDNS

2.2.1 Framework Overview

KINDNS promises to address the challenges that come with information abundance by proposing a purposefully compact set of DNS security best practices, as these are considered to be the most important ones. The ultimate objectives of KINDNS shine through its very name, intended to be pronounced "kindness", as this is what it strives to promote in the DNS ecosystem. The new initiative is inspired by another recent initia-

tive, that is, the Mutually Agreed Norms for Routing Security (MANRS) [54], intended to be pronounced "manners". KINDNS seeks to complement MANRS, by promoting DNS security so as to bridge the knowledge and feasibility gaps that are usually complemented with poor security decisions. Like MANRS, KINDNS is a non-compulsory framework, intended as a movement rather than an obligation. By proving compliance to its guidelines, an organization may publicly join KINDNS, hereby increasing not just its security posture but also its public reliability. Proving this compliance relies on means of self-assessment, which entails a relative flexibility for DNS operators but, at the same time, it creates challenges with third-party measurability [78].

KINDNS is not designed with third-party measurability in mind. For one, KINDNS is intended towards public *and* private infrastructure, the latter being inherently more challenging to measure without direct access inside a network. But even for those guidelines intended towards public infrastructure, some practices are just impossible to measure from the outside. KINDNS requirements on logging, monitoring, versioning, and user permissions are just a few examples where internal network access is essential in their measuring. This research acknowledges these limitations by choosing to focus on a set of practices that are at least somewhat measurable with mere external network access. This research further acknowledges the time constraints and general complexities that come with the measuring, hence limits itself to a subset of the proposed practices. That is because, though the totality of KINDNS guidelines is compact compared to the totality of DNS-focused RFCs, a complete overview of every KINDNS practice would still be daring. Rather, this thesis focuses on merely three types of KINDNS practices that, together, formulate the kinds of services that DNS servers should and should not offer.

2.2.2 Guidelines on Duplex DNS

A widely used principle in the protection of computer systems in general is the so-called Separation of Duties (SoD) [16]. The idea refers to any set of controls taken to ensure that no single entity, be it human or machine, is responsible to complete a certain task by itself. By requiring the involvement of multiple entities instead decreases the risk of a system's abuse, consequently increasing its resilience.

For DNS, the SoD principle entails a separation of the two kinds of DNS services one can offer on DNS port 53, that is, authoritative DNS and recursive DNS. This separation has several benefits. From a performance perspective, offering both services on the same server increases the traffic load targeted towards the server hereby degrading its responsiveness. From a security perspective, combining the two services not only makes for a single point of failure but actually increases the server's attack vectors. Distributed Denial of Service (DDoS) abuse provides a good example of the latter case: Because authoritative nameservers can be configured to reduce their amplification factor, they are generally considered to bear a reduced likelihood of DDoS abuse [20]. This is not the case for recursive resolvers, whose client-side nature allows to query any authoritative nameserver of any amplification factor [17]. This makes them great candidates for DDoS abuse, and even more so when they are open to the wider Internet. Combining an authoritative and recursive service consequently overshadows the otherwise harmless nature of the authoritative service.

Although some DNS vendors, as is the case with PowerDNS [67], already satisfy the SoD principle by providing distinct products for running authoritative and recursive services, most vendors allow running either service through the same software package. KINDNS is concerned with the latter case of software, requiring that DNS operators steer away from offering both types of services, be it intentionally or accidentally. This constitutes the first of the three KINDNS practices of interest to this study.

2.2.3 Guidelines on Unencrypted DNS

Cryptography is a common application in achieving security, typically used to satisfy integrity, authentication and confidentiality requirements. DNSSEC uses cryptography as a means of integrity and authentication, but not to conceal data content, something that entails several concerns regarding the privacy of Internet users [87]. DNS-over-Encryption is a relatively recent DNS advancement that uses cryptography to encrypt the content of DNS messages so as to ensure the confidentiality in the communication between stub and recursive resolvers. DNS-over-Encryption further provides integrity and authentication via the use of X.509 certificates [18], yet not in the is same way that DNSSEC provides these security guarantees, though in many ways similar. In particular, similarly with how DNSSEC certifies the delegations in DNS hierarchy, X.509 certificates offer a way to verify that cryptographic keys and their encrypted messages are not in any way tampered during the exchange. The core limitation in the use of X.509 in DNS is that it can only guarantee that messages come from a specific host, but not if the sender is indeed delegated authorization to originate these messages. The latter can only be guaranteed via the use of DNSSEC, as it reflects the true relationships between DNS zones.

Practically, there are two main implementations of DNS-over-Encryption. DNS-over-TLS (DoT) [33] is the first, by applying TLS on top of DNS to protect the otherwise insecure network used for the transmit. A common way to configure a DoT server is to receive DoT queries on the DoT dedicated port 853. DNS-over-HTTPS (DoH) [32] is a second option, by delivering DNS messages as if HTTPS traffic, that is, by encoding them into DNS-purposed media types. As DoH traffic is essentially no different than normal HTTPS traffic, implementing DoH typically relies on receiving DNS queries on the HTTPS dedicated port 443 and by using a certain URI template. The standardized template to use is /dns-query, though, just like with ports, this is the recommended configuration but not an obligation.

KINDNS includes both DNSSEC and DNS-over-Encryption as requirements of compliance. For authoritative nameservers, the requirement is that they perform DNSSEC signing while adhering to key management best practices. For open resolvers, the requirement is that they not only perform DNSSEC validation before accepting DNS responses from the authoritative side, but they also deploy either DoT or DoH in their communication with stub clients. As DNSSEC does not constitute a service as such but a mechanism, it falls outside this research's objectives. DNS-over-Encryption, on the other hand, is the second of the three KINDNS practices of interest to this study.

2.2.4 Guidelines on Beyond DNS

To minimize DNS attack surface in general, KINDNS further proposes that DNS servers commit to their intended role in the Internet ecosystem by offering DNS and nothing beyond that. The harm that this requirement is striving to address is less explicit than the previous requirements, but rather it is interlocked with whatever harm is particularly relevant to the various kinds of non-DNS services. Offering HTTP on a server, for instance, comes with different vulnerabilities than offering SSH on the same server; the former can be abused by a malicious input injection, while the latter can be abused by a brute force password attack. Though the violation options are different, of course, both cases can achieve the same objective, that is, to gain unauthorized access to a DNS server.

KINDNS proposes a set of practices on non-DNS behaviour. The first is for DNS operators to restrict their network traffic using firewall policies that deny all non-DNS incoming and outgoing traffic by default. KINDNS acknowledges, however, that merely restricting network access with firewalls is not enough, and that a minimized attack surface should further rely on actually restricting the functionalities of a server. This entails that a server has nothing but DNS software installed and that is serves nothing but DNS services. The only exception to this concerns administrator tools and portals such as SSH or other remote administration mechanisms, as these are essential to DNS management. Even with that exception provided, however, proper firewall configuration entails that accessing such management ports and services should still be infeasible from outside the network.

The relevant KINDNS requirements on non-DNS behaviour concern both authoritative and recursive infrastructure. For authoritative infrastructure, KINDNS only permits serving DNS on port 53 and nothing else. For recursive infrastructure, KINDNS further permits serving DoT and/or DoH on their relevant ports. This last set of requirements completes the set of objectives of interest to this study.

3 Related Work

Contents

3.1 Research on DNS Hygiene

DNS has been a highly researched area in academia. This work aims to complement existing research by focusing on a subset of best practices proposed by KINDNS. In doing that, this thesis stands as an initial attempt to measure DNS hygiene according to the requirements of a unified framework, that is, KINDNS. Furthermore, this thesis also serves as an initial attempt to provide some insights into KINDNS readiness in general, revealing to what degree DNS operators would or would not be able to participate in KINDNS with their current practices.

At the absence of a unified framework before KINDNS, research focus on DNS hygiene has been dispersed, with some areas receiving different attention than others. The case of DNSSEC for example, a practice proposed by KINDNS, though has received significant attention on the authoritative side [63, 70, 22, 1, 84, 77, 15], measurements on the recursive side have been more targeted on private resolvers [31, 29, 90, 46, 51, 48, 85, 15] and less so on open ones [24, 15, 53]. This study aims to bridge similar research gaps by examining practices that received less or no attention throughout the years.

3.2 Research on Duplex DNS

Though the SoD principle in itself received its fair share of attention in DNS literature, the so far attention focused on a different application of the principle as opposed to what KINDNS proposes. Namely, a number of studies showed interest in measuring infrastructure sharing between different DNS zones [75, 76, 2, 91], yet there appears to be little research interest in measuring infrastructure sharing between different DNS services.

As part of identifying open resolvers in IPv4, Kuhrer et al. [45] explicitly addressed the possibility that hosts can serve both authoritative and recursive roles, but the frequency of the practice was not measured. A similar comment was made by Nijenhuis [59] as part of identifying open resolvers in IPv6, but further characterization of the phenomenon was again omitted.

The Measurement Factory [27] seems to provide the only measurement of recursion support on authoritative space to date. However, the study was performed more than 10 years ago and only for a small subset of IPs. This study seeks to present a more recent and broader picture instead.

3.3 Research on Unencrypted DNS

Though both DoT and DoH are fairly new DNS security advancements, they have both received their relative share of attention. For one, various public lists [68, 88, 7, 82] provide information on resolvers known to support DoT and/or DoH. These lists, however, are far from complete, built based on information announced by large organizations, hereby reflecting only a portion of the real deployment.

In 2019, Lu et al. [52] were the first to identify DoT and DoH resolvers outside the public lists, followed by Deccio and Davis [21] a few months later. The two studies used a fairly similar approach in discovering DoT/DoH support, by actively querying the candidate resolvers on their standardized ports and with popular URI templates in DoH connections. The two studies also addressed similar limitations in their measurements, among which the possibility that some resolvers may deploy the protocols on non-standardized ports or even using non-obvious URI templates as DoH endpoints.

Garcia et al. [30] identified a further limitation in their analysis in 2021, that is, the necessity of the Server Name Indication (SNI) extension when performing DoT/DoH requests. This limitation is in respect to servers that host more than a single service over TLS on the same port. When the SNI value is not specified when connecting to such servers, the servers cannot declare the desired target host and so neither the relevant certificate, which leads to a connection failure. Luo et al. [53] took this into account to present a more representative deployment picture in 2022, trying several candidate SNI values obtained from zone data, certificate data and third-party data.

This study seeks to complement previous measurements by investigating how DNS-over-Encryption deployment translates to KINDNS readiness in particular. The chosen DoT/DoH identification method is not particularly novel as such, as it rather resembles earlier research attempts, focusing on standardized implementations while discarding SNI in the connections altogether. These choices yield an under-representative picture of the real deployment, missing to identify non-standardized implementations and SNI failures. For the case of non-standardized implementations such as non-predictable ports or endpoints, these servers are inherently challenging to identify because of the unpredictability of their implementations. A choice to focus on standardized implementations thus offered a trade-off between experimental efficiency and accuracy. For the case of SNI failures, however, these actually reflect KINDNS violations. That is because, when a connection fails as a result of a missing SNI, it is essentially because the server runs more than a single service on the port. Failing to discover DoT/DoH compliant servers *because* of a missing SNI is thus not entirely a false classification, as these servers are in fact KINDNS non-compliant in another way.

3.4 Research on Beyond DNS

Open port discovery forms an integral part of penetration testing [23], which certainly makes it a widely used and valuable tool for the industry. In academia too, open port discovery is equally important, with some research, for instance, targeted towards discovering open ports on common user computers [55], mobile applications [39], but also cloud infrastructure [58]. Surprisingly, however, no similar focus was ever directed towards DNS infrastructure.

Indeed, several DNS-focused measurements utilized network scanning to identify DNS software [27, 81, 45] and DNS services [52, 21, 30, 53], but such approaches were only interested in measuring DNS on the respective DNS ports. A study to measure non-DNS behaviour on DNS infrastructure is hereby still missing.

4 Methodology

Contents

4.1 Third-Party Data Sources

4.1.1 OpenINTEL Queries

To gain a global picture of the authoritative side of DNS would, ideally, require a global viewpoint of all DNS zone files. The problem is, however, that, with a few exceptions of zones that allow public access to their zone data for transparency reasons [38, 28], what is more commonly the case is that zone data are kept private, both for security and privacy reasons. As a result, the typical means of authoritative data reconnaissance is with DNS resolution, relying on data passively observed or actively queried.

This research deployed existing active measurements of NS records, A records and MX records to create lists of authoritative nameservers, webservers and mailservers respectively. These data were made available by OpenINTEL [60], an active DNS measurement platform developed in collaboration with the University of Twente back in 2015 with the intention to scan the global namespace at scale.

OpenINTEL measures hundreds of millions of domains on a daily basis. This coverage, however, though significant, is not exhaustive. For one, OpenINTEL covers an important portion of gTLDs and ccTLDs and their delegations but not all of them. For

the domains included in the measurements, the process of DNS resolution is not exhaustive either. Querying the nameservers of a zone, for example, does not cover every nameserver but instead it stops at the first responsive one. As a result, the KINDNS investigation presented in this research is not exhaustive either, concerned only with a subset of zones and a subset of their servers.

4.1.2 Tranco Domain Ranking

An important factor to consider when discussing poor security practices is the potential impact these can have. Whether as a result of geographical location, daily traffic or visitor engagement, some servers can be considered to be more important than others. For DNS authoritative space, a common metric of importance is that of domain popularity, though the correctness and methodology behind the construction of known top lists such as Alexa [41] or Umbrella [42] is sometimes dubious [71]. Given that domain popularity rankings do not entail an immediate or even single construction approach [74], choosing to trust the importance reflected in one top list or the other becomes its own challenge.

Tranco [65] is a relatively recent ranking approach that seeks to minimize the bias in the construction of known lists by providing an estimation of the top 1 million domains based on data aggregated from *other* lists. Combined with OpenINTEL information, Tranco popularity was used to construct two different sets of important authoritative nameservers.

For one, the Tranco ranking in itself was used to filter a list of nameservers responsible for the top 100,000 most popular Tranco domains. Furthermore, the totality of Tranco domains was used to create a second list of important nameservers based on shared infrastructure between different zones. Whereas the first list took into account the ranking calculated and provided by Tranco, the second list considered all Tranco domains equal and filtered those nameservers responsible for multiple different domains appearing in the Tranco list, essentially depicting importance in the sense of increased domain responsibility. Throughout this work, the two lists of important authoritative nameservers are referred to as the most *popular* set of authoritative nameservers and the most *shared* set of authoritative nameservers respectively.

4.1.3 DACS Open Resolver Census

As part of ongoing DACS research, Yazdani et al. [89] run open resolver discovery scans on a weekly basis. Rather than performing a discovery of open resolvers from scratch, this research deployed this census to obtain a list of recursive target IPs to investigate further. This choice was made for the sake of experimental efficiency but also to adhere to the data deduplication principle, which proposes to avoid running unnecessary network measurements when the circumstances allow it [3].

Just because an IP, however, is classified as an open resolver in one weekly scan does not necessarily mean that it will remain an open resolver forever. In fact, many open resolvers trace back to dynamic IPs and customer devices which are configured to offer recursion by accident [45, 89]. As a result, the identification of open resolver IPs

does not stay entirely consistent between scans. To avoid investigating this portion of disappearing IPs throughout this research, the target list was generated as the intersection of multiple weekly scans, in this way creating a list of open resolvers that remained consistent over the course of 2 months. This timeslot was chosen as a best effort to exclude as many false resolvers as possible from the list while including even recent open resolvers, that is, resolvers that maybe did not appear prior to the 2 months yet stayed consistent ever since their appearance.

Access to the weekly open resolver discovery scans allowed for the composition of another dataset, one attempting to filter open resolvers based on their overall persistence through time. This dataset was calculated as another intersection, in this case with the weekly scans performed in the earlier stages of the project. This intersection reflected those resolvers that stayed persistent over the course of 2 years, intended as a metric of importance for open resolvers. Throughout this work, this list of important resolvers is referred to as the most *persistent* set of open resolvers.

4.1.4 Digineo Public Resolver List

A server that stays persistent through the course of time, however, is not necessarily of much relevance to the average Internet user. It may, for example, be the case that an open resolver receives very little traffic hence, under these lenses, it may be considered of very little importance.

Considering that popularity lists like Tranco are not publicly available for open resolvers, importance on the basis of popularity was challenging. Instead, a second list of important resolvers was estimated from public knowledge, namely, using Digineo's public DNS server list [24], maintained since 2009 and verified based on the truthful and timely nature of the servers' responses. Though the appearance of an IP in Digineo's list does not necessarily make the resolver relevant to many Internet users, it might make them more likely to be used, at least compared to unidentified resolver IPs. Throughout this work, this list of important resolvers is referred to as the most *known* set of open resolvers.

4.1.5 CAIDA Network Mappings

Rather than focusing merely on the operational practices of individual IPs, this research further took into account the origins behind individual servers, namely, the Autonomous System (AS) networks and organizations they belong to. Such topology mappings, unfortunately, are known to suffer in their accuracy, with some networks, for instance, falsely reflected in the connectivity graph [92] and other networks absent from the graph altogether [14]. Still, these limitations are not bound to individual datasets but rather they constitute a problem in network mappings in general. An entirely truthful reflection of Internet topology could thus not be obtained in general.

For this research, network mappings were performed using CAIDA's pfx2as dataset [10] and as2org dataset [9], composed of routing information collected through RouteViews [62] and registration information provided via WHOIS [19] respectively. CAIDA's mappings were used in combination with the results derived from this research but also

in combination with OpenINTEL data. In the first case, the intention was to focus on top lists of networks behind the highest number of IPs following certain KINDNS operational patterns. In the second case, the intention was to focus on top lists of networks behind most DNS market share. The latter analysis strived to examine yet another important subset of servers, as the networks dominating DNS market are naturally relevant to any Internet user. For authoritative space, market dominance was determined on the number of different .com SLDs under a network's responsibility. For recursive space, market share *could* be determined based on traffic observations of which resolvers receive the most requests. At the absence of such traffic distribution, however, no such list was generated for open resolvers. Throughout this work, the list of important networks on the basis of their market share is referred to as the most *dominant* set of networks.

4.2 Command-Line Measurement Utilities

4.2.1 kdig

Measuring DNS behaviour at scale, as is the case with OpenINTEL, requires a distributed and efficient approach for the sake of accuracy and time. However, when the target list of DNS servers is known and small enough, as is the case for this research, measuring DNS can be as simple as querying the servers from a constant vantage point as if any typical DNS client. This approach was used to perform two different types of measurements throughout this research, using simple UDP queries to identify authoritative nameservers that support recursion on port 53, but also advanced TCP queries to identify resolvers that serve DoT and DoH on ports 853 and 443 respectively.

Though the overall methodology was similar, the two discoveries relied on performing different queries. Namely, for identifying duplex authoritative nameservers, the servers were asked to resolve the NS record for the .com zone. This information was chosen for reflecting unauthorized knowledge for all of the target nameservers. Hereby, any server that successfully retrieved this information revealed itself as a recursion supporter. For identifying DNS-over-Encryption, on the other hand, the resolvers were requested to retrieve the A record of google.com. Indeed, requesting any other information could as well work, but the specific information was chosen as a best effort to receive a timely answer and with a minimum impact to the nameservers of google.com, assuming that the information is popular enough to already reside in one's cache.

The two discoveries also used different factors to base their classification. Namely, for discovering recursion support over a target list of authoritative nameservers, recursion support was decided on the actual content of a reply, expecting that a recursion supporter should successfully and truthfully reflect the NS records of the .com zone. The described discovery was made as an attempt to capture less obvious recursion support, namely, servers that did not appear in the known intersection of the authoritative with the recursive servers. For identifying DNS-over-Encryption support, on the other hand, the decision could merely rely on the responsiveness of the server. In other words, a resolver merely needed to reply with *some* DNS answer within the specified time frame. DoT and DoH resolvers were thus decided simply on the reception of a

NOERROR response but without checking the legitimacy of the response. Note that DoH classification further took into account a resolver's ability to respond to both the GET and the POST method, as this is a fundamental requirement of the DoH RFC.

In both the discovery of duplex and unencrypted DNS, the measurements were performed with the same open-source toolset, that is, kdig [64]. The tool was favoured over simple DNS lookup utilities for allowing advanced DNS queries such as DoT and DoH. The tool was also favoured over other advanced DNS lookup utilities for allowing flexible parameter tuning, among which the option to connect to a server using its IP rather than its hostname as well as the option to use or omit certificate verification in DoT and DoH queries.

The main challenge with using kdig in general comes down to efficiency, as the tool is not designed for scanning at scale. This limitation traces back to the process of DNS resolution itself, whose efficiency suffers from slow responsive DNS servers. To ensure a feasible experimental duration, the waiting time per query was limited to a maximum of 3 retries, each time with a maximum of 10 seconds timeout. The value of this timeout was chosen as a best effort to capture responses even from slow servers, with 10 seconds simulating more than double the waiting time of a typical Windows DNS client [49]. If a target server did not respond within this time frame, that is, within 30 seconds at most, then kdig terminated the connection, considering it unsuccessful.

4.2.2 openssl

In DNS-over-Encryption scans, kdig was configured to skip X.509 verification, as that would attempt to verify the certificate chain on the fly. Though it is important that certificate verification succeeds to guarantee the security benefits of DNS-over-Encryption, it is still legitimate for a DoT/DoH server to use a self-signed certificate and share it with its clients so they can verify the chain offline. Given no access to arbitrary self-signed certificates, attempting to verify the chain online using kdig would fail to capture legitimate DoT/DoH servers that use self-signed certificates. The certificates behind DoT/DoH successful connections were instead analysed *after* their initial discovery with kdig. This was made possible using the openssl toolkit [61], which allowed fetching the certificates on the relevant ports.

4.2.3 zmap

Identifying non-DNS behaviour, of course, required a completely different approach than the one described so far. Considering the large scale of ports that fall under non-DNS behaviour, using existing measurements did initially seem like the most practical solution. However, though there exist several projects that gather large volumes of non-DNS port measurements [12, 47, 25], these measurements do not follow the same methodology or even frequency. Relying on these measurements to estimate KINDNS readiness would thus fail to paint the most accurate picture of non-DNS behaviour over specific DNS infrastructure at a specific point in time. Running non-DNS measurements from scratch was thus deemed like a better approach.

Measuring the relevant KINDNS requirements on non-DNS behaviour, however, is

not quite trivial from a third-party perspective, better yet from a single vantage point. Unrestricted firewall protection may seem like the simplest to measure, by noting how DNS servers respond to arbitrary requests to non-DNS ports, yet checking all 65,535 ports of a server can be particularly time-consuming. And even if time wasn't an issue, such measurements typically suffer in their accuracy too, with middlebox interference often complicating the accurate distinction of different port states [72].

To account for these limitations, the measurements were performed with the use of zmap [95], which allowed timely TCP port scanning over the totality of 1,024 well-known ports [37]. The scanner was particularly designed for scanning at scale, which allowed the measurements to finish within reasonable time frames. This speed, of course, doesn't come for free, but rather it traces back to the tool's stateless nature. In turn, zmap comes with no flexibility in configuring timeouts to base its waiting time, which means that some of its false negative classifications may actually include slow responsive servers. Nonetheless, any type of scanner cannot avoid false classifications altogether.

4.2.4 zgrab

Checking, however, if a server has a port open is one thing, and checking what kind of service is on that port, if any, is a whole different thing. The latter requires more advanced scanning techniques, relying on sending and analysing application-specific packets, known as banners. This process makes the scanning more time- and resource-exhausting than the simple port probing, while on top it relies on testing the right port for the right service to produce accurate results. Assuming that services are always offered on their IANA-assigned ports, service fingerprinting is a lot easier. But there are plenty of instances where services are offered on non-standardized ports instead [73], which makes accurate service fingerprinting a force to be reckoned with.

As far as KINDNS is concerned, port probing alone suffices to identify violating servers. This research used banner grabbing more for the sake of understanding the intentions of violating servers, that is, whether open ports are open intentionally, indicated by high banner predictability, or whether they are perhaps open by accident, indicated by low banner predictability. To minimize the disturbance that banner grabbing may cause to the target servers, this second phase of identifying non-DNS behaviour was even more constrained than the first phase, focusing only on a few services on the most commonly open ports. These measurements were performed with the use of zgrab [93], developed to support efficient service fingerprinting over a number of popular protocols [94].

Other than ports and services, which have been discussed this far, KINDNS proposes a third type of non-DNS violation, that is, of DNS-irrelevant software being installed on a server. This kind of non-DNS behaviour was left outside of research scope because software fingerprinting has more complications and requires even more effort than service fingerprinting. Namely, here too accuracy is constrained by the predictability of where software is installed. For instance, it is much easier to identify the details of SSH software on a server that is known to run SSH on the IANA-assigned port 22, but much more difficult to fingerprint SSH software on a server that runs SSH on a non-standardized port. What's more, a server may not even run SSH on any port

whatsoever, yet it may still have SSH software installed due to outdated, accidental or default configurations. For this case of servers, identifying non-DNS software violations becomes impossible without internal network access. With that said, non-DNS investigation was limited to investigating only a few popular services supported by zgrab while disregarding software fingerprinting altogether.

4.3 Ethical Considerations

4.3.1 Network Availability

Network measurements are fundamental in understanding and improving the Internet, an instrument for organizations to test their own networks, but also a widely used tool for academic research. Especially when used in academia, where the target networks are typically not owned by the researchers, network measurements come with important challenges. One of these challenges comes down to increased traffic volume, which network measurements are inherently guilty of. Increased traffic volume can be particularly problematic, with the potential to cause significant harm, because it can impact the availability of a network and its delivery of services. This dimension of harm involves not just the target networks but also any network initiating the scanning traffic.

This research acknowledged the importance of ensuring network availability by taking a number of ethical steps in the scanning process. Firstly, to ensure that the measurements did not hinder the availability of any target network, all target servers were scanned in a random order. This reduced the possibility that multiple IP addresses belonging to the same network were scanned all at the same time. To further ensure that the measurements did not hinder the availability of the network initiating the traffic, efficient configuration choices were made with restraint. These choices included process parallelism and network bandwidth. Namely, given a single vantage point, that is, a single server located within the University of Twente, the measurements were performed using up to 2,000 parallel Linux processes and with a network bandwidth of no more than 100 Mbits per second.

4.3.2 Data Confidentiality

A different dimension of harm comes down to the data involved. Other than merely running the scans with ethical considerations in mind, it was also important to handle the data with a similar responsibility.

As the research was performed as part of the DACS research group at the University of Twente, a lot of data were provided through the group. An important first step was thus to sign a non-disclosure agreement with DACS to ensure that any form of information provided is to stay confidential and bound to its research objectives.

4.3.3 Research Transparency

On top of ensuring that measurements did not cause any type of harm, a further consideration was to remain transparent with regard to the research. This is a generally proposed principle in and outside research context, while also an explicit requirement for responsible Internet based on the guidelines of the CATRIN project [11].

To adhere to the requirement of transparency, in this way further allowing for accountability, the scanning vantage point was configured to serve a simple webpage when queried on its HTTP or HTTPS port, that is, 80 and 443 respectively. This way, any DNS operator that received traffic from the server could connect to one of these ports and read about the research and how it concerned them. The webpage also included a contact e-mail address, allowing operators to ask for more information, a change in the scanning process, or even to opt-out from the research.

4.4 Pipeline Overview

Having described the data and tools used to answer the established research questions, as well as the ethical considerations complementing them, it is left to summarize the overall experimental pipeline. This last part of the chapter seeks to summarize the steps enumerated in the various previous parts.

Having generated a target set of authoritative nameservers and one of open resolvers using existing, third-party measurements, the discovery of the three types of KINDNS violations was performed as follows:

1. For the discovery of unwanted recursion support in authoritative nameservers, the intersection of the authoritative and recursive datasets alone was used as a first, easy step to identify obvious duplex supporters, that is, servers already known to appear in both target lists. To further identify duplex support that could potentially lie outside the intersection, the authoritative target nameservers were actively queried using kdig to retrieve the set of nameservers for .com TLD. Given that no authoritative nameserver in the list should have access to this information, any nameserver that successfully and truthfully provided this information upon request was then considered a duplex nameserver.

2. For the discovery of DoT and DoH support, kdig was configured to query all candidate resolvers on their standardized port, that is, 853 for DoT and 443 for DoH. The connections required resolvers to retrieve the IP address of a popular domain, that is, google.com, this way seeking to retrieve a response in a timely manner and while causing a minimum disturbance to the nameservers of google.com. Assuming a resolver responded to the request with a NOERROR status code, then the resolver was classified as a DoT/DoH supporter, regardless of the truthfulness of its reply or the validity of its X.509 certificate. The latter was fetched and analysed with the help of openssl as a second step of the discovery to identify potential patterns behind X.509 deployment.

3. For the discovery of DNS-irrelevant ports and services, the methodology was the same for both authoritative and recursive infrastructure. First, the servers were

queried on all their DNS-irrelevant ports and up to port 1,023 with the use of zmap. The most commonly open ports identified by zmap were then also scanned using zgrab to identify if those ports also served their IANA-assigned service.

In all cases, the measurements strived to estimate both the level of KINDNS compliance but also the potential drivers behind certain operational practices. The latter analysis was performed using a number of different data, among which importance metrics and topology mappings. Ultimately, by combining pre-collected with novel data provided for a more efficient and ethical experimental pipeline, allowing the experiments to finish in a timely manner and while imposing a minimal impact to all networks involved in the measurements.

5 Results

Contents

5.1 Target Selection

The results presented in this chapter are based on separate datasets for authoritative and recursive target servers, as these are summarized in **Table 5.1** and **Table 5.2**. For authoritative space, these datasets concern only SLD authoritative IPs based on OpenINTEL queries performed on 5 February 2023. The list consists of a total of 638,854 IPs, which, based on CAIDA's mappings, is found to trace back to approximately 30 thousand different networks.

Data Characterization	Relevant IPs
Entire Authoritative Target Space	638,854
Most Popular Authoritative Space	6.58%
Most Shared Authoritative Space	10.04%

Table 5.1: Target Authoritative Nameservers and Important Subsets

Data Characterization	Relevant IPs
Entire Recursive Target Space	1,119,221
Most Persistent Recursive Space	33.33%
Most Known Recursive Space	3.69%

Table 5.2: Target Open Resolvers and Important Subsets

A similarly generic target list was used for open resolvers based on weekly scans performed as part of other research at DACS. This list concerns a total of 1,119,221 IPs which constitute the intersection of all scans performed between 2 January 2023 and 6 March 2023. As an average weekly scan reveals approximately 2.6 million resolvers, the calculated intersection leaves out an important majority of IPs that appear and disappear between the various scans performed over the 10 weeks period. Using CAIDA's mappings, the totality of these IPs is found to trace back to approximately 20 thousand different networks.

The remaining datasets shown in **Table 5.1** and **Table 5.2** represent important subsets of the original target datasets. For authoritative nameservers, their important subsets reflect a total of 42,023 IPs responsible for the top 100,000 most popular Tranco

domains, as well as a total of 64,126 IPs having a shared responsibility over multiple Tranco domains. These two lists of important authoritative nameservers correspond to 6.58% and 10.04% of the initial list of authoritative IPs respectively.

As for open resolvers, their important subsets include a total of 373,000 IPs that are persistent in both the initial target list and in the scans performed on 4 January 2021, as well as a total of 41,329 IPs that appear in Digineo's public, known list of open resolvers. In turn, these two lists of important open resolvers correspond to 33.33% and 3.69% of the initial list of recursive IPs respectively.

Lastly, **Table 5.3** focuses on yet another important dataset of servers, that is, the top 5 networks responsible for hosting more than half of SLDs under the .com zone. Despite the overall dominance of the top 5 networks of **Table 5.3**, the networks are not dominant in the context of this research, as they correspond to only a small subset of the investigated authoritative infrastructure, with GoDaddy and Neustar actually tracing back to a few hundreds of servers as opposed to Cloudflare, Google and Amazon whose contribution to the target dataset accounts for a few thousand servers.

		In General	In Research Context	
Network	Organization	.com SLDs	Authoritative IPs	Recursive IPs
AS44273	GoDaddy	24.84%	0.02%	N/A
AS13335	Cloudflare	12.26%	0.86%	0.23%
AS15169	Google	6.08%	0.19%	0.02%
AS16509	Amazon	5.93%	1.98%	N/A
AS397233	Neustar	3.36%	0.04%	N/A

Table 5.3: Top 5 Most Dominant Networks Hosting 52.46% of .com SLDs

Interestingly, though the top 5 most dominant networks of **Table 5.3** are meant to reflect dominance with regard to authoritative space, a subset of those networks, namely, Cloudflare and Google, are actually also dominating more than half of open resolver market, a finding recurrently identified via traffic observations of previous research [69, 36]. Despite the dominance of those networks in relation to their traffic handling, however, similarly with the case of authoritative infrastructure, the IPs behind the responsibility of Cloudflare and Google only constitute a non-dominant portion of the recursive infrastructure examined in this research.

5.2 Discovery of Duplex DNS

Though the SoD principle applies equally to authoritative and recursive infrastructure operators, a characterization of how the former behave allows more flexibility than the latter. Namely, given a target list of authoritative IPs, identifying which of them violate KINDNS by offering recursion can merely rely on querying these IPs to retrieve information outside their authorization. Given a target list of recursive IPs instead, identifying which of them violate KINDNS by having authorization over a zone requires a global view of *every* zone file, which is impossible to have. But given a target list of authoritative as well as recursive IPs, as is the case for this research, measuring the SoD principle can take a simpler approach, that is, to base the list of violating IPs on the intersection of the two lists.

The intersection revealed a total of 12,015 different IPs that appear in both lists, corresponding to 1.88% of the initial list of authoritative nameservers and 1.07% of the initial list of open resolvers. Still, though these numbers seem negligible, they do not trace back solely to negligible servers.

For authoritative infrastructure, **Figure 5.1** indicates that 12% of the overall duplex behaviour actually traces back to most important authoritative nameservers. The responsibility of popular and shared servers in this number is also quite similar, with a similar portion of servers even corresponding to an overlap between the two important lists. Surely, an overall importance impact of 12% may not account for an impressive portion of servers, but it is an indicator that unwanted recursion support is in fact a problem relevant even to more popular and shared zones.

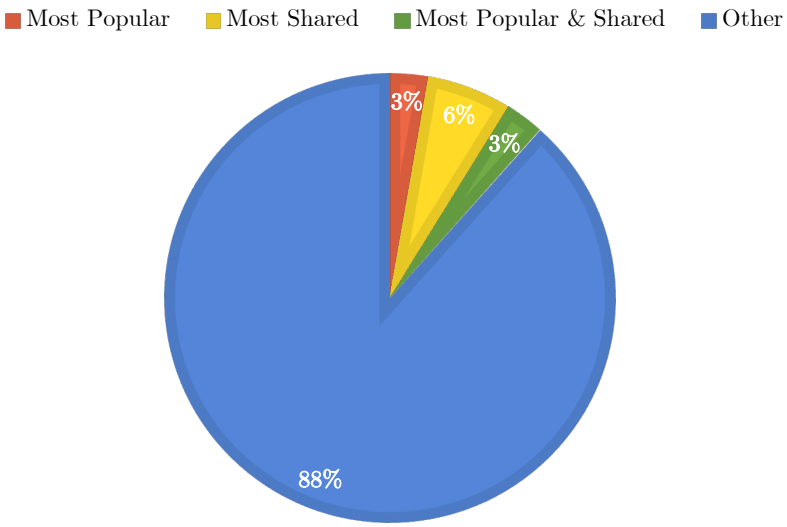

Figure 5.1: Impact of Important Authoritative Space on Duplex DNS Violations Identified *Inside* Intersection

As for recursive space, important infrastructure is even more prevalent in SoD violations, with **Figure 5.2** revealing that an impressive 68% of duplex behaviour traces back to most important resolvers, and particularly to most persistent ones. This finding implies that these resolvers may even serve duplex DNS for quite a long time, considering the persistence of most of them throughout at least a course of 2 years.

Regarding KINDNS, the calculated intersection is a definite indicator of violating servers. But just because a server does not appear in the intersection does not mean it is adhering to the relevant KINDNS requirement, as the intersection only includes a specific viewpoint of authoritative space and only a consistent viewpoint of open resolvers in the last months. The remaining of this section concerns measurements on all the remaining authoritative nameservers outside the intersection with the intention to identify more recent recursion support and non-ideal responses. For open resolvers outside the intersection, no further attempt was made to evaluate them, as that would require a global viewpoint of all potential zone files that a resolver may appear in. Rather, the duplex behaviour of open resolvers was naturally constrained

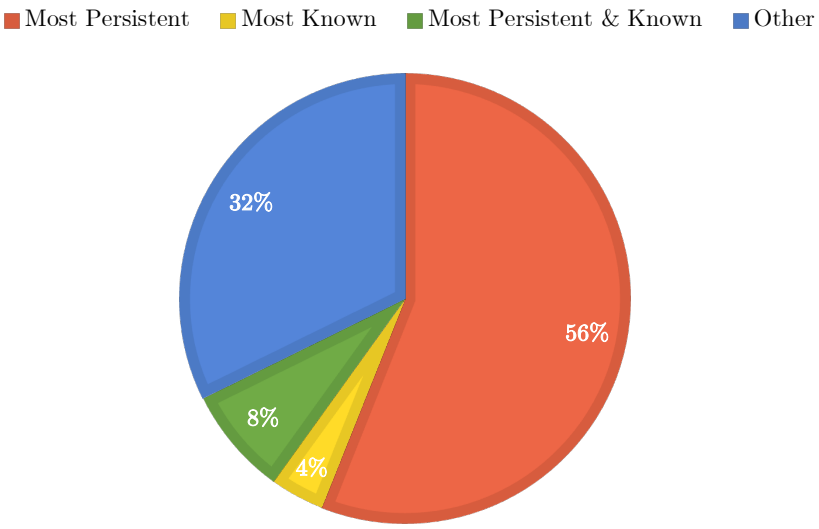

Figure 5.2: Impact of Important Recursive Space on Duplex DNS Violations Identified *Inside* Intersection

by the viewpoint provided by OpenINTEL.

As part of identifying problematic authoritative nameservers outside the intersection, a total of 626,839 authoritative IPs were queried to resolve the NS record of TLD .com, a piece of information that no SLD responsible nameserver, as is the case with this list, should be able to retrieve. These measurements were performed on 23 March 2023 and with a scanning duration that lasted approximately 3 hours.

Table 5.4 classifies the nameservers based on their response status, revealing a diversity of responses. Namely, though an important majority of servers responded with a REFUSED or a NOTAUTH status to the unauthorized request, in both cases declaring an explicit denial to serve the desired information, the behaviour of the remaining of servers is less explicit.

Query Response	IP Volume	IP Ratio
No Status	118,544	18.91%
REFUSED	471,431	75.21%
NOTAUTH	484	0.08%
SERVFAIL	7,042	1.12%
NXDOMAIN	2,367	0.38%
NOERROR	26,965	4.30%
Other Status	6	0.00%

Table 5.4: Types of SLD Authoritative Nameservers' Responses to Unauthorized TLD Queries

For one, a bit less than 19% of servers did not respond at all; indeed, they did not serve the information, but they did not declare an explicit denial to do it either. Secondly, a bit more than 1% of servers responded with a SERVFAIL status, indicating an inability to answer to the specific query, but without any clear indicator as to why

that is; a server may as well reply with a SERVFAIL for any indeliberate other failure, thus it remains unclear what caused the failure in this case. An even less ideal response is that of NXDOMAIN, which 0.38% of servers replied with. This type of answer is actually communicating false information, which is that the NS record of .com does not exist, in turn implying knowledge that the server does not have. Given the falsehood in this kind of response, it seems more like a problematic configuration choice than anything else.

But the most interesting kind of response is that of a NOERROR, which an impressive portion of 4.30% is guilty of. This kind of response implies a valid response, which in turn implies a responsive resolver. But the content of the provided NOERROR responses does not agree with this assumption.

Table 5.5 classifies the nameservers that responded with a NOERROR based on the legitimacy of their replies. The results reveal that a significant majority of almost 80% of these responses do not even contain a pointer to the nameservers of the desired domain. Like the case of NXDOMAIN responses, such NOERROR responses seem to trace back to odd configuration choices rather than real recursion support.

Response Content	IP Volume	IP Ratio
No Pointer to NS .com	21,067	78.13%
Invalid Pointer to NS .com	2,084	7.73%
Valid Pointer to NS .com	3,814	14.14%

Table 5.5: Legitimacy of SLD Authoritative Nameservers' NOERROR Responses to Unauthorized TLD Queries

As for the remaining NOERROR responses that *do* contain an answer section with some pointer to the .com nameservers, these are split into two categories, depending on the legitimacy of information they reply with. Because the NS records of .com zone are non-location dependent, retrieving this information should not differ between different stub or recursive resolvers. Yet this wasn't the case for 7.73% of NOERROR responses, which, though claimed to contain pointers to the .com nameservers, these pointers did not reflect DNS truthfully. Such configurations typically trace back to nameservers responsible for parked domains, that is, domains that are registered but not associated with any service. Since these domains are not functional as such, their nameservers are purposefully configured to redirect users somewhere else, regardless of what is asked in the query. Consequently, the portion of servers that replied in such way is not indicative of real recursion support but rather specific configurations.

As for the remaining NOERROR responses, these were indeed truthful depictions of .com, corresponding to a total of 3,814 duplex servers outside the initial intersection. Adding this value to the number of 12,015 servers identified through the initial intersection, this leads to a totality of 15,829 duplex servers that violate the SoD principle, that is, an overall 2.48% of all the target authoritative nameservers.

Just like with duplex behaviour identified through the intersection, **Figure 5.3** reveals that duplex behaviour outside the intersection traces back to a similar portion of important authoritative nameservers, namely, to a bit more than 12%. This impact is again similarly split between different importance metrics. Naturally, since this list of duplex servers includes only authoritative nameservers that do not appear in the intersection with open resolvers, importance impact can only be discussed in the

context of authoritative space.

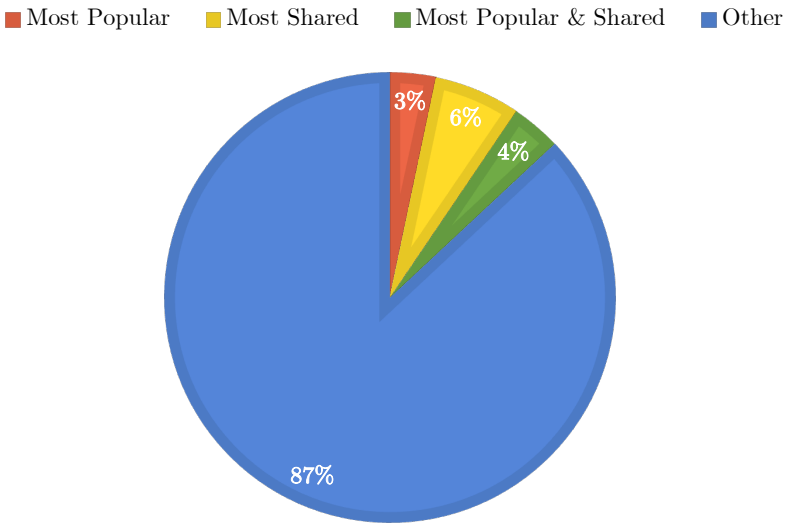

Figure 5.3: Impact of Important Authoritative Space on Duplex DNS Violations Identified *Outside* Intersection

Lastly, it is still left to examine if anything drives duplex behaviour in general. **Figure 5.4** illustrates the logical relationships between the top 15 networks behind the majority of violating IPs identified inside and outside the intersection. **Figure 5.4** reveals that the two duplex datasets are in fact quite interconnected, with 11 out of 15 networks in each one's list of top 15 actually re-appearing in the top 15 list of the other. Though the datasets are individually responsible for only 23.50% and 19.06% of all violations inside and outside the intersection respectively, their commonalities suggest that duplex behaviour is somewhat centralized, with a few networks being vastly responsible for duplex servers in general, regardless of the weekly stability of those servers in open resolver discovery scans.

Naturally, the commonalities between the top 15 networks in the two lists lead to a quite identical set of organizations responsible for most SoD violations in general. **Table 5.6** ranks the top 15 most insecure networks with regard to the volume of insecure IPs under their responsibility. **Table 5.6** reveals networks of diverse size and different countries, corresponding to an equal share of hosting providers and telecommunication providers, but no primarily DNS providers. **Table 5.6** further reveals the operational consistency of those networks, that is, the portion of violating IPs under a network's responsibility with relation to the totality of the network's IPs present in the target data. The calculated consistency of all networks is small to negligible, suggesting that duplex violations, though more prevalent in some networks, are actually quite isolated in their frequency. In turn, this makes duplex violations more relevant to individual server configurations rather than any systematic network effort.

When discussing the top 15 most violating networks of **Table 5.6** in the context of authoritative DNS market share presented in **Table 5.3**, neither GoDaddy, Cloudflare, Google, Amazon nor Neustar appear to play an important role in SoD violations. This

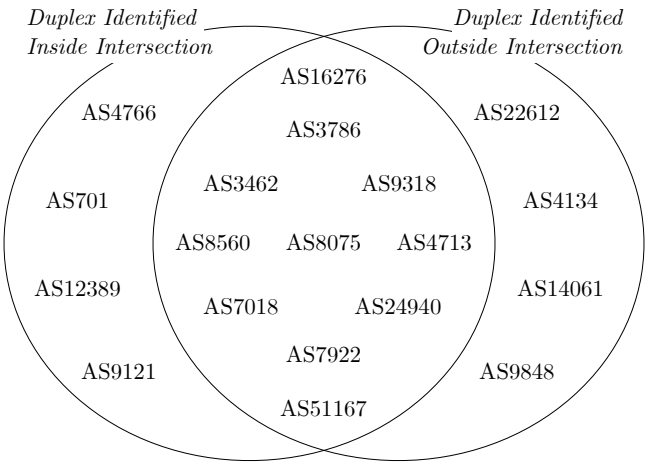

Figure 5.4: Logical Relationships Between Top 15 Networks Driving 23.50% of Duplex Behaviour *Inside* Intersection and Top 15 Networks Driving 19.06% of Duplex Behaviour *Outside* Intersection

Network	Organization	Violating IPs	Consistency
AS4766	Korea Telecom	569	29.12%
AS16276	OVHcloud	512	1.21%
AS3462	Chunghwa Telecom	423	17.94%
AS9318	SK Broadband	285	33.29%
AS3786	LG Dacom	228	25.76%
AS8075	Microsoft	221	7.57%
AS4713	NTT Com	207	8.59%
AS24940	Hetzner	167	0.71%
AS8560	Ionos	159	1.78%
AS7922	Comcast	142	7.16%
AS7018	AT&T	124	7.12%
AS51167	Contabo	123	1.71%
AS14061	DigitalOcean	102	0.93%
AS12389	Rostelecom	102	11.45%
AS9121	Turk Telekom	94	16.79%

Table 5.6: Operational Consistency in Top 15 Networks Driving 21.85% of SoD Violations

finding is reassuring, as the most dominant networks could be more susceptible to abuse because of their importance in the DNS ecosystem. Still, the fact that these networks do not appear in **Table 5.6** does not exclude the possibility that the networks aren't in fact serving duplex DNS in a smaller, less influential degree.

5.3 Discovery of Unencrypted DNS

The measurements presented in this section were collected on 19 March 2023 after a scanning duration of approximately 24 hours. This time entails three distinct connection attempts to each candidate resolver, including an attempt to connect to port 853,

another attempt to connect to port 443 using the GET method and /dns-query as endpoint, and a last attempt just like the second but using the POST method instead.

Despite the passage of more than 5 years since the standardization of DNS-over-Encryption protocols, their deployment levels seem to be significantly lagging behind. For DoT, the measurements disclose only 11,858 successful connections with the standardized implementation, corresponding to 1.06% of all target open resolvers. For DoH, the numbers are even smaller, with a total of 2,295 successful connections, corresponding to barely 0.21% of all target open resolvers. Though supporting both protocols is not a KINDNS requirement, the degree that this happens is quite frequent with regard to the totality of DoH support, with a total of 2,085 resolvers supporting both protocols, corresponding to 90.85% of all DoH resolvers. KINDNS requires supporting either one of the two protocols. This corresponds to a total of 12,068 compliant resolvers that serve either DoT, DoH, or both, corresponding to 1.08% of all target open resolvers. Unfortunately, the remaining majority of 98.92% of resolvers does not support any of the two protocols, at least not using the standardized implementations. A summary of the aforementioned results is presented in **Table 5.7**.

DoT Support	DoH Support	KINDNS Compliance	IP Volume	IP Ratio
Yes	Yes	Yes	2,085	0.19%
Yes	No	Yes	9,773	0.87%
No	Yes	Yes	210	0.02%
No	No	No	1,107,153	98.92%

Table 5.7: Categorization of DNS-over-Encryption Compliance in Entire Recursive Target Space

The so far commentary on compliance fails to discuss the importance of certificates deployed on the DoT/DoH relevant ports. **Table 5.8** and **Table 5.9** aim to shed more light onto the practices behind DNS-over-Encryption support by looking at the types of certificates that DoT/DoH supporters deploy. For DoT, there is an equally important portion of OK and self-signed certificates, the latter only verifiable assuming that the providers share the certificates with their clients. For DoH, the majority of certificates are OK, allowing any client to verify them online. In both cases, a small but still important portion of certificates cannot be trusted due to broken certificate chains or exceeded expiration dates. Also in both cases, there is a small portion of certificates whose status is left unknown due to a failure to capture the relevant certificate deployed on the port. This may happen due to a missing SNI for example, where the request fails to specify which of the many certificates deployed on a port the server should return. Whatever the reason behind those certificate fetching failures, the status of those certificates remains unknown.

Verification Status	IP Volume	IP Ratio
OK	5,870	49.50%
Self-Signed Certificate	4,479	37.77%
Broken Certificate Chain	301	2.54%
Expired Certificate	252	2.13%
Unknown	956	8.06%

Table 5.8: X.509 Certificate Deployment of Successful DoT Connections

Verification Status	IP Volume	IP Ratio
OK	1,939	84.49%
Self-Signed Certificate	4	0.17%
Broken Certificate Chain	81	3.53%
Expired Certificate	126	5.49%
Unknown	145	6.32%

Table 5.9: X.509 Certificate Deployment of Successful DoH Connections

KINDNS is not explicit about certificate usage and how it translates to compliance or lack thereof. Assuming that self-signed certificates are legitimate, they can only be trusted if they are shared with the clients. Consequently, the discovered self-signed certificates have the potential to translate to KINDNS compliance if the servers actually share them with their clients and the clients agree to trust them. Consequently, the discovered deployment does not necessarily include only true positives, but it may in fact include a few false positives. As for false negatives, these may potentially trace back to unpredictable configuration choices, that is, if some of the servers with which connections failed actually *do* support DNS-over-Encryption using less obvious implementations. However, unconventional configurations could impose a somewhat inconvenience to clients when they connect to these resolvers, which is why it seems unlikely that the phenomenon would appear at scale. In any case, the discovered deployment may fail to present the most accurate picture, but, considering the low incentives behind non-standardized implementations, it seems unlikely that the number of missed supporters would be extreme.

As for what drives the discovered deployment, **Figure 5.5** and **Figure 5.6** attempt to identify to what degree important resolvers are responsible for DoT and DoH respectively. The results show that neither the persistence of a resolver through the course of time nor the public knowledge of a resolver in known lists have much to do with a resolver's DNS-over-Encryption practices, as both DoT and DoH support are evenly split to important and non-important resolvers alike.

Table 5.10 further seeks to address what drives secure operational practices by focusing on the top 15 networks responsible for the majority of compliant resolvers. The analysis reveals an overall centralization of mostly but not entirely American networks driving DNS-over-Encryption deployment, with the top 15 networks being responsible for 54.57% of all DoT/DoH support, and the top 5 of them even driving 45.59% of all support. Most of these networks, however, show an obvious inconsistency in their operational practices, with most of them being behind an impressive volume of non-compliant IPs, and only a few of them clearly favouring DNS-over-Encryption in all or most of the IPs under their responsibility. The latter are primarily DNS providers (Cloudflare, NextDNS, ControlD), yet there are also networks whose activities are not concentrated in DNS yet they too are highly secure (Daniel Cid, Excitel).

When discussing the top 15 most compliant networks of **Table 5.10** in the context of recursive DNS market share presented in **Table 5.3**, Cloudflare's dominance in open resolver market is further reflected in the overall DNS-over-Encryption deployment. Namely, Cloudflare not only plays a leading role in driving DNS-over-Encryption support, but it is also almost entirely consistent in its security posture, unlike the case of most of the top 15 most compliant networks.

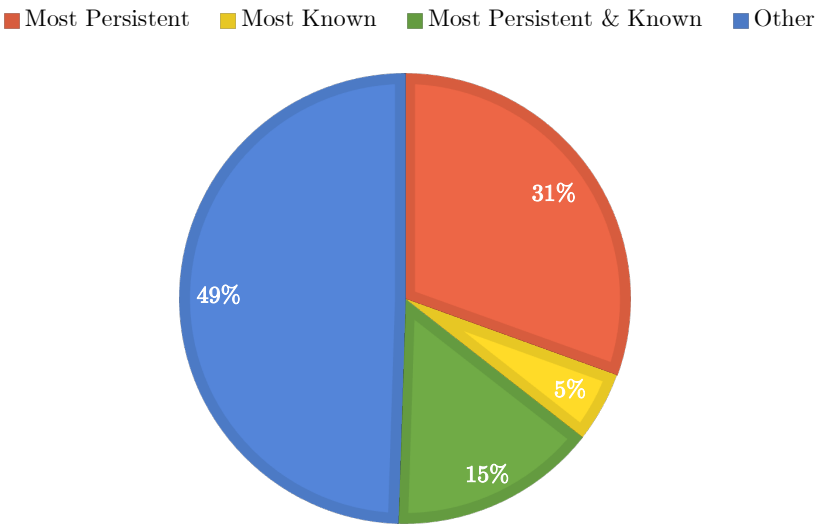

Figure 5.5: Impact of Important Recursive Space on DoT Deployment

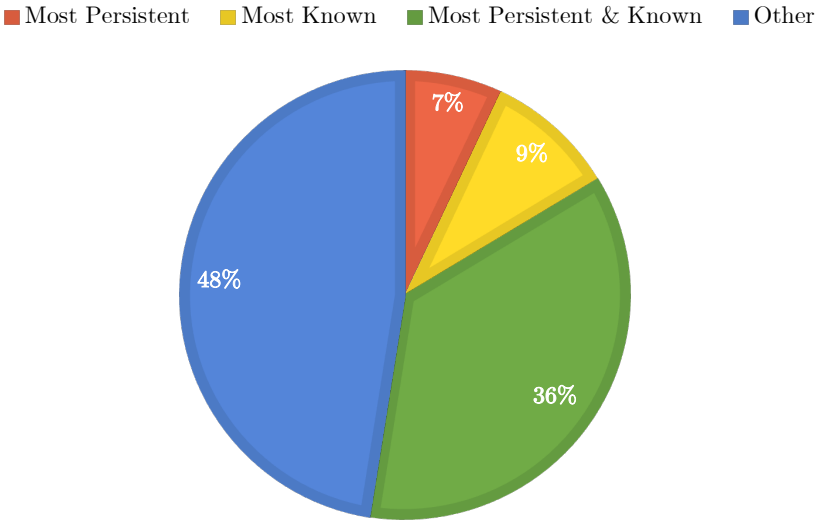

Figure 5.6: Impact of Important Recursive Space on DoH Deployment

5.4 Discovery of Beyond DNS

This last part of the analysis concerns behaviour that falls outside DNS needs. For both authoritative and recursive infrastructure, the measurements focus on all ports up to 1,023 included. For authoritative nameservers, non-DNS ports include any port other than DNS port 53; this entails that DoT and DoH ports 853 and 443 respectively also constitute violations. For open resolvers, the allowed ports include port 53, 853 and 443, as the last two are essential for running DNS-over-Encryption.

Network	Organization	Compliant IPs	Consistency
AS13335	*Cloudflare	2,537	97.84%
AS34939	*NextDNS	1,008	100.00%
AS205157	*Daniel Cid	851	99.18%
AS9318	SK Broadband	570	7.48%
AS4766	Korea Telecom	536	1.44%
AS398962	*ControlD	182	98.91%
AS31898	Oracle	151	12.40%
AS133982	*Excitel	119	92.97%
AS1257	Tele2	108	20.77%
AS16276	OVHcloud	102	1.82%
AS63949	Akamai	91	11.43%
AS3462	Chunghwa Telecom	88	1.67%
AS14061	DigitalOcean	87	4.23%
AS20473	Constant	79	13.81%
AS16509	Amazon	76	4.98%

Table 5.10: Operational Consistency in Top 15 Networks Driving 54.57% of DNS-over-Encryption Deployment

The measurements presented in this section were the most time-consuming to collect, with the initial port probing initiated on 21 March 2023 and finished after approximately 26 hours, and banner grabbing initiated on 23 March 2023 and finished after approximately 6 hours. These times include total durations for scanning both authoritative and recursive infrastructure.

Starting with the authoritative target servers, **Table 5.11** reveals quite some alarming tendencies happening on the 15 most commonly open ports. For one, both HTTP and HTTPS ports 80 and 443 are open and serving HTTP in approximately half of the target infrastructure. Given a known list of webservers discovered by OpenINTEL, the nameservers serving HTTP and HTTPS are quite evenly split between those that serve as webservers for their own domain and those that serve as webservers for a different domain, corresponding to 44% and 56% respectively. Both practices may come with their own share of risks, however, so whether one of them is more dangerous than the other remains subjective.

Table 5.11 further reveals a general tendency to serve e-mail protocols, among which SMTP, POP3, IMAP, as well as their TLS implementations. Similarly for the case of HTTP and HTTPS supporters, for the SMTP and SMTPS protocols too the number of authoritative nameservers that act as mailservers for their own domain compared to those that instead serve as mailservers for a different domain is quite evenly split to 42% and 58% respectively. Again, whether one practice exposes the server to more danger than the other remains subjective.

The FTP protocol for file transfers is also quite popular among authoritative nameservers, and so is SSH, though the latter is not responsive much to banner grabbing. Considering that the SSH protocol typically requires a valid username for the connection to succeed, this explains the lower predictability. Considering the high predictability in other services, of course, it seems unlikely that SSH wouldn't also show high protocol accuracy if proper usernames were used.

As for the few ports whose banner predictability is not presented in **Table 5.11**, this is because their expected protocols are not, at the point of performing this research

Port Number	IP Volume	IP Ratio	Assigned Service	Banner Predictability
80	329,131	51.52%	HTTP	80.43%
25	297,710	46.60%	SMTP	54.54%
443	290,191	45.42%	HTTP-over-TLS	90.53%
995	255,598	40.01%	POP3-over-TLS	92.67%
143	244,718	38.31%	IMAP	97.59%
465	234,048	36.64%	SMTP-over-TLS	91.21%
993	232,598	36.41%	IMAP-over-TLS	92.82%
110	231,403	36.22%	POP3	97.19%
587	230,504	36.08%	SMTP	96.56%
21	224,741	35.18%	FTP	96.20%
22	193,445	30.28%	SSH	3.60%
26	49,013	7.67%	Unassigned	Unmeasured
111	34,977	5.47%	SunRPC	Unmeasured
106	32,498	5.09%	3COM-TSMUX	Unmeasured
135	9,268	1.45%	EPMAP	Unmeasured

Table 5.11: Characterization of Top 15 Most Commonly Open Ports in Authoritative Target Space

at least, supported by zgrab. Whatever the behaviour on those ports thus remains unknown. Considering, however, the high banner predictability for most of the examined protocols, it seems very likely that the unmeasured services too are not open by accident.

When examining the most commonly open ports for recursive target space, **Table 5.12** reveals a very different picture compared to the authoritative target space. Namely, open resolvers seem significantly less guilty of unnecessary open port exposure compared to authoritative nameservers, though a lot of their open ports are the same. An important outlier that appears in **Table 5.12** but not in **Table 5.11** is Telnet, an inherently insecure protocol that uses plaintext message exchanges in its remote connections. The picture presented in **Table 5.12** is also different with that of authoritative target space in the sense that, with the exception of a couple of services, open resolvers have noticeably lower predictability in the banner grabbing. In turn, this implies that some of their open ports are perhaps open by accident, something that seems unlikely the case for authoritative infrastructure.

In the context of KINDNS, it doesn't matter if a server offers or does not offer a service on a port, or if a port is open intentionally or by accident; as long as the server has even a single non-DNS port open, this alone constitutes a violation. With that in mind, by merely looking at how many servers have their HTTP port 80 open, then it emerges that at least this number of servers is violating the relevant requirement.

When expanding beyond just the most commonly open port, the totality of violations is even more significant, with up to 70% of all authoritative nameservers having at least one port open, and a smaller but still important 24% of all recursive resolvers being guilty of the practice. These numbers are much more diverse than those identified in the investigation of duplex and unencrypted DNS, which revealed clear patterns in the operational patterns of most servers. Furthermore, non-DNS behaviour is also diverse with relation to the degree of a violation in itself, with some servers exposing a higher number of ports than others. The exact numbers behind these observations are presented in the leftmost graphs of **Figure 5.7** and **Figure 5.8** for authoritative and

Port Number	IP Volume	IP Ratio	Assigned Service	Banner Predictability
80	97,296	8.69%	HTTP	64.32%
22	68,652	6.13%	SSH	2.75%
23	51,679	4.62%	Telnet	42.71%
21	42,681	3.81%	FTP	81.18%
25	24,187	2.16%	SMTP	37.38%
135	23,443	2.09%	EPMAP	Unmeasured
110	20,419	1.82%	POP3	27.60%
139	15,304	1.37%	NetBIOS-SSN	Unmeasured
445	13,599	1.22%	SMB	98.02%
179	12,803	1.14%	BGP	Unmeasured
808	11,678	1.04%	Unassigned	Unmeasured
119	11,039	0.99%	NNTP	Unmeasured
88	10,033	0.90%	Kerberos	Unmeasured
554	8,736	0.78%	RTSP	Unmeasured
81	8,261	0.74%	Unassigned	Unmeasured

Table 5.12: Characterization of Top 15 Most Commonly Open Ports in Recursive Target Space

recursive infrastructure respectively.

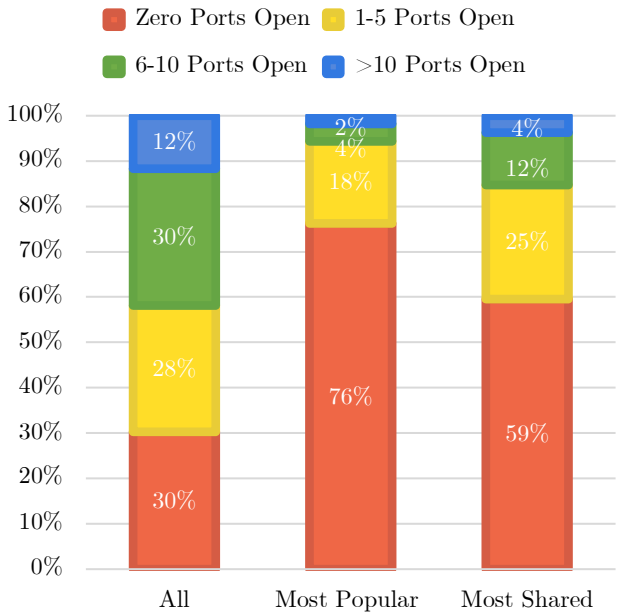

Figure 5.7: Comparison of Open Port Exposure Levels Between Entire and Important Authoritative Space

Despite the difference in the violation degree of authoritative and recursive space, the leftmost graphs in **Figure 5.7** and **Figure 5.8** both reveal a quite similar portion of servers guilty of having 1 to 5 open ports. For recursive resolvers, this portion is actually almost the same with the overall portion of violations, something that is not the case for authoritative nameservers, whose violations are more evenly distributed, expanding to higher open port exposure numbers. Given the centralization of open port

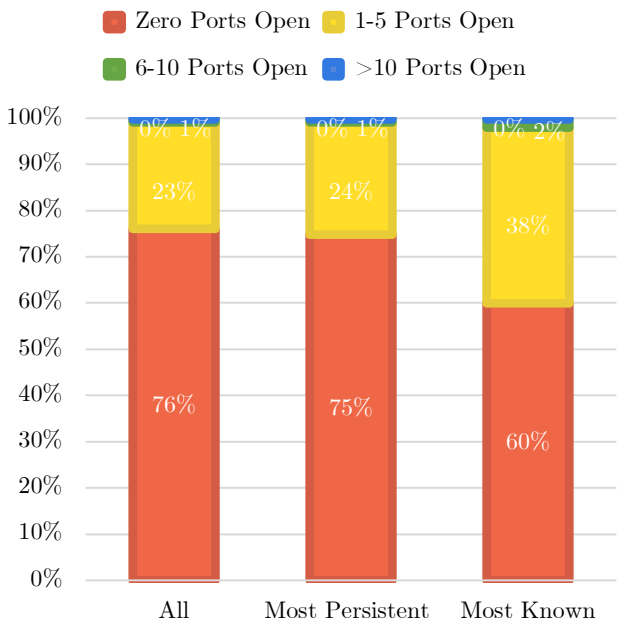

Figure 5.8: Comparison of Open Port Exposure Levels Between Entire and Important Recursive Space

exposure in recursive space, the configurations of open resolvers may not be accidental after all, unlike what their low banner predictability suggests.

As for how beyond DNS practices translate to more important infrastructure, the remaining graphs of **Figure 5.7** and **Figure 5.8** reveal very different pictures. For authoritative nameservers, **Figure 5.7** shows that important nameservers are far less guilty of unnecessary open port exposure, with only 24% non-compliant popular nameservers and 41% non-compliant shared nameservers, compared to an impressive 70% of non-compliant nameservers in general. Still, these numbers may seem more ideal, but the degree of affected nameservers is still alarming.

For open resolvers, on the other hand, **Figure 5.8** reveals that their hygiene in persistent recursive space is very similar to the hygiene of an average resolver. Namely, the case of persistent resolvers show almost identical compliance levels with their average counterpart, implying that both compliant and violating behaviour is just as much a responsibility of persistent and more recent resolvers. For the case of public, known resolvers, their compliance levels drop to 60%, yet this number is still better than in authoritative space, for the most part.

Considering there is no clear pattern in the non-DNS behaviour of neither authoritative nor recursive servers, there is little sense in discussing potential drivers of this diversity. Rather than examining beyond DNS in itself, it may be instead worth examining if the behaviour is in any way relevant to the other two kinds of services and *their* patterns. By simply looking at **Table 5.13**, the answer seems obvious; for authoritative nameservers behind non-compliant duplex DNS behaviour, **Table 5.13** shows that those servers are equally likely to be compliant and non-compliant with regard to non-DNS behaviour, whereas for open resolvers behind compliant encrypted

DNS behaviour, **Table 5.13** shows that those too are just as likely to be behind the compliant and non-compliant side of beyond DNS behaviour.

Dual Practice	Relevant DNS Role	Consistency Ratio
Duplex DNS & Beyond DNS	Authoritative	56.95%
Encrypted DNS & Strictly DNS	Recursive	65.35%

Table 5.13: Operational Consistency of Dual KINDNS Patterns in Entire Authoritative and Recursive Target Space

A similarly weak correlation between KINDNS patters is shown in **Table 5.14** and **Table 5.15** on the potential dual consistency patterns in the top 15 networks most responsible for duplex and encrypted DNS of **Table 5.6** and **Table 5.10** respectively.

Network	Organization	Repeating Violations
AS4766	Korea Telecom	46.05%
AS16276	*OVHcloud	90.23%
AS3462	Chunghwa Telecom	63.36%
AS9318	SK Broadband	62.81%
AS3786	LG Dacom	45.61%
AS8075	Microsoft	23.53%
AS4713	NTT Com	64.73%
AS24940	*Hetzner	92.81%
AS8560	Ionos	76.73%
AS7922	Comcast	48.59%
AS7018	AT&T	37.10%
AS51167	*Contabo	91.87%
AS14061	*DigitalOcean	83.33%
AS12389	Rostelecom	66.67%
AS9121	Turk Telekom	48.94%

Table 5.14: Insecure Operational Consistency Between Duplex DNS and Beyond Authoritative DNS in Top 15 Networks Driving SoD Violations

Starting with authoritative DNS, **Table 5.14** reveals that duplex and beyond DNS violations only sometimes go together. As almost all of the organizations presented in **Table 5.14** have an equal share of duplex servers with and without DNS-irrelevant open port exposure, this in turn implies that one violation is not necessarily suggestive of the other. An important exception to this includes a subset of networks that are in fact highly consistent in their insecurity. These consistent patterns of insecurity trace back to virtual private server providers (OVHcloud, Hetzner, Contabo, DigitalOcean), whose infrastructure can naturally be configured flexibly by their customers, taking into account their individual needs rather than any specific DNS security requirements.

The picture is slightly different with regard to secure operational practices in recursive space. Namely, **Table 5.15** suggests that DNS-over-Encryption support is somewhat relevant to more secure open resolvers in general, with around one third of the presented organizations being fully or almost fully consistent in serving both encrypted DNS and no DNS-irrelevant traffic. These consistently secure networks correspond to primarily DNS providing networks (Cloudflare, NextDNS, ControlD), but also telecommunication providing networks (SK Broadband, Korea Telecom, Tele2). For the remaining organizations, however, **Table 5.15** suggests that their proper pri-

Network	Organization	Repeating Compliance
AS13335	*Cloudflare	86.13%
AS34939	*NextDNS	100.00%
AS205157	Daniel Cid	43.83%
AS9318	*SK Broadband	100.00%
AS4766	*Korea Telecom	97.20%
AS398962	*ControlD	100.00%
AS31898	Oracle	10.60%
AS133982	Excitel	66.39%
AS1257	*Tele2	100.00%
AS16276	OVHcloud	23.53%
AS63949	Akamai	13.19%
AS3462	Chunghwa Telecom	72.73%
AS14061	DigitalOcean	25.29%
AS20473	Constant	39.24%
AS16509	Amazon	31.58%

Table 5.15: Secure Operational Consistency Between Encrypted DNS and Strictly Recursive DNS in Top 15 Networks Driving DNS-over-Encryption Deployment

vacy configurations are not complemented also with the necessary port restrictions, making these servers compliant to one practice but non-compliant to the other.

As for the most dominant networks of **Table 5.3**, with the exception of Cloudflare's impact on DNS-over-Encryption deployment, the operational patterns of the remaining cases are not discussed up to this point. **Table 5.16** seeks to bridge this gap by presenting the KINDNS behaviour of the top 5 most dominant networks in authoritative space as well as the KINDNS behaviour of the top 2 of those networks that are also dominant in recursive space.

		In Authoritative Space			In Recursive Space		
Network	Organization	Duplex	Beyond	Either	Unencrypted	Beyond	Either
AS44273	GoDaddy	0.00%	0.00%	0.00%	N/A	N/A	N/A
AS13335	Cloudflare	0.22%	18.77%	18.89%	2.16%	13.92%	15.73%
AS15169	Google	1.09%	69.43%	69.85%	92.24%	51.72%	96.98%
AS16509	Amazon	0.67%	48.39%	48.62%	N/A	N/A	N/A
AS397233	Neustar	0.00%	0.00%	0.00%	N/A	N/A	N/A

Table 5.16: KINDNS Violations in Top 5 Most Dominant Networks

Regarding GoDaddy and Neustar, **Table 5.16** reveals zero violations in their authoritative nameservers, a pattern that does not repeat itself in the other three networks. The good security posture of GoDaddy and Neustar may be justified by reflecting on the services provided by the two networks. Namely, GoDaddy and Neustar are primarily DNS hosting providers, with actually both of them operated by GoDaddy since Neustar's acquisition in 2020 [26].

Regarding the other three networks, their operational patterns are very diverse, with Google having noticeably higher violation portions and Cloudflare having noticeably higher compliance portions than the rest. From the perspective of their authoritative IPs, all three networks are responsible for some degree of duplex servers that is, however, smaller than the overall portion of duplex behaviour in authoritative space

in general. Their beyond DNS behaviour is more varied, with the portion of violating servers under Google's responsibility being almost as high as the overall portion of violations in authoritative space, whereas the respective violation degree under Cloudflare's responsibility is much scarcer than every other examined dataset of this research.

Google and Cloudflare further show similar patterns in their recursive space, both compared to their respective authoritative patterns but also compared to each other. Again, Google is responsible for a high level of unencrypted DNS violations that is only slightly improved to the overall level of unencrypted DNS violations, while its beyond DNS behaviour is even worse than the overall number of beyond DNS violations, with half of IPs under Google's responsibility having one or more unnecessary ports open. Cloudflare, on the other hand, is not just highly secure in its DNS-over-Encryption deployment, but also much less guilty of unnecessary open port exposure in its resolvers.

As for what potentially justifies these findings, the case of Cloudflare is very similar with that of GoDaddy and Neustar; Cloudflare too is primarily a DNS hosting provider that is particularly favoured for its increased security and performance features, which can explain the network's better security posture. Google and Amazon, on the other hand, offer a more varied set of products and services, among which DNS, which can in turn explain their varied security posture.

Ultimately, the aforementioned discussion on the top 5 most dominant networks may reveal their overall KINDNS readiness, that is, by looking at the degree of IPs under a network's responsibility that does not violate *either* of the relevant practices. The zero violating IPs under GoDaddy's and Neustar's responsibility naturally make them both KINDNS compliant in their totality. For the remaining networks, their compliance depends on the union of their other violations. As **Table 5.16** suggests, this compliance corresponds to approximately 30%, 50% and 80% of authoritative nameservers under Google's, Amazon's and Cloudflare's responsibility respectively. For recursive space, KINDNS readiness changes to barely 3% of the IPs under Google's responsibility and almost 85% of the IPs under Cloudflare's responsibility. In all cases, **Table 5.16** shows that the number of IPs that violate either practice is higher than the maximum number of violations regarding a single practice. This again verifies the previous findings that operational patterns in one practice only randomly seem to repeat in another practice, in turn making one configuration as unpredictable as the next.

6 Conclusions

Contents

6.1 Key Takeaways

This thesis aimed to provide a first glance into KINDNS readiness by investigating the configuration practices of a number of authoritative as well as recursive DNS operators. Sadly, the results suggest that, with the exception of a few dominant DNS hosting providers, most DNS operators are not yet ready to join KINDNS with their current practices.

So as to minimize the attack surface of individual DNS servers as well as that of the DNS ecosystem in general, KINDNS proposes that DNS servers keep to their intended role in the DNS ecosystem by offering only authoritative DNS as a service or only recursive DNS as a service but without duplexing the two services on the same port. This requirement is especially important for protecting the authoritative side of DNS from arbitrary and potentially malicious client requests that recursive resolvers are inherently prone to. Namely, though the size and information one can retrieve from an authoritative nameserver is entirely relevant to how that particular server is configured, open resolvers are by nature intended to communicate with arbitrary clients to retrieve information from arbitrary authoritative nameservers. This makes open resolvers great candidates for DDoS abuse when directing them to retrieve increased data volumes from particular vulnerable authoritative nameservers. Consequently, an otherwise securely configured authoritative nameserver can become highly insecure in this context if the server is also configured to offer recursion, be it intentionally or accidentally.

The investigation of duplex DNS violations revealed a small but not negligible portion of 2.5% of the target authoritative servers that support recursion despite of their authoritative identity. Though the overall number of discovered violations is small, the results are rather alarming, with more than two thirds of all violations tracing back to important open resolvers. As these violating servers may actually be susceptible to abuse for a long time and/or used by many Internet users, the potential harm in their duplex configurations is not to be underestimated. The networks behind these violations now are characterized by little centralization and even less consistency in serving duplex DNS. In turn, both these observations suggest that unwanted recursion support is more so an isolated phenomenon relevant to individual IPs rather than any clear tendency within specific networks.

The second kind of KINDNS practice examined in this research is that of unencrypted DNS violations. This requirement is concerned with protecting the privacy

and integrity of the communication between Internet users with recursive resolvers via the use of DNS-over-Encryption protocols. In configuring their recursive resolvers to serve either DoT or DoH, DNS operators can ensure that the information and truthfulness of DNS queries is not violated in any way during the message exchange.

Unlike with the investigation of duplex DNS, the investigation of unencrypted DNS revealed much more insecure results, identifying merely a bit more than 1% of DoT and/or DoH supporters. Unlike with duplex DNS, this investigation also revealed an obvious network centralization, with the majority of compliant IPs tracing back to only a few networks. A similar centralization did not repeat in the context of important space, as compliant resolvers were found to trace back equally to most persistent and known resolvers as well as less important ones. A similar lack of centralization was further observed in networks themselves, with most networks behind the highest volumes of DoT/DoH supporting IPs actually being responsible for even higher volumes of non-compliant IPs. As for the X.509 practices of the few supporting resolvers, these were found to include important volumes of self-signed certificates. This raises the question as to whether all of these certificates and consequently the servers can be trusted. If not, then the discovered KINDNS compliant servers may be even fewer.

So as to further minimize the attack surface of individual DNS servers as well as that of the whole DNS ecosystem as a result, a further KINDNS requirement forbids offering beyond DNS as a service. This requirement relates both to authoritative and recursive DNS operators, requiring them to commit to their role in the Internet ecosystem by serving only DNS and no other service on any other port. The harm in violating this requirement is less explicit than in the other two cases, since different services entail different vulnerabilities. A DNS server also acting as an HTTP server, for example, can be violated by a malicious input injection. Similarly, a DNS server also acting as an SSH server can be violated by a brute force password attack. Though the abuse possibilities of the two protocols are indeed different, both of them can lead to unauthorized access to a DNS server, yet its DNS role had nothing to do with the exploitation of these vulnerabilities. To account for these dangers, KINDNS proposes to protect DNS servers behind strict firewall rules and to restrict their functionality to serving only DNS protocols or DNS management protocols.

The investigation of beyond DNS revealed much more varied results compared to the other two investigations. The findings are more alarming for authoritative infrastructure and less so for recursive infrastructure, with 70% of authoritative servers having DNS-irrelevant ports open compared to 24% of recursive servers guilty of that behaviour. Still, though the overall number of violations between authoritative and recursive servers differs, the intentions of both appear more intentional than accidental, as a significant majority of the violating servers actually serve their IANA-assigned service on the relevant ports. Particularly, authoritative nameservers are mostly guilty of further acting as webservers and mailservers, half the time for their own domain, half the time for another domain. Recursive servers, on the other hand, are also somewhat guilty of web and mail protocols, while further guilty of some inherently insecure protocols such as Telnet. The picture between authoritative and recursive DNS operators further differs with relation to important infrastructure, as the port exposure of important authoritative nameservers is significantly smaller for most popular and shared domains. Important and less important open resolvers, on the other hand, follow fairly

similar security patterns. In any case, since the analysis focused on identifying open port exposure and specific protocol fingerprinting, the risk of a server's abuse is only implied in its potential. In other words, even though the number of servers responsible for beyond DNS behaviour is large, it is unclear how many of them actually have exploitable vulnerabilities and which of them would be of interest to an attacker in practice. Considering the overall volume of violating servers, of course, the danger is very much prevalent.

Ultimately, this research strived to identify potential drivers behind DNS operational practices with an aim to understand the relevance and implications of the results. Though the overall numbers of violations as such indicate an overall lack of readiness to join KINDNS yet, this observation does not apply to all networks or all kinds of networks. Namely, dominant networks like GoDaddy, Neustar and Cloudflare are noticeably more consistently secure in their configuration in all of the examined KINDNS practices compared to networks like Google and Amazon, the former being primarily DNS providers as opposed to the latter which offer a more diverse set of products and services. This finding is reassuring for organizations or individuals which outsource their DNS operations on the relevant DNS hosting providers. As for the case of organizations or individuals that operate DNS themselves, these seem to be important drivers of DNS insecurity. The high consistency in offering duplex and beyond DNS within virtual private server providers, for instance, serves as a good example of how individual poor configurations impact the overall DNS insecurity, most likely as an unfortunate result of lack of expertise.

Since KINDNS itself does not attempt to prioritize the importance of one practice over the other, whether one violation or another implies a higher danger is left unknown. In the end of the day, the importance of insecure operational practices is up to DNS operators to decide and, if they deem necessary, act upon. Yet these dangers seem not to be properly accounted for at the moment, either because of their implementation trade-offs or perhaps because there is still a lot way to go to reach an adequate level of DNS security expertise. Still, KINDNS is still in its very first steps. Considering the fairly recent nature of the new framework, as well as the relatively low attention that the specific practices examined in this research received in previous DNS research, it remains hopeful that the future shall hold a more ideal picture.

6.2 Future Work

For the sake of experimental do-ability, this research presumed a few assumptions and shortcuts in its measurements, which future research can address and build upon. First, the discovery of DoT and DoH support was performed using the most predictable and simple configuration choices, this way failing to present the most accurate picture. Future research may expand beyond conventional connection attempts so as to characterize further DNS-over-Encryption supporters. Similarly, the investigation of non-DNS behaviour was also limited to only a few ports and even fewer services. Taking into account the ethical implications that come with the measuring, future research may address non-DNS behaviour on different ports, services, or even software, the latter not being examined at all in this research. A more complete picture would further reside

in measuring multiple entities and/or paths in the DNS hierarchy, this way taking into account the relationships between DNS servers, which were not accounted at all as part of this research. Finally, value could also lie in repeating this research a while after the standardization of KINDNS, this way showcasing to what degree, if at all, DNS hygiene improved through the years.

References

[1] N. van Adrichem et al. *DNSSEC Misconfigurations: How Incorrectly Configured Security Leads to Unreachability*. 2014. In: IEEE Joint Intelligence and Security Informatics Conference.

[2] M. Allman. *Comments on DNS Robustness*. 2018. In: Proceedings of the Internet Measurement Conference.

[3] M. Allman and V. Paxson. *Issues and Etiquette Concerning Use of Shared Measurement Data*. 2007. In: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement.

[4] R. Arends et al. *DNS Security Introduction and Requirements*. 2005. Url: https://datatracker.ietf.org/doc/html/rfc4033.

[5] R. Arends et al. *Protocol Modifications for the DNS Security Extensions*. 2005. Url: https://datatracker.ietf.org/doc/html/rfc4035.

[6] R. Arends et al. *Resource Records for the DNS Security Extensions*. 2005. Url: https://datatracker.ietf.org/doc/html/rfc4034.

[7] AdGuard DNS Knowledge Base. *Known DNS Providers*. 2022. Url: https://kb.adguard.com/en/general/dns-providers.

[8] S. Bortzmeyer, R. Dolmans, and P. Hoffman. *DNS Query Name Minimisation to Improve Privacy*. 2021. Url: https://datatracker.ietf.org/doc/html/rfc9156.

[9] CAIDA. *Mapping Autonomous Systems to Organizations: CAIDA's Inference Methodology*. 2020. Url: https://www.caida.org/archive/as2org.

[10] CAIDA. *Routeviews Prefix to AS mappings Dataset (pfx2as) for IPv4 and IPv6*. 2008. Url: https://www.caida.org/catalog/datasets/routeviews-prefix2as.

[11] CATRIN. *Controllable, Accountable, Transparent: the Responsible Internet*. Url: https://www.catrin.nl.

[12] Censys. *Internet Scanning Intro*. Url: https://support.censys.io/hc/en-us/articles/360059603231-Censys-Internet-Scanning-Intro.

[13] R. Chandramouli and S. Rose. *Secure Domain Name System (DNS) Deployment Guide*. 2013. Url: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-81-2.pdf.

[14] H. Chang et al. *Towards Capturing Representative AS-Level Internet Topologies*. 2002. In: ACM SIGMETRICS Performance Evaluation Review, Volume 30.

[15] T. Chung et al. *A Longitudinal, End-to-End View of the DNSSEC Ecosystem*. 2017. In: Proceedings of the 26th USENIX Conference on Security Symposium.

[16] D. Clark and D. Wilson. *A Comparison of Commercial and Military Computer Security Policies*. 1987. In: IEEE Symposium on Security and Privacy.

[17] Cloudflare. *DNS amplification DDoS attack*. 2022. URL: https://www.cloudflare.com/en-gb/learning/ddos/dns-amplification-ddos-attack.

[18] D. Cooper et al. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. 2008. URL: https://datatracker.ietf.org/doc/html/rfc5280.

[19] L. Daigle. *WHOIS Protocol Specification*. 2004. URL: https://datatracker.ietf.org/doc/html/rfc3912.

[20] J. Damas and F. Neves. *Preventing Use of Recursive Nameservers in Reflector Attacks*. 2008. URL: https://datatracker.ietf.org/doc/html/rfc5358.

[21] C. Deccio and J. Davis. *DNS Privacy in Practice and Preparation*. 2019. IN: Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies.

[22] C. Deccio et al. *Quantifying and Improving DNSSEC Availability*. 2011. IN: Proceedings of 20th International Conference on Computer Communications and Networks.

[23] M. Denis, C. Zena, and T. Hayajneh. *Penetration testing: Concepts, attack methods, and defense strategies*. 2016. IN: IEEE Long Island Systems, Applications and Technology Conference.

[24] Digineo. *Public DNS Server List*. 2009. URL: https://public-dns.info.

[25] RIPE Atlas Docs. *Built-in Measurements*. URL: https://atlas.ripe.net/docs/built-in-measurements.

[26] T. Drennan. *Neustar Sells Its Registry Business To GoDaddy*. 2020. URL: https://www.home.neustar/about-us/news-room/press-releases/2020/neustar-sells-its-registry-business-to-godaddy.

[27] The Measurement Factory. *DNS Surveys*. 2005-2010. URL: http://dns.measurement-factory.com/surveys.

[28] The Swedish Internet Foundation. *Internetstiftelsen Zone Data*. 2019. URL: https://zonedata.iis.se.

[29] K. Fukuda, S. Sato, and T. Mitamura. *A Technique for Counting DNSSEC Validators*. 2013. IN: Proceedings of IEEE INFOCOM.

[30] S. Garcia et al. *Large Scale Measurement on the Adoption of Encrypted DNS*. 2021. IN: DBLP Computer Science Bibliography.

[31] O. Gusmundsson and S. Crocker. *Observing DNSSEC validation in the wild*. 2011. IN: Workshop on Securing and Trusting Internet Names.

[32] P. Hoffman and P. McManus. *DNS Queries over HTTPS (DoH)*. 2018. URL: https://datatracker.ietf.org/doc/html/rfc8484.

[33] Z. Hu et al. *Specification for DNS over Transport Layer Security (TLS)*. 2016. URL: https://datatracker.ietf.org/doc/html/rfc7858.

[34] C. Huitema, S. Dickinson, and A. Mankin. *DNS over Dedicated QUIC Connections*. 2022. URL: https://datatracker.ietf.org/doc/html/rfc9250.

[35] G. Huston. *DNS evolution: Trust, privacy and everything else*. 2020. URL: https://blog.apnic.net/2020/10/27/dns-evolution-trust-privacy-and-everything-else.

[36] G. Huston. *Looking at centrality in the DNS*. 2022. URL: https://blog.apnic.net/2022/11/22/looking-at-centrality-in-the-dns.

[37] IANA. *Service Name and Transport Protocol Port Number Registry*. 2022. URL: https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml.

[38] ICANN. *Centralized Zone Data Service (CZDS)*. URL: https://czds.icann.org.

[39] Y. Jia et al. *Open Doors for Bob and Mallory: Open Port Usage in Android Apps and Security Implications*. 2017. IN: IEEE European Symposium on Security and Privacy.

[40] Welcome to the Jungle - Coder Stories. *An interview with Paul Mockapetris, the creator of the DNS*. 2020. URL: https://www.welcometothejungle.com/en/articles/btc-interview-paul-mockapetris.

[41] Kaggle. *Alexa Top 1 Million Sites: Rankings of the top 1 million websites, in the world*. URL: https://www.kaggle.com/datasets/cheedcheed/top1m.

[42] Kaggle. *Cisco Umbrella List: Umbrella Popularity List*. URL: https://www.kaggle.com/datasets/adebayo/cisco-umbrella-list.

[43] D. Kaminsky. *It's The End Of The Cache As We Know It*. 2008. URL: https://www.blackhat.com/presentations/bh-jp-08/bh-jp-08-Kaminsky/BlackHat-Japan-08-Kaminsky-DNS08-BlackOps.pdf.

[44] KINDNS. *Knowledge-Sharing and Instantiating Norms for DNS and Naming Security*. URL: https://kindns.org.

[45] M. Kuhrer et al. *Going Wild: Large-Scale Classification of Open DNS Resolvers*. 2015. IN: Proceedings of the 2015 Internet Measurement Conference.

[46] APNIC Labs. *DNSSEC Validation Rate by country*. 2022. URL: https://stats.labs.apnic.net/dnssec.

[47] Rapid7 Labs. *Open Data*. URL: https://opendata.rapid7.com.

[48] SIDN Labs. *.nl statistics - DNSSEC*. 2022. URL: https://stats.sidnlabs.nl/en/dnssec.html.

[49] Microsoft Learn. *DNS client resolution timeouts*. 2023. URL: https://learn.microsoft.com/en-us/troubleshoot/windows-server/networking/dns-client-resolution-timeouts.

[50] E. Lewis and A. Hoenes. *DNS Zone Transfer Protocol (AXFR)*. 2010. URL: https://datatracker.ietf.org/doc/html/rfc5936.

[51] W. Lian et al. *Measuring the practical impact of DNSSEC Deployment*. 2013. IN: Proceedings of the 22nd USENIX conference on Security.

[52] C. Lu et al. *An End-to-End, Large-Scale Measurement of DNS-over-Encryption: How Far Have We Come?* 2019. IN: Proceedings of the Internet Measurement Conference.

[53] M. Luo et al. *Measurement for encrypted open resolvers: Applications and security*. 2022. In: Computer Networks, Volume 213.

[54] MANRS. *Mutually Agreed Norms for Routing Security*. URL: https://www.manrs.org.

[55] K. Mathew, M. Tabassum, and M. Siok. *A study of open ports as security vulnerabilities in common user computers*. 2014. In: International Conference on Computational Science and Technology.

[56] P. Mockapetris. *Domain names - concepts and facilities*. 1987. URL: https://datatracker.ietf.org/doc/html/rfc1034.

[57] P. Mockapetris. *Domain names - implementation and specification*. 1987. URL: https://datatracker.ietf.org/doc/html/rfc1035.

[58] B. Navamani, C. Yue, and X. Zhou. *An Analysis of Open Ports and Port Pairs in EC2 Instances*. 2017. In: IEEE 10th International Conference on Cloud Computing.

[59] T. Nijenhuis. *Discovery and Quantification of Open DNS Resolvers on IPv6*. 2018. In: University of Twente Student Theses: Repository home.

[60] OpenINTEL. *Active DNS Measurement Project*. 2015-2022. URL: https://openintel.nl.

[61] OpenSSLWiki. *Command Line Utilities*. URL: https://wiki.openssl.org/index.php/Command_Line_Utilities.

[62] University of Oregon. *Route Views Project*. 2022. URL: https://www.routeviews.org/routeviews.

[63] E. Osterweil, D. Massey, and L. Zhang. *Deploying and Monitoring DNS Security (DNSSEC)*. 2009. In: Annual Computer Security Applications Conference.

[64] Arch manual pages. *kdig - Advanced DNS lookup utility*. URL: https://man.archlinux.org/man/kdig.1.

[65] V. le Pochat et al. *Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation*. 2019. In: Proceedings of the 26th Annual Network and Distributed System Security Symposium.

[66] PowerDNS. *DNS Camel*. URL: https://powerdns.org/dns-camel.

[67] PowerDNS. *Migrating from using recursion on the Authoritative Server to using a Recursor*. 2022. URL: https://doc.powerdns.com/authoritative/guides/recursion.html.

[68] DNS Privacy Project. *Public Resolvers :: dnsprivacy.org*. 2022. URL: https://dnsprivacy.org/public_resolvers.

[69] R. Radu and M. Hausding. *Consolidation in the DNS resolver market – how much, how fast, how dangerous?* 2020. In: Journal of Cyber Policy, Volume 5.

[70] R. van Rijswijk-Deij et al. *A High-Performance, Scalable Infrastructure for Large-Scale Active DNS Measurements*. 2016. In: IEEE Journal on Selected Areas in Communications, Volume 34.

[71] K. Ruth et al. *Toppling Top Lists: Evaluating the Accuracy of Popular Website Lists*. 2022. In: ACM Internet Measurement Conference.

[72] Nmap Network Scanning. *Bypassing Firewall Rules*. URL: https://nmap.org/book/firewall-subversion.html.

[73] Nmap Network Scanning. *Clever Trickery*. URL: https://nmap.org/book/nmap-defenses-trickery.html.

[74] GitHub - sdstrowes/comp.md. *Alexa/Umbrella comparison notes*. URL: https://gist.github.com/sdstrowes/f75bc77581702bfad9467cb311c428a2.

[75] C. Shue, A. Kalafut, and M. Gupta. *The Web is Smaller than it Seems*. 2007. In: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement.

[76] H. Shulman and S. Ezra. *Poster: On the Resilience of DNS Infrastructure*. 2014. In: Proceedings of ACM SIGSAC Conference on Computer and Communications Security.

[77] H. Shulman and M. Waidner. *One Key to Sign Them All Considered Vulnerable: Evaluation of DNSSEC in the Internet*. 2017. In: Proceedings of the 14th USENIX Conference on Networked Systems Design and Implementation.

[78] R. Sommese, M. Jonker, and K. Claffy. *Observable KINDNS: Validating DNS Hygiene*. 2022. In: Proceedings of the 22nd ACM Internet Measurement Conference.

[79] S. Son and V. Shmatikov. *The Hitchhiker's Guide to DNS Cache Poisoning*. 2010. In: International Conference on Security and Privacy in Communication Systems.

[80] M. van Steen, G. Pierre, and S. Voulgaris. *Challenges in very large distributed systems*. 2011. In: Journal of Internet Services and Applications.

[81] Y. Takano et al. *A Measurement Study of Open Resolvers and DNS Server Version*. 2013. In: Proceedings of the Internet Conference.

[82] Privacy Tools. *Encrypted DNS Resolvers for Improved Internet Privacy*. 2022. URL: https://www.privacytools.io/encrypted-dns-resolver.

[83] O. van der Toorn et al. *Addressing the challenges of modern DNS a comprehensive tutorial*. 2022. In: Computer Science Review, Volume 45.

[84] M. Wander. *Measurement Survey of Server-Side DNSSEC Adoption*. 2017. In: Network Traffic Measurement and Analysis Conference.

[85] M. Wander and T. Weis. *Measuring Occurrence of DNSSEC Validation*. 2013. In: International Conference on Passive and Active Network Measurement.

[86] Z. Wang. *Understanding the Performance and Challenges of DNS Query Name Minimization*. 2018. In: IEEE International Conference on Trust, Security and Privacy in Computing and Communications.

[87] T. Wicinski. *DNS Privacy Considerations*. 2021. URL: https://datatracker.ietf.org/doc/html/rfc9076.

[88] GitHub - curl/curl Wiki. *DNS over HTTPS: Publicly available servers*. 2022. URL: https://github.com/curl/curl/wiki/DNS-over-HTTPS.

[89] R. Yazdani et al. *Mirrors in the Sky: On the Potential of Clouds in DNS Reflection-based Denial-of-Service Attacks*. 2022. In: Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses.

[90] Y. Yu et al. *Check-Repeat: A New Method of Measuring DNSSEC Validating Resolvers*. 2013. In: IEEE Conference on Computer Communications Workshops.

[91] L. Zembruzki. *Measuring Centralization of DNS Infrastructure in the Wild*. 2020. In: International Conference on Advanced Information Networking and Applications.

[92] M. Zhang et al. *How DNS Misnaming Distorts Internet Topology Mapping*. 2006. In: Proceedings of USENIX Annual Technical Conference.

[93] GitHub - zmap/zgrab2. *ZGrab 2.0: Fast Go Application Scanner*. Url: https://github.com/zmap/zgrab2.

[94] GitHub - zmap/zgrab2. *zgrab2/modules at master*. Url: https://github.com/zmap/zgrab2/tree/master/modules.

[95] GitHub - zmap/zmap. *ZMap: The Internet Scanner*. Url: https://github.com/zmap/zmap.