Dial "N" for NXDomain: The Scale, Origin, and Security Implications of DNS Queries to Non-Existent Domains

Guannan Liu
Colorado School of Mines
Golden, Colorado, USA
guannan.liu@mines.edu

Lin Jin
University of Delaware
Newark, Delaware, USA
linjin@udel.edu

Shuai Hao
Old Dominion University
Norfolk, Virginia, USA
shao@odu.edu

Yubao Zhang
University of Delaware
Newark, Delaware, USA
ybzhang@udel.edu

Daiping Liu
University of Delaware
Newark, Delaware, USA
dpliu@udel.edu

Angelos Stavrou
Virginia Tech
Arlington, Virginia, USA
angelos@vt.edu

Haining Wang
Virginia Tech
Arlington, Virginia, USA
hnw@vt.edu

ABSTRACT

Non-Existent Domain (NXDomain) is one type of the Domain Name System (DNS) error responses, indicating that the queried domain name does not exist and cannot be resolved. Unfortunately, little research has focused on understanding *why* and *how* NXDomain responses are generated, utilized, and exploited. In this paper, we conduct the first comprehensive and systematic study on NXDomain by investigating its scale, origin, and security implications. Utilizing a large-scale passive DNS database, we identify 146,363,745,785 NXDomains queried by DNS users between 2014 and 2022. Within these 146 billion NXDomains, 91 million of them hold historic WHOIS records, of which 5.3 million are identified as malicious domains including about 2.4 million blocklisted domains, 2.8 million DGA (Domain Generation Algorithms) based domains, and 90 thousand squatting domains targeting popular domains. To gain more insights into the usage patterns and security risks of NXDomains, we register 19 carefully selected NXDomains in the DNS database, each of which received more than ten thousand DNS queries per month. We then deploy a honeypot for our registered domains and collect 5,925,311 incoming queries for 6 months, from which we discover that 5,186,858 and 505,238 queries are generated from automated processes and web crawlers, respectively. Finally, we perform extensive traffic analysis on our collected data and reveal that NXDomains can be misused for various purposes, including botnet takeover, malicious file injection, and residue trust exploitation.

CCS CONCEPTS

• **Security and privacy → Network security**; **Web protocol security**; • **Networks → Naming and addressing**.

KEYWORDS

DNS, NXDomains, Expired Domains

IMC '23, October 24–26, 2023, Montréal, QC, Canada.

ACM ISBN 979-8-4007-0382-9/23/10.
https://doi.org/10.1145/3618257.3624805

ACM Reference Format:
Guannan Liu, Lin Jin, Shuai Hao, Yubao Zhang, Daiping Liu, Angelos Stavrou, and Haining Wang. 2023. Dial "N" for NXDomain: The Scale, Origin, and Security Implications of DNS Queries to Non-Existent Domains. In *Proceedings of the 32nd ACM Internet Measurement Conference (IMC '23), October 24–26, 2023, Montréal, QC, Canada.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3618257.3624805

1 INTRODUCTION

Domain Name System (DNS) is one of the core components on the Internet. It provides fundamental naming services to Internet users by performing domain name resolutions, which enable users to access specific Internet resources through simple and memorable domain names. Statistics show that there exist about 350 million domain names across all top-level domains (TLD) by the end of December, 2022 [23].

For decades, industry and research communities have devoted countless efforts to measure and improve DNS performance [47–50, 58, 76, 77, 79], enhance DNS security [38, 44, 52, 65, 84, 89], and propose novel DNS architectures [35, 57, 82, 93]. Nowadays, DNS has become a trusted platform for Internet users to receive valid and legitimate contents as desired. Users who attempt to access domain names that do not exist in DNS records (*e.g.*, they have never been registered) would receive "NXDomain" responses from DNS servers [26].

Previous studies [58, 81] have discovered that 10% to 42% of DNS responses are "NXDomain" responses. NXDomains typically come from the following three circumstances: (1) the domain name has expired, (2) the domain name has never been registered, and (3) the domain name has been taken down by the authorities [24]. Unfortunately, little research has focused on the scale, origin, and security implications of NXDomains from the perspective of user queries. In particular, many questions remain unanswered:

- Scale: How many DNS queries trigger NXDomain responses? Which NXDomains attract user queries? How much traffic do unregistered domains receive?
- Origin: What are the causes for domains to become non-existent? What are the detailed history and profiles of NXDomains?
- Security: Who are the visitors of NXDomains? Why do users visit unregistered domains? What security problems can arise from those visits to unregistered domains?

Previous studies have investigated the security aspects of expired domains [64, 66, 88], but such a security issue has not yet been extensively studied through the lens of NXDomains. In this work, we aim to address the aforementioned questions by conducting a systematic study of NXDomains. Our investigation is based on the Farsight passive DNS database [29] that extends more than 8 years (2014 to 2022). In total, we discover 1,069,114,764,701 DNS queries returning NXDomain responses. These DNS queries attempt to obtain the IP addresses of 146,363,745,785 NXDomains. We reveal that the number of NXDomains is over 225 times greater than the total number of registered domains. Our data analysis covers NXDomains collected from various vantage points by Farsight, including ISPs, enterprises, academia, and research organizations.

Furthermore, we investigate the origin of NXDomains in the Farsight database. We identify about 91 million NXDomains that have been previously registered, representing only 0.06% of the total number of NXDomains in the Farsight database. Among them, we uncover about 3 million NXDomains that are potentially generated by DGA (Domain Generation Algorithms). We also reveal 90,604 NXDomains that have been used for various types of domain squatting attacks [42]. Specifically, 45,175 NXDomains are registered to launch typosquatting attacks, 38,900 domains for combosquatting attacks, 6,090 are dotsquatting attacks, 313 for bitsquatting attacks, and 126 for homosquatting attacks. In addition, we cross reference 20 million randomly selected NXDomains with our domain blocklist. Using our domain blocklist and historical information from our research partner, we find 382,135 NXDomains used to host malware, 42,050 NXDomains containing grayware, 39,834 NXDomains used as phishing websites, and 19,868 NXDomains associated with Command and Control (C&C) activities.

Finally, we reveal the security implications reside in the NXDomain queries. We register 19 NXDomains with comparably high NXDomain queries for investigation. Inspired by many previous studies [85, 90], we deploy honeypots in the hosting server of our registered domains. The honeypots serve as a vantage point to collect network traffic for each domain. The data collection lasts for 6 months. We obtain 5,925,311 incoming HTTP/HTTPS traffic, which account for 81.7% of the total network traffic received by our registered domains. Our network traffic analysis suggests that the majority of the domain visits are originated from web crawler, automated process, referral, and user visits. Furthermore, we reveal that adversaries could take advantage of high-traffic NXDomains for potential botnet takeover, malicious files injection, and residual trust exploitation.

Our experiment is carefully designed to mitigate ethical concerns. The domains that we register in our work have been in NXDomain status for at least 6 months. This reduces the chance of our experiments interfering with the general public who may be interested in registering these domains. We establish our domain in cloud services with a landing page explaining the details of our study. We also provide our contact information in the landing page so that visitors can obtain more information from us if needed.

The major contributions of this work are summarized below:

- We comprehensively investigate the scale, origin, and security implications of NXDomains. To the best of our knowledge, this is the first large-scale measurement study with a focus on exploring Internet activities and security implications through the lens of NXDomains.
- We conduct measurement studies, showing the wide existence and long history of NXDomains. We reveal that a large number of NXDomains have frequently been receiving DNS queries even though they are in non-existent for an extended period of time. More importantly, we uncover that many NXDomains have been used for malicious purposes before they become non-existent.
- We establish 19 NXDomains and set up dedicated honeypots to analyze their network traffic. Different from existing research focused on recently expired domains, we select 19 domains that have been frequently receiving queries despite being non-existent for at least 6 months. We observe that some of the traffic to these NXDomains is still malicious, indicating their exploitation by adversaries.

2 BACKGROUND

The Lifecycle of Domain Names Domain registration and expiration procedures are administered by the Internet Corporation for Assigned Names and Numbers (ICANN) [11] and domain registrars. The domain registration policies are quite similar across the majority of registrars. Typically, users are allowed to register any legitimate domain names, given that their domain names must be uniquely identified within a top-level domain (TLD). Users can create new domain names that have never been registered before or claim the ownership of expired domains. Typically, domain names are initially registered for at least one year. After that, domain owners have the option of renewing their domains annually.

Domain owners have full control and the right to use their registered domain names unless they fail to renew the domains. In such a case, domain names enter the formal expiration procedure regulated by ICANN. The expiration procedure is described as the Expired Registration Recovery Policy of ICANN [7]. Specifically, registrars must notify domain owners about domain termination at least three times (two times before the expiration date and one time after). If domain owners fail to renew their domains during this period, the domains enter the Redemption Grace Period (RGP) of 30 days. Domain owners can still regain control over expired domains, but additional fees will be charged for domain restoration. After the RGP, the domains will be released to the public for everyone to register.

Since domain names are considered valuable assets, many domain registrars specialize in providing drop-catching services [2, 6, 20]. Once domain names enter the RGP, the drop-catching platform begins to advertise these domains to the public. If some users express interest in owning pending-deletion domains, drop-catching platforms will help users reserve these domains immediately after their releases. The rest of the domains are open to public registration through generic domain registrars.

DNS and NXDomains The domain name system (DNS) serves a critical functionality of the Internet for translation between domain names and IP addresses. Figure 1 illustrates an example of DNS successfully resolving `www.example.com` to its IP address. The user queries the local DNS server about the IP address of

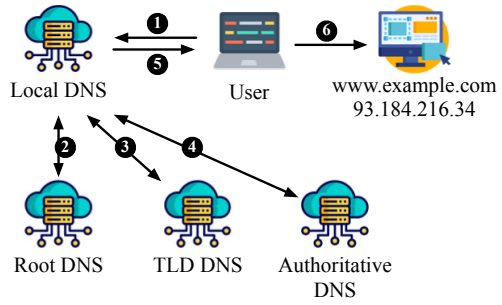

Figure 1: DNS resolution for *www.example.com*.

`www.example.com` (❶). If the local DNS server does not have the corresponding IP address, it begins to iteratively query the IP address in Root DNS (❷), Top Level Domain (TLD) DNS (❸), and authoritative DNS (❹). The IP address of `www.example.com` is returned to the user by the local DNS server (❺). Finally, the user can access `www.example.com` using the provided IP address (❻). DNS also incorporates caching functionality to minimize network traffic. For instance, if the DNS response for `www.example.com` has already been cached at the local DNS server, the response can be directly provided to the user via ❺, without the need to traverse ❷, ❸, and ❹.

If a queried domain name cannot be found in DNS, such domain is considered as NXDomain, *i.e.*, non-existing domain, and a response with the NXDomain error code is returned. Such an NXDomain response is different from a NOERROR DNS response with an empty answer, which represents that a valid domain does not contain the specific type of DNS record the user requires. Note that previous research [58] discovered that most NXDomain responses are caused by reverse IP lookups, but our work focuses on domain names that cannot be resolved to IP addresses.

Despite that DNS cannot successfully resolve NXDomains, some DNS queries are still frequently generated for an attempt to access certain NXDomains again and again, resulting in much more NXDomain responses than normal for this same set of NXDomains. In this paper, we focus on those queried domain names that trigger a significant amount of NXDomain responses. We analyze the network traffic of these domains to reveal the underlying security risks.

3 METHODOLOGY

In order to characterize and understand NXDomains, we collect and analyze both passive DNS data and active DNS traffic. We use Farsight's DNS database [29] to retrieve historical NXDomains collected by its collection servers all over the world. Moreover, we select and register 19 NXDomains, and host them on cloud services. We set up our own authoritative DNS server to resolve the registered domains. Then we design and deploy a honeypot, named NXD-Honeypot, to actively collect inbound network traffic received by the servers. Figure 2 illustrates the overview of our methodology.

3.1 Passive DNS Database

A passive DNS database contains historical DNS records, enabling researchers to investigate potential security risks and malicious activities on the Internet [39, 55, 94]. In particular, Farsight Passive DNS database [29] is one of the popular historical DNS traffic datasets contributed by collection servers from individuals and organizations around the world, providing rich and comprehensive DNS data for analysis. The Farsight Passive DNS data is collected from multiple vantage points, including users and many tiers of DNS servers. Consequently, DNS caching is unlikely to have a significant influence on the overall NX responses recorded within the Farsight Passive DNS database. To process the large-scale DNS database, we mirror the database to BigQuery servers.

3.2 NXDomain Analysis

To understand the scale and origin of NXDomains, we conduct a measurement study by analyzing the Farsight passive DNS NXDomains [30]. Our focus is on the DNS queries to NXDomains and the lifespan of NXDomains. This enables us to identify uncommon NXDomains for further analysis, in particular those with high DNS queries and long lifespan.

We are specifically interested in the origin of high-traffic NXDomains caused by domain expiration. To achieve this, we search their historical WHOIS information and leverage WhoisXML [31] that contains 15.6 billion historic WHOIS records. We aim to reveal how these domains are used before their expiration. This could provide us an in-depth understanding on why some expired domains still receive a significant amount of DNS queries, despite that they have become expired for a long time.

3.3 Domain Selection

To further explore the usage and security implications of NXDomains, we collect and analyze the inbound network traffic received by NXDomains. Unfortunately, it is infeasible and unnecessary to obtain the ownership of all NXDomains because (1) registering a large number of NXDomains requires massive financial costs and (2) processing the network traffic of NXDomains could be overwhelming. Therefore, we carefully select the representative NXDomains and register them for detailed investigation.

Our domain selection criteria are two-fold. First, we are particularly interested in domain names that receive a substantial volume of traffic, and thus we choose the NXDomains that receive more than 10,000 DNS queries per month based on the Farsight database. Second, we select NXDomains that remain in non-existent status for at least six months. This ensures that such domains (1) have been frequently queried over an extended period of time and (2) are not those domains that provide active services but are accidentally expired. Moreover, the NXDomains that we select should contain both benign and malicious domains. In particular, malicious NXDomains should fall into a variety of categories, including blocklisted domains, DGA-based domains, and squatting domains (Section 5.2). In total, we select 19 NXDomains for our study. We register these domains in different domain registrars, including 101domain [1], GoDaddy [10], and Namecheap [15]. We host our registered domains in cloud instances provided by both Amazon AWS and Google

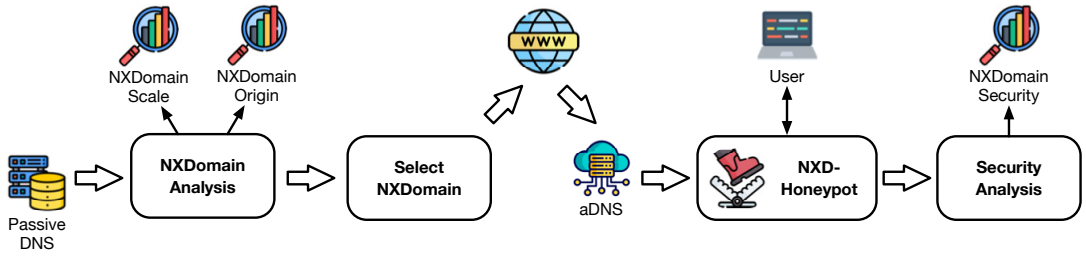

Figure 2: Overview of the methodology.

Cloud. We also establish an authoritative nameserver to resolve our domain names to the IP addresses.

3.4 NXD-Honeypot

To collect incoming queries for our registered domains, we set up a dedicated honeypot, NXD-Honeypot, for each of our domain-hosting servers. The NXD-Honeypot serves the functionalities of both a traffic recorder and a barebone web server. The traffic recorder accepts TCP and UDP packets from all well-known and standardized ports. It also collects detailed information about incoming traffic, including IP addresses, port numbers, and payloads. Our NXD-Honeypot is also equipped with our own traffic filtering mechanism to effectively reduce unwanted network traffic (we discuss the filtering mechanism in detail in Section 6.1). We conduct such experiments with careful ethical considerations and configurations (Appendix A).

Finally, we analyze the network traffic collected from NXD-Honeypot over a period of six months. We aim to understand who visits our registered domains and why they visit these domains. One challenge is to differentiate the traffic that specifically intends to visit our NXDomains from those that accidentally reach our NXD-Honeypots (*e.g.*, various crawlers and bots). We design comprehensive mechanisms and conduct careful experiments to address this challenge, and we also attempt to explore potential security risks residing in the collected traffic. Details of the experiments and observations from our NXD-Honeypots are presented in Section 6.

4 SCALE OF NXDOMAINS

In this section, we explore the scale of NXDomains using Farsight's passive DNS database. Specifically, we demonstrate that NXDomains widely exist on the Internet, as a large number of DNS queries trigger NXDomain responses. More importantly, numerous NXDomains have been in non-existent status for an extended amount of time, but still they frequently receive DNS queries.

4.1 NXDomain in the Wild

To comprehensively understand the scale of NXDomains, we first quantify the existence and prevalence of NXDomains in the wild. The Farsight passive DNS database utilizes Security Information Exchange (SIE) channel for NXDomain collection [8]. Acquired from this channel, we obtain a copy of the NXDomain data over the past eight years. The dataset contains 1,069,114,764,701 DNS responses with the "NXDomain" error. More specifically, these queries attempt to inquire about the IP addresses of 146,363,745,785 NXDomains. As statistics show [5], about 650 million domain names have been registered as of 2022. The significantly larger name space implies that NXDomains have likely been leveraged in uncommon usage patterns or potential exploitation, which has not been substantially understood.

Next, we compare the number of DNS queries that result in NXDomain responses collected by Farsight's database from 2014 to 2022. The distribution is shown in Figure 3. We can see that the average number of NXDomain responses per month increases from 2014 to 2016, and then the trend becomes relatively flat until 2020. In 2021, there is a steep rise, reaching an average of nearly 20 billion NXDomain responses per month. In 2022, this average number further increases to more than 22 billion.

4.2 Data Sampling

Unfortunately, with the tremendously large number of NXDomains in the wild, we are unable to process all NXDomains collected in the dataset due to resource limitations, even though we run the analysis with BigQurey server in the commercial cloud platform. Instead, we adopt a random sampling process with a sampling ratio of 1/1,000. This can effectively reduce the size of our data while maintaining the relative statistical distribution of the NXDomains in the wild. To this end, from the 146,363,745,785 NXDomains we acquired from the Farsight DNS database, we randomly select 146 million NXDomains for further analysis.

4.3 NXDomains under Different TLDs

In this study, our investigation is centered on NXDomains under various Top-Level Domains (TLDs). We have intentionally excluded the analysis of any subdomains. This is due to the fact that attackers can more easily register domain names under generic TLDs than leverage subdomains of legitimate domains for malicious purposes, and consequently, we have chosen to narrow our scope accordingly. Figure 4 illustrates the 20 most popular TLDs with the highest number of NXdomains, as well as the distribution of their NXDomain responses. Not surprisingly, we observe that `.com`, `.net`, `.cn`, `.ru`, and `.org`, which are the top 5 TLDs with the highest number of NXDomains, also receive the highest number of DNS queries for NXDomains. Meanwhile, all top five country code TLDs (ccTLDs), according to the Domain Name Stat [5], also appear in the top NXDomain TLD list. Both observations indicate that the distribution of the number of DNS queries for NXDomains aligns with the number of NXDomains in different TLDs.

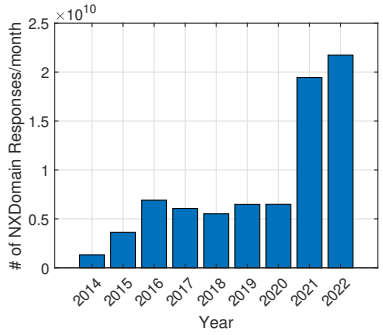

Figure 3: Average number of NXDomain responses per month between 2014 and 2022.

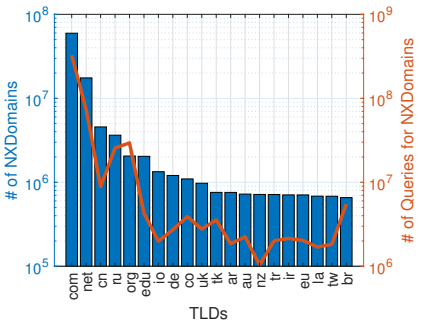

Figure 4: Distribution of the number of NXDomains and queries in different TLDs.

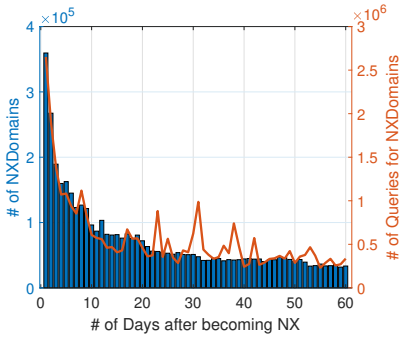

Figure 5: Number of NXDomains and their DNS queries across different lifespans.

4.4 NXDomain Lifespan

Many existing studies have shown that domains continue to receive a large amount of inherited network traffic immediately after their expiration dates [64, 85, 88]. Besides those expired domains, there is a larger number of NXDomains that have remained in non-existent status for an extended period of time (see Section 5.1). Their activities and long existence, however, have not yet been comprehensively examined. We discover 1,018,964 NXDomains receiving a total of 107,020,820 DNS queries as of 2022, while they have been in non-existent status for more than 5 years.

We investigate 146 million NXDomains in total after the sampling process in Section 4.2. The blue bar graph in Figure 5 illustrates the number of NXDomains that frequently receive DNS queries within 60 days of being in non-existent status. It is obvious that the number of such NXDomains decreases considerably in the first ten days. This is reasonable since one can be aware of the domains becoming available and start registering these domains. After ten days, the decrease becomes much slower, indicating that the remaining NXDomains become less likely to be registered by the general public.

A similar trend also appears in the number of DNS queries for such NXDomains but with high fluctuation, as shown in the line graph of Figure 5. The number of DNS queries decreases at a relatively similar rate compared to the number of NXDomains. This is different from our expectations. We anticipate that the number of DNS queries should drop faster because users are aware of the domains becoming non-existent and should have stopped sending requests to these domains. However, the similar trend of the bar and line graphs shown in Figure 5 suggest that domains continue receiving DNS queries despite their non-existent status.

We further explore the difference in DNS traffic before and after a domain becomes non-existent. Specifically, we randomly select 10,000 NXDomains that frequently receive DNS queries for more than two years in non-existent status. Figure 6 shows the average number of DNS queries 60 days before and 120 days after these domains become non-existent. On one hand, we observe a spike that appears around 30 days after the change of domain status (highlighted in the red circle in Figure 6). This clearly indicates that NXDomains receive significantly more DNS queries about 30 days since they first appear in the Farsight database, although we are unsure of the cause of this spike. The number of queries even exceeds that before domain expiration. On the other hand, the result does show a decrease in DNS queries overall after the domains expire. This aligns with our previous observations that domain expiration is not the only factor that affects the DNS traffic received by NXDomains, indicating the involvement of those NXDomains with abnormal network behaviors.

5 ORIGIN OF NXDOMAINS

In this section, we explore the history and profiles of the 146 billion NXDomains from Farsight's passive DNS database to identify their origins. In particular, we explore whether these NXDomains have been exploited for malicious activities.

5.1 NXDomain History

The domain name history (also known as WHOIS history) contains various domain ownership information, such as registration/expiration dates, domain registrars, past DNS records, *etc.* To investigate the history of domain ownership and registration, we join the NXDomains identified in the Farsight passive DNS database with the WHOIS history database [31] to reveal an NXDomain's registration history.

In Section 4.1, we identify 146,363,745,785 NXDomains over the eight years from the Farsight passive DNS database. By retrieving from the WHOIS database, we find that 91,545,561 (0.06%) NXdomains have a valid registration record. These domains have passed the expiration date, and their previous owners did not renew their registration.

Meanwhile, 146,272,200,224 NXDomains have no historical registration records. While we do not have enough computing power to process all the never-registered domains, we anticipate that a significant portion of these domains are likely algorithmically generated domains (*i.e.*, DGA-based domains). This has also been confirmed by Plohmann *et al.* [80], who discovered that only 0.62% DGA-based domains are actually registered. These algorithmically generated domains are typically used to confuse detection algorithms with no other legitimate purposes. Furthermore, it is possible that some of these never-registered domains are the result of mistyped domain

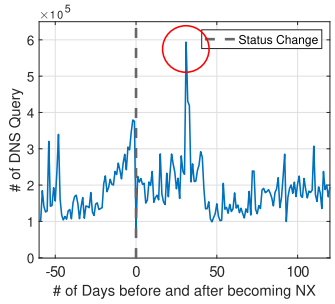

Figure 6: DNS queries before and after a domain become non-existent.

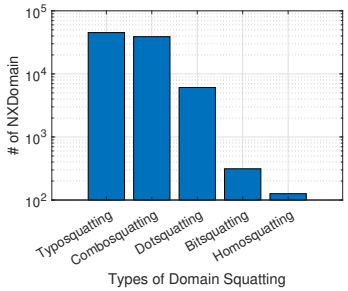

Figure 7: Number of NXDomains for different domain squatting.

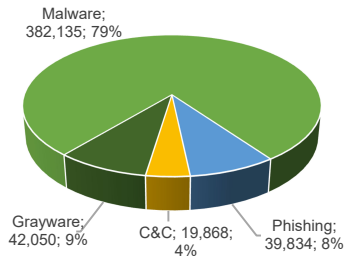

Figure 8: NXDomain distribution of blocklisted domains.

names, as demonstrated by existing research related to typosquatting attacks [34]. This implies that the never-registered NXDomains could originate from users inadvertently making typing mistakes for domain names.

5.2 Malicious NXDomains

Existing studies reveal that domain names have been exploited for various malicious purposes, such as squatting [34, 60, 73], controlling botnets [86, 95], phishing [75], *etc.* Here, we further investigate the 91,545,561 NXdomains that represent expired domains. We aim to explore whether these NXDomains have been used for malicious purposes before expired.

DGA-based NXDomains Botnets extensively rely on Command and Control (C&C) domains to receive commands from their botmaster. However, a static C&C domain can be easily defended by domain blocklists to prevent devices from communicating with the malicious domain. To retain the attack stealthiness and reliability, botmasters employ domain generation algorithms (DGAs) to produce a large number of random domain names. In this way, adversaries can register a small set of DGA domains to control the entire botnet. The rest of the DGA domains result in NXDomain responses from DNS when a bot attempts to establish communication with its botmaster over domain names. Chen *et al.* [41] studied the relationship between DGA and NXDomains in great details. They uncovered 12 new types of DGA malware with more than 76,457 newly discovered DGA domains within 12 days of their DNS data collection.

We employ a commercial DGA identification algorithm to further explore the existence of DGA-based NXDomains in our database. This identification algorithm was developed by our research partner, Palo Alto Networks. More comprehensive information on DGA-detection techniques used by Palo Alto Networks can be found in the latest patent [67]. Such detection algorithms have been integrated into their latest firewall product to effectively detect and block DGA-based attacks [32].

We process all 91 million NXDomains with WHOIS records. Our result shows 2,770,650 potential DGA-based NXDomains, which represent 3% of all expired NXDomains. These DGA-based NXDomains are consistently queried in DNS, which potentially leads to security risksas such NXDomains may be registered and serve as C&C servers for botnet activities.

Squatting NXDomains Domain squatting refers to a group of attacks in which adversaries register a large number of domain names that are extremely similar to the targeted domains. This increases the chance that adversaries successfully bait users to accidentally visit the squatting domains. Adversaries can establish phishing web pages or inject malicious programs into these domains to perform malicious activities. Specifically, recent studies have revealed many types of domain-based squatting attacks, such as typosquatting [34], combosquatting [60], and bitsquatting [73].

In our study, we also investigate expired NXDomains that are used for various types of domain squatting. Within the 91 million expired domains, we discover 90,604 domains that belong to squatting domains. As shown in Figure 7, our commercial identification algorithm finds 45,175 domains used for typosquatting attacks, 38,900 domains for combosquatting attacks, 6,090 are dotsquatting attacks [91], 313 for bitsquatting attacks, and 126 for homosquatting attacks [12], respectively.

The existence of squatting NXDomains proves that users may make various mistakes when accessing domains. It indicates that users are constantly exposed to possible squatting attacks. In addition, it also suggests that these squatting NXDomains have been well distributed over the Internet. People who misinterpret a squatting domain as a benign one can often visit those squatting domains, leading to a large number of NXDomains responses.

Blocklisted Domains A domain blocklist has been used for many years to defend against various attacks. It is a reactive defense mechanism, in which firewalls hold a list of malicious domains that are identified by administrators. The firewall inspects all domain queries in network traffic and, if a malicious domain in the blocklist is identified, the firewall can immediately intercept the domain query. This protects users from accessing malicious domains.

We cross reference expired NXDomains with the domain blocklist maintained by Palo Alto Networks. This blocklist has also been integrated into its commercial firewall products to offer URL filtering control, and the list itself is frequently updated by Palo Alto Networks. Unfortunately, due to the rate limit of querying the blocklist database, we decide to randomly select 20 million expired NXDomains for investigation. In total, we uncover 483,887 NXDomains with historical records of hosting malicious activities. Our research

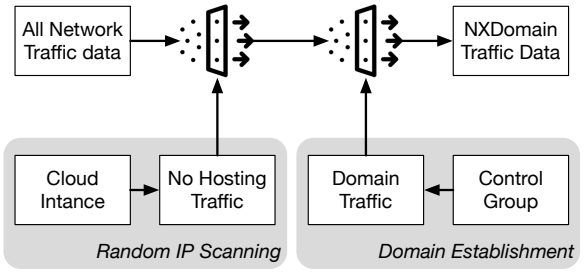

Figure 9: Overview of data filtering mechanism.

partners have also kept track of the malicious behaviors with these domains. As Figure 8 shows, we identify 382,135 NXDomains used to host malware, 42,050 NXDomains containing grayware, 39,834 NXDomains utilized as phishing websites, and 19,868 NXDomains with C&C activities, respectively.

Our observation on blocklisted NXDomains raises an alerting security concern. Although such NXDomains have already been classified as malicious in our blocklist, DNS queries to these NXDomains are still recognized in the passive DNS database, indicating that in practice users are not well protected from accessing such malicious domains.

6 SECURITY IMPLICATIONS OF NXDOMAINS

In this section, we shed light on the security implications of NXDomains using a honeypot system, NXD-Honeypot (Figure 2). We focus primarily on NXDomains that receive a high volume of DNS traffic and have been observed as NXDomains for more than six months.

6.1 Experiment Setup

In Section 4, we analyze Farsight's passive DNS database and identify a large number of NXDomains that attract a significant number of DNS queries. To further explore the motivation for generating those DNS queries so as to reveal potential security implications of NXDomains, we need to collect the real application requests (*e.g.*, HTTP requests) that are supposed to be issued after the DNS queries can be correctly resolved. In doing so, we register a subset of representative NXDomains and host them on public clouds to act as a honeypot to collect their incoming network traffic. We carefully configure our NXD-Honeypot to minimize any potential negative impacts (Appendix A).

Using the domain selection criteria mentioned in Section 3.3, we select a total of 19 NXDomains (shown in Table 1) for investigation, including 8 malicious domains and 11 benign domains. All selected domains receive an average of at least 10,000 DNS queries per month. We register our selected NXDomains and set up a dedicated authoritative DNS server (aDNS) for all these domains. We host our selected domains on two cloud services: Amazon AWS and Google Cloud. This duplication of hosting the domains can effectively help us differentiate and reduce irrelevant network traffic resulting from the hosting platforms.

Our NXD-Honeypot is deployed to record all incoming network traffic received by our domain hosting servers. However, they inevitably collect unwanted traffic introduced from many different sources. In general, undesirable traffic can be introduced by (1) random IP scanning in the cloud servers and (2) domain registration & establishment. For example, the hosting servers on both Amazon AWS and Google Cloud are constantly probed by IP scanners. Some web crawlers may recognize our domains as newly registered domains, so they visit our websites to collect domain data. In addition, services such as the TLS certificate authority (we use *Let's Encrypt* [13] in our experiment) regularly query our websites for certificate validation and verification. All these undesirable traffic contaminates our data collection, and so effective filtering mechanisms should be adopted to distinguish and eliminate unwanted traffic. Simple traffic filtering mechanisms, such as only focusing on requests with a correct Common Name (CN) or hostname, are insufficient in effectively eliminating unwanted data. For example, certain services like Let's Encrypt consistently querying our registered domains with correct hostnames. In this study, we propose our own methodology to filter network traffic data, and an overview of such a mechanism is illustrated in Figure 9.

Traffic from IP Scanning To effectively minimize data contamination resulting from the random IP scanning in cloud services, we adopt a two-step data collection process. Specifically, we first establish cloud instances on Amazon AWS and Google Cloud without hosting any domains. We record the network traffic received by the hosting servers for two months (referred to as *no-hosting-traffic*). Next, we establish our registered domains in the cloud and configure our aDNS server to resolve our domains to the corresponding IP addresses. We record the network traffic of the cloud instances with domain hosting (referred to as *hosting-traffic*). We observe the difference between *hosting-traffic* and *no-hosting-traffic*, and we identify all source IP addresses that appear in *no-hosting-traffic*. By excluding these source IP addresses from our network traffic collection, we can effectively reduce traffic noise introduced by random IP scanning.

Traffic from Domain Establishment Another type of unwanted data is introduced by domain establishment. To reduce network traffic resulting from domain establishment, we conduct an additional experiment in which we register ten domains to serve as a control group. We ensure that these domains do not hold any historical registration records by checking two WHOIS databases, including WhoisXML database [31] used in Section 3.2 and WHOISIQ [27], which have been extensively used in many previous studies [40, 69, 96]. We host these 10 domains on both Amazon AWS and Google Cloud with the same landing page as our selected NXDomains and collect incoming network traffic for two months. Because these domains are newly registered, the network traffic collected by our control group should only attract queries introduced by the domain establishment. Such network traffic is utilized as filtering parameters (*e.g.*, URLs, source IP addresses, and hostname) for the traffic analysis of our selected NXDomains. By excluding similar data in the network traffic of our selected NXDomains, we can effectively reduce the undesirable network data.

6.2 Traffic Categorization

Our data collection lasts 6 months. Figure 10a illustrates the top eight ports that receive the most network queries for our registered domains. Compared to Figure 10b, which shows the network traffic

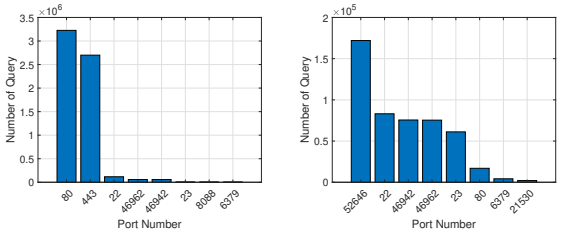

Figure 10: Network traffic received by (a) NXDomains and (b) control groups.

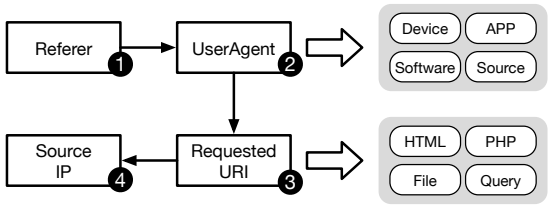

Figure 11: Overview of Traffic Categorization.

received by our control group, it is easy to see that our selected NXDomains receive significantly more queries. This suggests that the background noise of our collected traffic is at a low level. Note that in Figure 10b, port 52646 predominates in network traffic. This port is primarily used by Amazon AWS EC2 to monitor server status, and network traffic on this port is unrelated to our study since it is not generated by our domains. With our data traffic filtering technique, we can effectively exclude this traffic. Consequently, as shown in Figure 10a, port 52646 does not appear in the NXDomain traffic.

Moreover, as shown in Figure 10a, the network traffic received by NXDomains is primarily on ports 80 (HTTP) and 443 (HTTPS). Therefore, in this study, we primarily focus our security implication analysis on these two protocols. While we acknowledge that many types of attacks might occur on other protocols, their contribution to network traffic is significantly less than that of HTTP/HTTPS. Thus, we do not integrate them into our honeypot for monitoring network traffic. Figure 11 presents the HTTP headers that we use to further categorize our received requests. The detailed descriptions are as follows.

❶ *Referer.* The *Referer* field of the HTTP header contains the URL of the referring page through which users visit our domains. If a URL exists in the *Referer* field of a request, it indicates that visitors are redirected to our domains from other web pages. Thus, we categorize such a request as *Referral*.

❷ *User-Agent. User-Agent* field of the HTTP header discloses many aspects about our domain visitors, such as devices, OS, software, and application. First, many web crawlers provide their service names and/or URLs of their official websites in the *User-Agent* header. Therefore, we classify HTTP/HTTPS requests as *Web Crawler* only if they declare themselves as web crawling services in the *User-Agent* field. Second, *User-Agent* also contains information about the visitor's devices, such as PC (Windows/Mac) and cellphones (Android/iOS). Also, we observe many *User-Agent* containing the names of in-app browsers used to visit our domains. Requests with such *User-Agent* information are classified as *User Visit*. Finally, we consider other HTTP/HTTPS requests as *Automated Process*.

❸ *Requested URI.* The *Requested URI* contains a webpage or file resource that the HTTP/HTTPS request is accessing. This field is particularly important because we can obtain various information based on the types of requested files. With the file names and file types, we can have a better understanding of the intention of domain visitors. In particular, we search the National Vulnerability Database provided by NIST [16] to discover existing vulnerabilities associated with such URIs. We consider a URI with sensitive file names if the associated vulnerabilities have a higher than or equal to medium severity [25] (*e.g.*, wp-login.php, changepasswd.php, *etc.*). A URI is considered less sensitive if it does not appear in the database or the associated vulnerabilities have a severity lower than medium. In addition, we also identify URIs that contain query strings (with "?" in the URI), since these additional query parameters can be utilized for malicious activities [33].

❹ *Source IP.* We check the source IP address of the HTTP requests. Particularly, we check the hostname of the source IP by using reverse IP lookup [22]. This information could help us determine the legitimacy of an HTTP request. If the reverse IP lookup results in a hostname that belongs to a popular service, such as Google or Yahoo crawler, we could have high certainty that such requests are benign. If the IP address belongs to a generic cloud provider or cannot be resolved to hostnames, we treat them as generic visits.

6.3 Result

During our 6-month experiment, we gather a total of 5,925,311 HTTP/HTTPS requests to the registered domains. Based on the source IP addresses, request URLs/URIs, referral links, and *User-Agent* of each HTTP/HTTPS, we categorize our collected network traffic into four major groups, including web crawler, automated process, referral, and user visits. We perform a systematic study on each category and uncover security risks based on our network analysis. Table 1 lists detailed statistics of the traffic collected from our selected NXDomains.

Note that not all DNS queries lead to follow-up domain visits. Many software [4, 21] and online services [17, 18] provide DNS scanning capabilities only to resolve domain names to their IP addresses. Therefore, it is anticipated that we record less HTTP/HTTPS requests for some of our selected NXDomains.

Web Crawler Our collected data indicates that a large number of domain visits are generated by web crawlers. We discover two types of web crawlers from our collected traffic: (1) Search Engine and (2) File Grabber. We consider Search Engine as crawlers that query HTML web pages from our registered domains, while File Grabber attempts to access other types of files, such as scripts, pictures, and videos.

Our domains receive a large amount of network traffic from both global and regional search engines. In our experiments, we record a total of 82,942 requests from search engines. In particular, `porno-komiksy.com` and `resheba.online` attract search engine visits of 43,285 (52.2%) and 15,223 (18.4%), respectively, while other domains receive less than 10,000 visits. The search engines also show geographic correlation with our selected NXDomains. For example, `porno-komiksy.com`, which was previously hosted in

Registered Domain	Web Crawler		Automated Process		Referral			User Visit		Others	Total
	Search Engine	*File Grabber*	*Script & Software*	*Malicious Request*	*Search Engine*	*Embedded URL/URI*	*Malicious Link*	*PC & Mobile*	*In-App Browser*		
resheba.online	15,223	105,221	1,866,523	52,263	1,052	655	265	56	20	55,874	2,097,152
1x-sport-bk7.com	4,058	328	1,215,606	725	3,054	143	522	2,952	43	15,428	1,242,859
fanserials.moda	2,536	5,622	996,968	6,225	1,556	4,112	2,189	106	122	4,071	1,023,507
qpclick.com	415	144	365	939,420	10,524	248	115	1,014	22	5,014	957,281
porno-komiksy.com	43,285	105,412	2,952	7,441	2,482	10,244	3,052	25,112	1,825	4,552	206,357
conf-cdn.com	2,653	55,842	10,228	1,699	3,455	2,568	623	2,004	652	11,957	91,681
pro100diplom.com	796	48,868	16,500	9,734	83	261	53	351	108	1,026	77,781
yebeda.org	5,509	25,742	26,564	2,094	1,993	351	314	205	30	4,625	67,367
oboru.work	1,052	49,954	2,651	6,048	50	366	30	4,852	66	501	65,570
kinopack.org	1,205	5,624	6,401	3,255	1,054	213	201	83	304	522	18,862
sfscl.info	421	10,566	2,946	1,098	152	62	97	401	65	957	16,765
ipserv1.net	2,016	7,815	3,297	1,552	336	105	78	105	63	1,192	16,559
cserv11.net	1,487	263	92	65	2,055	263	102	198	105	6,234	10,852
ipserv2.net	323	52	144	1,486	203	96	58	98	86	6,811	9,354
redirectmyquery.com	266	128	62	1,547	269	75	63	188	42	5,022	7,662
adrenali.gq	1,089	357	215	98	52	144	82	1,096	65	3,054	6,252
dns2.name	396	88	105	93	835	35	56	48	51	3,987	5,694
akamai-technology.com	86	85	85	196	65	88	352	620	73	672	2,322
twitter-sup0rt.com	126	185	58	57	107	63	65	118	66	589	1,434
Total	82,942	422,296	4,151,762	1,035,096	29,317	20,092	8,317	39,592	3,808	132,088	5,925,311

Table 1: HTTP/HTTPS traffic received by our registered domains (malicious domains are highlighted).

Russia, receives considerably more requests from a popular Russian search engine named *mail.ru*. Also, `resheba.online` is mainly crawled by Google and Microsoft Bing, due to the fact that this domain is previously registered in the USA.

Another type of web crawler is a file grabber. We collect an even more significant number of requests from file grabbers than those from search engines, with a total of 422,296 requests during the period of our study. In particular, we observe that the *.jpeg*, *.png*, and *.xml* files receive more requests from web crawlers.

An observation that draws our attention is that `conf-cdn.com` receives 53,094 crawling requests from many email providers, and they account for 95.1% of the total requests from file grabbers. Specifically, `conf-cdn.com` receives 30,884 requests from Gmail image crawler, 13,528 from Yahoo mail crawler, and 5,483 from Microsoft email crawler, respectively. This observation suggests that the contents previously hosted on `conf-cdn.com` have been included in many email messages. The fact that email providers repeatedly attempt to obtain these contents may imply that users still access the contents of these emails even though `conf-cdn.com` has become non-existent.

Automated Processes While many web crawlers reveal their identities in *User-Agent*, many other web requests do not disclose any service information in their HTTP headers, nor can we obtain their hostnames through reverse IP lookup. We categorize such behaviors as automated processes. In particular, we observe that many requests disclose the scripting tools and software in their *User-Agent* headers. The scripts and software include Python, Java, curl, wget, etc. We consider these requests in the category of *Script & Software* under automated processes. For requests that do not contain scripts and software information in *User-Agent*, we further extract the requested URIs from their HTTP/HTTPS requests. As mentioned in Section 6.2, we search the National Vulnerability Database provided by NIST [16] to discover the existing vulnerabilities associated with such URIs. If a URI does not contain sensitive file names, we consider the request as *Script & Software*. Otherwise, we classify these requests as *Malicious Request* because they are likely to be a vulnerability probe. We observe that the majority of such malicious requests contain sensitive files, such as `wp-login.php` and `changepassword.php`.

Our traffic analysis shows that, among the four types of network traffic, the automated process accounts for the largest portion of network traffic received by our selected NXDomains. Moreover, we discover that many requests have a repetitive pattern, *i.e.* the same URIs are frequently and periodically accessed. These requests are often issued as streams, meaning that the same URI is requested multiple times by the same IP address.

The total number of web requests from the *Script & Software* category is significantly large, and these requests are concentrated on a small set of domains, *i.e.*, `resheba.online`, `fanserials.moda`, and `1x-sport-bk7.com`. Specifically, `1x-sport-bk7.com` receives a large number of requests from many different addresses. The *User-Agent* of these requests is the same, "`Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2272.118 Safari/537.36`", and they all attempt to access the file called `status.json`. For `resheba.online` and `fanserials.moda`, we observe that many processes attempt to

```
"http.request.url_tree":{
  "http.request.uri.path":"/getTask.php",
  "http.request.uri.query"imei=A-BBBBBB-CCCCCC-D&balance=0&country=us&
        phone=+11112223333&op=Android&mnc=220&mcc=310&
        model=Nexus%205X&os=23",
  "http.request.uri.query_tree":{
    "http.request.query.parameter":"imei=A-BBBBBB-CCCCCC-D",
    "http.request.query.parameter":"balance=0",
    "http.request.query.parameter":"country=us",
    "http.request.query.parameter":"phone=+11112223333",
    "http.request.query.parameter":"op=Android",
    "http.request.query.parameter":"mnc=220",
    "http.request.query.parameter":"mcc=310",
    "http.request.query.parameter":"model=Nexus%205X",
    "http.request.query.parameter":"os=23"
    }
}
```


Figure 12: Example of a malicious request received by *qpclick.com*. We replace the real IMEI and phone number to preserve private information.

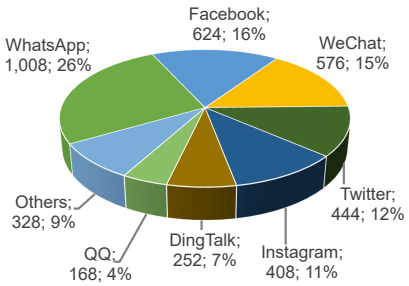

Figure 13: Distribution of In-App Browser used by domain visitors.

access URLs that are related to videos of online courses. A small number of requests also attempt to download the BitTorrent seed of these videos.

Another type of automated process is *Malicious Request*. Among all our registered domains, `qpclick.com` is responsible for 90.8% (939,420 requests) of the malicious HTTP/HTTPS traffic that we record. These malicious requests also account for 98.1% of the network traffic received by `qpclick.com`, and they request the same file called `getTask.php`. The URLs in HTTP/HTTPS requests reveal additional information about the visit. Figure 12 shows the URL structure of the requests. Specifically, it includes an IMEI number that is a unique identification number for each cellphone, a cellphone number, country code, cellphone model, and other information, posing a serious privacy leakage to visitors.

Referral We investigate the types of *referral* by applying FortiGuard Web Filter [9] to the redirecting domains. The results indicate three categories of referral visits, including search engines, embedded URLs/URIs, and malicious links. The referer pages that fall into the category of search engines could represent that users are accessing our domains through web search. We discover that many visitors browse our website through search engine referrals, which account for 29,317 requests.

Furthermore, we investigate the redirecting URLs of our collected HTTP/HTTPS requests that do not belong to search engines. We obtain the redirecting web page using `cURL` and check if the URLs associated with our registered domains are embedded in the websites. If the URLs appear in the redirecting web page, we consider such a request in the category of *Embedded URL/URI*. A user can click the embedded URLs in the referral webpage to be referred to our domains or the browser may automatically initiate a request to our domains when the referral webpage is accessed. We confirm that all the referral websites are benign and that most of the websites fall into the category of forums and blogs. On the other hand, if the referral URLs are not valid or we cannot find our domain URLs in the redirecting webpages, we consider them as *Malicious Link*. We have also identified that numerous redirecting webpages either are invalid or do not contain our domain URLs. This observation implies that such domain requests are intentionally crafted with false information. Among the 8,317 requests in *Malicious Link*, we find 1,524 requests with valid referral URLs, but these URLs do not contain any hyperlinks that point to our registered domains.

User Visits Despite the fact that our selected domains have been in non-existent status for at least six months, many users still visit our domains frequently. Particularly, as shown in Table 1, `porno-komiksy.com` receives the most user visits among all domains we established, with a total of 25,112 requests. From the *User-Agent* header of these requests, we further reveal additional information about the users' devices. For example, we observe that the majority of visitors access `porno-komiksy.com` using Windows/MacOS computers and cellphones made by Apple, Huawei, XiaoMi, and Samsung.

In addition, the *User-Agent* also reveals the browser information, and interestingly, we find that many requests to `porno-komiksy.com` are sent by In-App browsers. Figure 13 shows the distribution of those In-App browsers. We can see that many users attempt to access our domains through short-messaging services (*e.g.*, WhatsApp and WeChat) and social media (*e.g.*, Facebook and Twitter). This observation may suggest that our registered domains are widely distributed on these platforms, which results in users frequently visiting our domains, despite the fact that they have been in non-existent status for more than six months.

6.4 Security Analysis

Registering an NXDomain can automatically inherit all network traffic and residual trust if the domain was previously used by others. This can be exploited by adversaries for malicious purposes, especially if the adversary becomes the owner of frequently queried NXDomains. In this section, we analyze the network traffic that we acquire from our selected NXDomains to uncover potential security implications.

Botnet Takeover As shown in Table 1, `qpclick.com` receives a total of 939,420 requests during our experiment. The requested URI follows a specific pattern shown in Figure 12. All malicious requests have the same *User-Agent* of "`Apache-HttpClient/UNAVAILABLE (java 1.4)`". The URI field of these malicious requests leaks sensitive information about the victims, including their cellphone's IMEIs, models, phone numbers, and their residing countries. In particular, we observe a total of 40 different cellphone models from the malicious URIs, with the most popular cellphone model being Nexus 205X (4,139, 55.9%) and Nexus 205 (3,131, 42.3%). The rest

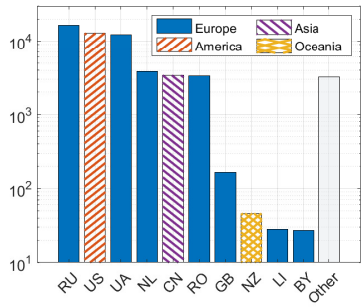

Figure 14: *qpclick.com* **cellphone country code.**

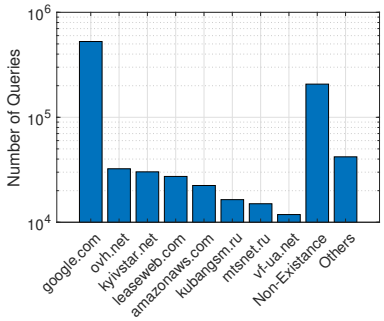

Figure 15: *qpclick.com* **hostname overview.**

1.8% come from 38 different cellphone models, including Samsung, LG, Vivo, HTC, HUAWEI, XiaoMi, and Motorola.

Interestingly, `qpclick.com` has already been reported in 2013 about its malicious behaviors of possible botnet hosting [28]. The report uncovers similar findings as those observed by us, suggesting that such malicious activities have been active for nearly 10 years. The disclosure also suggests that the malware running on the victim's side is embedded in a Russian-based browser. Thus, the malicious activities only aim to target Russian-speaking users. This is different from our observation. Figure 14 illustrates the distribution of the country codes of 55,829 cellphone numbers we collect. It is obvious that the victims of this malware are now spread across many countries besides Russian-speaking countries, such as the USA, Uruguay, the Netherlands, and China.

Although cellphone numbers suggest a global botnet activity, the actual IP addresses that initiate these malicious requests are not widely spread. Figure 15 shows the hostname distribution for each IP address that sends malicious requests to `qpclick.com`. This figure suggests that such a botnet infrastructure utilizes cloud services to route malicious requests to `qpclick.com`. Especially, 527,226 of the malicious requests have the source IP addresses hosted by google-proxy server, which accounts for 56.1% of the total requests. For ethical considerations, we have reported this issue to Google.

Malicious File Injection Our network traffic data indicates that users attempt to acquire various web pages that previously existed in NXDomains. Such web pages contain executable or scripting programs, such as JavaScript and PHP files. Adversaries can intentionally inject malicious programs so that victims who access these web pages could be infected. Another potential method for malicious file injection exploits automated processes. For instance, `1x-sport-bk7.com` receives the network download requests for *status.json*, while other domains are frequently requested by automated processes such as `cURL` and `Wget` to download scripts and executable programs. Adversaries can feed automated processes with malicious programs to compromise victims' systems. Our investigation also observes many traffic from email crawlers that attempt to obtain certain images and files from our domains. Such a behavior can be abused by attackers to conduct adversarial activities by injecting malicious images and files. This threatens the security of the victims' email systems, as Hu *et al.* [54] has explored this type of attack in more detail.

Residual Trust Exploitation The security risk of residual trust has been extensively studied in the past. We observe the same problem in our experiment. Adversaries can register frequently queried NXDomains to host malicious content. In particular, the use of In-App browsers, while not accounting for a large number of network traffic, could also suggest potential security risks. Adversaries could register these NXDomains to bait potential victims.

7 LIMITATIONS & FUTURE WORK

Database Coverage Our NXDomain scale and origin measurements are based on the Farsight passive DNS database, from which we discover a significant number of NXDomains between 2014 and 2022. Such a database has been widely used in previous studies. Khalil *et al.* [59] and Nabeel *et al.* [71] analyzed the Farsight passive DNS database to discover malicious domains. The same database was also utilized by Alrwais *et al.* [36] to conduct measurement studies on BulletProof Hosting (BPH) services.

To the best of our knowledge, the Farsight Passive DNS dataset is one of the largest passive DNS datasets that are available to researchers. Unfortunately, due to security reasons, Farsight protects all information about its data contributors from public access. Because of this, we are unable to validate the bias in the data collected in the Farsight database. One example of biased data could be due to the geolocation of the data contributors. If more contributors are located in certain regions, the NXDomains in the database could reflect more on these regions. In addition, if the contributors are mainly from large enterprises, we could miss the NXDomains accessed by residential IPs.

To reduce such a limitation, we plan to expand our measurement in the future by leveraging additional DNS databases. In addition to the Farsight DNS database, other popular DNS databases CIRCL.lu [3], DNSIQ [19], and Mnemonic [14] can also be analyzed to further enhance our experiments. Moreover, we can obtain more data from regional passive DNS services, such as the 114DNS database utilized by Xie *et al.* [94] which target primarily in the greater China area. Our future efforts will involve acquiring access to additional DNS databases, which enables us to conduct a more comprehensive analysis of NXDomains on a global scale.

DNS Hijacking As demonstrated in previous studies [43, 92], NXDomain responses could be hijacked by ISPs to generate extra profits. When a user attempts to query an IP address of an NXDomain,

ISPs can intercept the DNS query and return an IP address of an advertising domain, instead of an "NXDomain" response. Given the large number of NXDomains, ISPs could gain significant financial benefits by hijacking "NXDomain" responses.

Under the DNS hijacking attack, "NXDomain" responses are replaced with IP addresses of advertising domains. As a result, such a behavior prevents us from observing the "NXDomain" responses, making these NXDomains invisible in our passive DNS database. However, NXDomain hijacking rarely occurs in the current DNS ecosystem. According to [43], only 4.8% NXDomain responses are hijacked in the wild. Furthermore, our study focuses particularly on investigating the scale, origin, and security implications of high-traffic NXDomains. Considering that it is quite rare that all ISPs from which the clients come will hijack NXDomain responses, those frequently queried NXDomains are highly likely captured in the Farsight passive DNS database.

Experiments To investigate the security implications of NXDomains, we register 19 NXDomains and deploy NXD-honeypot to collect all inbound network traffic. These domains represent three types of NXDomain origins: benign domains, malicious domains, and squatting domains. In total, we collect 5,925,311 HTTP/HTTPS queries during the 6 months of our experiments. In the future, we aim to expand our investigation by registering additional NXDomains. We will also explore security implications of protocols other than HTTP/HTTPS.

Furthermore, we plan to enhance our NXD-honeypot by implementing the capability to interact with domain visitors. This will provide us with additional information in order to comprehensively understand the purpose of their visits. We will also work with enterprises specialized in firewall and network traffic analysis. We attempt to sinkhole NXDomain traffic to dedicated analysis servers, so we can identify security problems directly based on DNS traffic analysis.

8 RELATED WORK

8.1 DNS and NXDomain

DNS serves as one critical component in today's Internet infrastructure to translate between domain names and IP addresses. For decades, researchers have devoted significant efforts to DNS security. Previous works have investigated different malicious attacks on DNS, including DDoS [72, 74] and cache poisoning [45, 46, 53, 56, 68, 97], as well as misconfigurations [78, 87].

NXDomains, as one of the DNS error responses, have also been extensively studied. Jung *et al.* [58] and Plonka *et al.* [81] discovered that 10%-42% of DNS responses are NXDomain responses. Schüppen *et al.* [83] and Antonakakis *et al.* [37] proposed methods to identify DGA malware using NXDomains. Kreibich *et al.* [61] studied the NXDomain wildcarding. Weaver *et al.* [92] and Chung *et al.* [43] revealed that many ISPs take advantage of NXDomain wildcarding and redirect users to ad servers to generate profits.

Our study complements existing research on DNS and NXDomains. Utilizing a passive DNS database, we conduct a comprehensive study to uncover the long-lasting existence of some NXDomains in DNS queries. We demonstrate that adversaries could misuse these high-traffic NXDomains to effectively take over botnet, inject malicious files, and exploit the residual trust of expired domains.

8.2 Expired Domain

Expired domains, which are one of the main causes of NXDomains, have also been studied before. Lauinger *et al.* [62] highlighted the significant demand for expired domains, and hinted at highly competitive re-registrations. Lauinger *et al.* [63] showed that a large number of domains are re-registered immediately after their expiration date/time. Moore *et al.* [70] and Lever *et al.* [64] found that the residual trust of expired domains can pose serious security problems. Expired domains can also be used as an attack vector. For example, Hao *et al.* [51] discovered that many expired domains are misused as spammer domains; Liu *et al.* [66] revealed that dangling DNS records of expired domains can be easily manipulated by adversaries for domain hijacking attacks; and Vissers *et al.* [88] demonstrated that the nameservers could be compromised to hijack domain names.

Our work differs from previous studies as we investigate domains that have been in non-existent status for an extended period of time. We discover that a large number of expired domains receive millions of queries despite the fact that DNS returns NXDomain responses to their inquirers. These NXDomains expose severe security risks and are attractive targets for exploitation by adversaries.

9 CONCLUSION

In this paper, we present an in-depth study on the scale, origin, and security implications of NXDomains. First, we conduct a comprehensive analysis based on a passive DNS database. We reveal a large quantity of NXDomains in the database, and these NXDomains account for even larger number of DNS queries in the current DNS infrastructure. Next, we investigate the NXDomains found in the database. We discover three types of origins of these NXDomains: (1) expired domains, (2) never-existed domains, and (3) malicious domains. In particular, many NXDomains with high DNS traffic fall into the category of expired domains and malicious domains. Finally, we establish 19 NXDomains and deploy NXD-honeypots for data collection. We reveal many security risks based on the network traffic analysis of our registered NXDomains, including botnet takeover, malicious file injection, and residual trust exploitation.

ACKNOWLEDGEMENTS

We thank our shepherd, Gautam Akiwate, and the anonymous reviewers for their insightful comments, which help to improve the quality of this paper. This work is supported in part by NSF grant CNS-2317829.

REFERENCES

[1] 101domain. https://www.101domain.com/.
[2] CatchTiger. https://www.catchtiger.com/.
[3] CIRCL.lu. https://www.circl.lu/.
[4] Dig - DNS lookup utility. https://man.openbsd.org/dig.1.
[5] Domain Name Registration's Statistics. https://domainnamestat.com/statistics/overview.
[6] DropCatch. https://www.dropcatch.com/.
[7] Expired Registration Recovery Policy. https://newgtlds.icann.org/en/about/program.

[8] Farsight Security Information Exchange (SIE). https://docs.farsightsecurity.com/sie/sie-channel-guide/.
[9] FortiGuard Web Filter Lookup. https://www.fortiguard.com/webfilter.
[10] GoDaddy. https://www.godaddy.com/.
[11] ICANN. https://www.icann.org/.
[12] IDN homograph attack. https://en.wikipedia.org/wiki/IDN_homograph_attack.
[13] Let's Encrypt. https://letsencrypt.org/.
[14] Mnemonic. https://passivedns.mnemonic.no/.
[15] Namecheap. https://www.namecheap.com/.
[16] National Vulnerability Database. https://nvd.nist.gov/vuln/search.
[17] NMAP DNS Resolution. https://nmap.org/book/host-discovery-dns.html.
[18] NsLookup.io. https://www.nslookup.io/.
[19] PassiveTotal DNSIQ. https://help.passivetotal.org/passive_dns_sources.html.
[20] pool.com. https://pool.com/.
[21] Public DNS scans. https://dnsspy.io/scan.
[22] Reverse DNS Lookup. https://whatismyipaddress.com/ip-hostname.
[23] The Domain Name Industry Brief. https://www.verisign.com/en_US/domain-names/dnib/index.xhtml.
[24] The top four DNS response codes and what they mean. https://bluecatnetworks.com/blog/the-top-four-dns-response-codes-and-what-they-mean/.
[25] Vulnerability Metrics. https://nvd.nist.gov/vuln-metrics/cvss#.
[26] What you can learn from an NXDOMAIN response. https://bluecatnetworks.com/blog/what-you-can-learn-from-an-nxdomain-response/.
[27] WHOISIQ API. https://api.riskiq.net/api/whois/.
[28] Discovered a new malware for Android which subscribes its victims to paid services via SMS! https://www.tgsoft.it/english/news_archivio_eng.asp?id=565, 2013.
[29] Farsight dnsdb 2.0. https://www.farsightsecurity.com/solutions/dnsdb/, 2017.
[30] Security information exchange (sie) nx domains. https://docs.farsightsecurity.com/sie/sie-221-nxdomains/, 2022.
[31] Whoisxml. https://www.whoisxmlapi.com, 2022.
[32] Palo alto networks dns security. https://docs.paloaltonetworks.com/dns-security, 2023.
[33] Dilara Acarali, Muttukrishnan Rajarajan, Nikos Komninos, and Ian Herwono. Survey of Approaches and Features for the Identification of HTTP-Based Botnet Traffic. *Journal of Network and Computer Applications*, 2016.
[34] Pieter Agten, Wouter Joosen, Frank Piessens, and Nick Nikiforakis. Seven Months' Worth of Mistakes: A Longitudinal Study of Typosquatting Abuse. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2015.
[35] Muneeb Ali, Jude Nelson, Ryan Shea, and Michael J Freedman. Blockstack: A Global Naming and Storage System Secured by Blockchains. In *Proceedings of the USENIX Annual Technical Conference (ATC)*, 2016.
[36] Sumayah Alrwais, Xiaojing Liao, Xianghang Mi, Peng Wang, XiaoFeng Wang, Feng Qian, Raheem Beyah, and Damon McCoy. Under the Shadow of Sunshine: Understanding and Detecting Bulletproof Hosting on Legitimate Service Provider Networks. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2017.
[37] Manos Antonakakis, Roberto Perdisci, Yacin Nadji, Nikolaos Vasiloglou, Saeed Abu-Nimeh, Wenke Lee, and David Dagon. From Throw-Away Traffic to Bots: Detecting the Rise of DGA-Based Malware. In *Proceedings of the USENIX Security Symposium*, 2012.
[38] Giuseppe Ateniese and Stefan Mangard. A New Approach to DNS Security (DNSSEC). In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2001.
[39] Leyla Bilge, Engin Kirda, Christopher Kruegel, and Marco Balduzzi. EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2011.
[40] Marzieh Bitaab, Haehyun Cho, Adam Oest, Penghui Zhang, Zhibo Sun, Rana Pourmohamad, Doowon Kim, Tiffany Bao, Ruoyu Wang, Yan Shoshitaishvili, et al. Scam pandemic: How attackers exploit public fear through phishing. In *APWG Symposium on Electronic Crime Research (eCrime)*, 2020.
[41] Yizheng Chen, Yacin Nadji, Athanasios Kountouras, Fabian Monrose, Roberto Perdisci, Manos Antonakakis, and Nikolaos Vasiloglou. Practical Attacks Against Graph-based Clustering. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2017.
[42] Zhanhao Chen and Janos Szurdi. Cybersquatting: Attackers Mimicking Domains of Major Brands Including Facebook, Apple, Amazon and Netflix to Scam Consumers. https://unit42.paloaltonetworks.com/cybersquatting/.
[43] Taejoong Chung, David Choffnes, and Alan Mislove. Tunneling for Transparency: A Large-Scale Analysis of End-to-End Violations in the Internet. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, 2016.
[44] Taejoong Chung, Roland van Rijswijk-Deij, Balakrishnan Chandrasekaran, David Choffnes, Dave Levin, Bruce M Maggs, Alan Mislove, and Christo Wilson. A Longitudinal, End-to-End View of the DNSSEC Ecosystem. In *Proceedings of the USENIX Security Symposium*, 2017.
[45] Tianxiang Dai, Philipp Jeitner, Haya Shulman, and Michael Waidner. From IP to Transport and Beyond: Cross-Layer Attacks Against Applications. In *Proceedings of the ACM SIGCOMM Conference*, 2021.
[46] Tianxiang Dai, Philipp Jeitner, Haya Shulman, and Michael Waidner. The Hijackers Guide To The Galaxy: Off-Path Taking Over Internet Resources. In *Proceedings of the USENIX Security Symposium*, 2021.
[47] Peter B Danzig, Katia Obraczka, and Anant Kumar. An Analysis of Wide-Area Name Server Traffic: A study of the Internet Domain Name System. In *Proceedings of the ACM SIGCOMM Conference*, 1992.
[48] Hongyu Gao, Vinod Yegneswaran, Yan Chen, Phillip Porras, Shalini Ghosh, Jian Jiang, and Haixin Duan. An Empirical Reexamination of Global DNS Behavior. In *Proceedings of the ACM SIGCOMM Conference*, 2013.
[49] Shuai Hao and Haining Wang. Exploring Domain Name Based Features on the Effectiveness of DNS Caching. *ACM SIGCOMM Computer Communication Review*, 2017.
[50] Shuai Hao, Haining Wang, Angelos Stavrou, and Evgenia Smirni. On the DNS Deployment of Modern Web Services. In *Proceedings of the 23rd IEEE International Conference on Network Protocols (ICNP)*, 2015.
[51] Shuang Hao, Matthew Thomas, Vern Paxson, Nick Feamster, Christian Kreibich, Chris Grier, and Scott Hollenbeck. Understanding the Domain Registration Behavior of Spammers. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, 2013.
[52] Rebekah Houser, Shuai Hao, Chase Cotton, and Haining Wang. A Comprehensive, Longitudinal Study of Government DNS Deployment at Global Scale. In *Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2022.
[53] Rebekah Houser, Shuai Hao, Zhou Li, Daiping Liu, Chase Cotton, and Haining Wang. A Comprehensive Measurement-based Investigation of DNS Hijacking. In *Proceedings of the International Symposium on Reliable Distributed Systems (SRDS)*, 2021.
[54] Hang Hu, Peng Peng, and Gang Wang. Characterizing Pixel Tracking through the Lens of Disposable Email Services. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2019.
[55] Luca Invernizzi, Paolo Milani Comparetti, Stefano Benvenuti, Christopher Kruegel, Marco Cova, and Giovanni Vigna. EVILSEED: A Guided Approach to Finding Malicious Web Pages. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2012.
[56] Philipp Jeitner and Haya Shulman. Injection Attacks Reloaded: Tunnelling Malicious Payloads over DNS. In *Proceedings of the USENIX Security Symposium*, 2021.
[57] Lin Jin, Shuai Hao, Huang Yan, Haining Wang, and Chase Cotton. DNSonChain: Delegating Privacy-Preserved DNS Resolution to Blockchain. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP)*, 2021.
[58] Jaeyeon Jung, Emil Sit, Hari Balakrishnan, and Robert Morris. Dns performance and the effectiveness of caching. In *IEEE/ACM Transactions on Networking*, 2002.
[59] Issa Khalil, Ting Yu, and Bei Guan. Discovering Malicious Domains through Passive DNS Data Graph Analysis. In *Proceedings of the ACM ASIA Conference on Computer and Communications Security (ASIACCS)*, 2016.
[60] Panagiotis Kintis, Najmeh Miramirkhani, Charles Lever, Yizheng Chen, Rosa Romero-Gómez, Nikolaos Pitropakis, Nick Nikiforakis, and Manos Antonakakis. Hiding in Plain Sight: A Longitudinal Study of Combosquatting Abuse. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2017.
[61] Christian Kreibich, Nicholas Weaver, Boris Nechaev, and Vern Paxson. Netalyzr: Illuminating The Edge Network. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, 2010.
[62] Tobias Lauinger, Abdelberi Chaabane, Ahmet Salih Buyukkayhan, Kaan Onarlioglu, and William Robertson. Game of Registrars: An Empirical Analysis of Post-Expiration Domain Name Takeovers. In *Proceedings of the USENIX Security Symposium*, 2017.
[63] Tobias Lauinger, Abdelberi Chaabane, Ahmet Salih Buyukkayhan, Kaan Onarlioglu, and William Robertson. Game of Registrars: An Empirical Analysis of Post-Expiration Domain Name Takeovers. In *USENIX Security Symposium*, 2017.
[64] Chaz Lever, Robert Walls, Yacin Nadji, David Dagon, Patrick McDaniel, and Manos Antonakakis. Domain-Z: 28 Registrations Later Measuring the Exploitation of Residual Trust in Domains. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2016.
[65] Wilson Lian, Eric Rescorla, Hovav Shacham, and Stefan Savage. Measuring the Practical Impact of DNSSEC Deployment. In *Proceedings of the USENIX Security Symposium*, 2013.
[66] Daiping Liu, Shuai Hao, and Haining Wang. All Your DNS Records Point to Us: Understanding the Security Threats of Dangling DNS Records. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2016.
[67] Daiping Liu, Martin Walter, Ben Hua, Suquan Li, Fan Fei, Seokkyung Chung, Jun Wang, and Wei Xu. In-line detection of algorithmically generated domains, 2023. US Patent 11,729,134.
[68] Keyu Man, Xin'an Zhou, and Zhiyun Qian. DNS Cache Poisoning Attack: Resurrections with Side Channels. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2021.
[69] Ariana Mirian, Christopher Thompson, Stefan Savage, Geoffrey M Voelker, and Adrienne Porter Felt. HTTPS Adoption in the Longtail. 2018.

[70] Tyler Moore and Richard Clayton. The Ghosts of Banking Past: Empirical Analysis of Closed Bank Websites. In *Proceedings of the International Conference on Financial Cryptography and Data Security*, 2014.
[71] Mohamed Nabeel, Issa M Khalil, Bei Guan, and Ting Yu. Following Passive DNS Traces to Detect Stealthy Malicious Domains Via Graph Inference. *ACM Transactions on Privacy and Security*, 2020.
[72] Marcin Nawrocki, Mattijs Jonker, Thomas C Schmidt, and Matthias Wählisch. The Far Side of DNS Amplification: Tracing the DDoS Attack Ecosystem from the Internet Core. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, 2021.
[73] Nick Nikiforakis, Steven Van Acker, Wannes Meert, Lieven Desmet, Frank Piessens, and Wouter Joosen. Bitsquatting: Exploiting Bit-flips for Fun, or Profit? In *Proceedings of the Web Conference (WWW)*, 2013.
[74] Yevheniya Nosyk, Maciej Korczyński, and Andrzej Duda. Routing Loops as Mega Amplifiers for DNS-Based DDoS Attacks. In *Proceedings of the International Conference on Passive and Active Network Measurement (PAM)*, 2022.
[75] Adam Oest, Penghui Zhang, Brad Wardman, Eric Nunes, Jakub Burgis, Ali Zand, Kurt Thomas, Adam Doupé, and Gail-Joon Ahn. Sunrise to Sunset: Analyzing the End-to-end Life Cycle and Effectiveness of Phishing Attacks at Scale. In *Proceedings of the USENIX Security Symposium*, 2020.
[76] John S Otto, Mario A Sánchez, John P Rula, and Fabián E Bustamante. Content Delivery and the Natural Evolution of DNS: Remote DNS Trends, Performance Issues and Alternative Solutions. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, 2012.
[77] Jeffrey Pang, James Hendricks, Aditya Akella, Roberto De Prisco, Bruce Maggs, and Srinivasan Seshan. Availability, Usage, and Deployment Characteristics of the Domain Name System. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, 2004.
[78] Vasileios Pappas, Zhiguo Xu, Songwu Lu, Daniel Massey, Andreas Terzis, and Lixia Zhang. Impact of Configuration Errors on DNS Robustness. In *Proceedings of the ACM SIGCOMM Conference*, 2004.
[79] KyoungSoo Park, Vivek S Pai, Larry L Peterson, and Zhe Wang. CoDNS: Improving DNS Performance and Reliability via Cooperative Lookups. In *Proceedings of the USENIX Conference on Operating Systems Design and Implementation (OSDI)*, 2004.
[80] Daniel Plohmann, Khaled Yakdan, Michael Klatt, Johannes Bader, and Elmar Gerhards-Padilla. A Comprehensive Measurement Study of Domain Generating Malware. In *Proceedings of the USENIX Security Symposium*, 2016.
[81] David Plonka and Paul Barford. Context-aware Clustering of DNS Query Traffic. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, 2008.
[82] Paul Schmitt, Anne Edmundson, and Nick Feamster. Oblivious DNS: Practical Privacy for DNS Queries. *arXiv preprint arXiv:1806.00276*, 2018.
[83] Samuel Schüppen, Dominik Teubert, Patrick Herrmann, and Ulrike Meyer. FANCI: Feature-based Automated NXDomain Classification and Intelligence. In *Proceedings of the USENIX Security Symposium*, 2018.
[84] Haya Shulman and Michael Waidner. One Key to Sign Them All Considered Vulnerable: Evaluation of DNSSEC in the Internet. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2017.
[85] Johnny So, Najmeh Miramirkhani, Michael Ferdman, and Nick Nikiforakis. Domains Do Change Their Spots: Quantifying Potential Abuse of Residual Trust. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2022.
[86] Brett Stone-Gross, Marco Cova, Lorenzo Cavallaro, Bob Gilbert, Martin Szydlowski, Richard Kemmerer, Christopher Kruegel, and Giovanni Vigna. Your Botnet is My Botnet: Analysis of a Botnet Takeover. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2009.
[87] Niels LM van Adrichem, Norbert Blenn, Antonio Reyes Lúa, Xin Wang, Muhammad Wasif, Ficky Fatturrahman, and Fernando A Kuipers. A Measurement Study of DNSSEC Misconfigurations. *Security Informatics*, 2015.
[88] Thomas Vissers, Timothy Barron, Tom Van Goethem, Wouter Joosen, and Nick Nikiforakis. The Wolf of Name Street: Hijacking Domains Through Their Nameservers. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2017.
[89] Michael Walfish, Hari Balakrishnan, and Scott Shenker. Untangling the Web from DNS. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2004.
[90] Yi-Min Wang, Doug Beck, Xuxian Jiang, Roussi Roussev, Chad Verbowski, Shuo Chen, and Sam King. Automated Web Patrol with Strider HoneyMonkeys: Finding Web Sites That Exploit Browser Vulnerabilities. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2006.
[91] Yi-Min Wang, Doug Beck, Jeffrey Wang, Chad Verbowski, and Brad Daniels. Strider typo-patrol: Discovery and analysis of systematic typo-squatting. *SRUTI*, 2006.
[92] Nicholas Weaver, Christian Kreibich, and Vern Paxson. Redirecting DNS for Ads and Profit. In *USENIX Workshop on Free and Open Communications on the Internet*, 2011.
[93] Pengcheng Xia, Haoyu Wang, Zhou Yu, Xinyu Liu, Xiapu Luo, and Guoai Xu. Ethereum Name Service: the Good, the Bad, and the Ugly. arXiv Preprint, https://arxiv.org/abs/2104.05185, 2021.
[94] Qinge Xie, Shujun Tang, Xiaofeng Zheng, Qingran Lin, Baojun Liu, Haixin Duan, and Frank Li. Building an Open, Robust, and Stable Voting-Based Domain Top List. In *Proceedings of the USENIX Security Symposium*, 2022.
[95] Kui Xu, Patrick Butler, Sudip Saha, and Danfeng (Daphne) Yao. DNS for Massive-Scale Command and Control. *IEEE Transactions on Dependable and Secure Computing*, 2013.
[96] Eric Zeng, Frank Li, Emily Stark, Adrienne Porter Felt, and Parisa Tabriz. Fixing HTTPS Misconfigurations at Scale: An Experiment with Security Notifications. 2019.
[97] Xiaofeng Zheng, Chaoyi Lu, Jian Peng, Qiushi Yang, Dongjie Zhou, Baojun Liu, Keyu Man, Shuang Hao, Haixin Duan, and Zhiyun Qian. Poison Over Troubled Forwarders: A Cache Poisoning Attack Targeting DNS Forwarding Devices. In *Proceedings of the USENIX Security Symposium*, 2020.

APPENDIX

A ETHICS

Our experimental methodology is inspired by many previous studies with similar data collection and analysis techniques. For example, So *et al.* [85] explored the abuse of residue trust using honeypots. Similarly, Wang *et al.* [90] utilized honeypots to identify malicious websites. Stone-Gross *et al.* [86] registered DGA-based domains to investigate the Torpig botnet. In our study, we carefully design our methodology to reduce potential ethical concerns.

We gain access to the Farsight passive DNS database through our research partner, who securely stores the entire database in its BigQuery server. Our NXDomain measurement and analysis on the scale and origin of NXDomains are conducted directly on the BigQuery server. While we gain access to the Farsight Database from our research partner, we explain our methodologies and experiments to Farsight, informing them about the usage of their NXDomain database in this work.

To investigate the security implications of NXDomains, we register 19 high-traffic NXDomains and host them on Amazon AWS and Google Cloud. Our selected NXDomains must remain in non-existent status for at least 6 months, which implies that the public has less interest in registering such domains, reducing the chance that other users will not be able to register these domains due to our experiments. Moreover, our registered domains receive network traffic that would be exploited if these domains were registered by adversaries. Our experiments can potentially protect these domains from being exploited and domain visitors from being attacked.

We deploy our NXD-Honeypot to collect all incoming traffic of our registered domains. We carefully configure our NXD-Honeypot to minimize any negative impacts on domain visitors. First, our NXD-Honeypot passively collects communication packets sent to our hosting server only. We do not intend to initiate any forms of interaction with the domain visitors. Second, NXD-Honeypot is also utilized as a web server for our registered domains. The web server displays a landing page with detailed information about our experiments. We also post our email addresses on the landing page so that domain visitors can contact us in case they have any concerns about our data collection. Throughout our experiments, we have not received any complaints or communications for our registered domains.

Our NXD-Honeypots unavoidably collect Personally Identifiable Information (PII) transmitted to our registered domains. As illustrated in Figure 12, our honeypots capture and record sensitive data, such as victims' phone numbers, phone models, and country

codes. We take data privacy seriously. To ensure restricted access to this sensitive information, our data is securely housed on our lab server, which is located behind the firewall of our university. Additionally, our lab server room is kept locked when not in use, with access granted only for routine maintenance purposes. Our experimental data can only be made available upon request, subject to rigorous review. Any sensitive information that was collected has been thoroughly anonymized, and we ensure that only the anonymized data can be released. Furthermore, all data containing PIIs has been permanently deleted prior to the publication of this paper.

Furthermore, we will continue monitoring and renew all 19 domains for further investigation. This ensures that the security vulnerabilities exposed by these NXDomains cannot be exploited by adversaries for malicious purposes. Regarding disclosure and mitigation, our findings have been recognized by our research partners. We are actively collaborating in developing countermeasures and integrating them into existing security solutions and products.