Evaluating design decisions and bias resistance for passive DNS-based domain rankings

Victor Le Pochat *, Simon Fernandez §, Samaneh Tajalizadehkhoob ‖,
Lieven Desmet *, Andrzej Duda §, Wouter Joosen *, Maciej Korczyński §
* DistriNet, KU Leuven § Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG ‖ ICANN

***Abstract*—'Top sites' rankings of the most popular domains are a core resource for the large-scale measurements that are crucial in Web and Internet research. Recent rankings evolved towards using passive DNS traffic data, but this data's suitability for measuring website popularity is poorly understood. In this paper, we holistically evaluate how design decisions influence the composition and desired properties of passive DNS-based domain rankings. We isolate the effects of these decisions by generating a ranking from the ground up using aggregated "post-recursor" passive DNS data. We confirm that decisions for bucketing and aggregation produce more stable rankings, and see that corrections for resolver caching, CDNs, and service classification strongly impact suitability for Web measurements. We further analyze the resistance of rankings to inadvertent biases or even active manipulation, and find that design choices such as TTL weighting severely impact robustness. Our goal is to give transparent insight into the process of using passive DNS data for domain rankings, as a framework for the research community to understand how to develop future rankings that address their needs.**

***Index Terms*—algorithm design and analysis, Domain Name System, manipulation resistance, passive DNS, popularity ranking**

I. Introduction

Large-scale Web and Internet measurements play a crucial role in enabling research into how the modern Internet works. Researchers have emphasized that such measurements should be accurate, reproducible, and representative [1–5]. A core element in conducting these measurements according to those expectations concerns the selection of a (representative) sample of websites that form the subject of the study. Most often, researchers use 'top sites' rankings of the most popular domains for this purpose. These rankings are a valuable and necessary tool that the research community relies on, to the count of hundreds of studies [6–9].

The significant impact of ranking properties on research results across all areas of Web and Internet measurement [7] warrants a thorough understanding of these rankings. The academic research community has recently raised awareness on the issues surrounding top sites rankings that threaten their reliability: low agreement [6, 10], stability, vulnerability to manipulation and transparency [7, 8]. The decisions taken when designing how top sites rankings are constructed heavily affect the properties of those rankings. For example, a shorter period of time over which data is aggregated to rank domains makes rankings much more volatile and therefore worse for longitudinal studies that need stable data [7, 8]. The need to carefully assess design decisions is reinforced by a recent shift within rankings towards reliance on passive DNS traffic. Early top sites rankings were mainly based on Web traffic data, but due to privacy challenges around data acquisition, these Web rankings have steadily become unavailable. Several providers have therefore developed new rankings based on DNS resolver traffic, including Cisco [11], Cloudflare [12], and DomainTools [13].

In this paper, we set out to evaluate the impact of the design decisions made by DNS-based ranking providers in their ranking designs, to establish whether DNS-based rankings are appropriate and sufficiently reliable for conducting Web and Internet measurements. For example, two design decisions introduced in the CrUX, Radar, and Tranco rankings [8, 12, 14] ('bucketing' ranks and long-term averaging) seek to address issues with stability that prior rankings experienced [7, 8]. More fundamentally, the shift to DNS-based rankings affects how reliably these DNS-based *domain* rankings approximate *website* popularity well, which was the focus of the disappearing Web-based rankings. However, we still lack a good understanding of whether the design decisions made for DNS-based rankings effectively contribute to improving their reliability and suitability across the dimensions important to research.

To identify which measures contribute to improving these rankings, we design a custom ranking method for passive DNS traffic data, from the ground up using aggregated "post-recursor" data. This enables us to holistically and independently evaluate the influence of design decisions on the composition and desired properties of domain rankings, and isolate these effects, as opposed to comparing existing rankings where the underlying data collection and processing are confounding factors [6–8, 10]. Since we use a feed of raw passive DNS traffic data, we also cover low-level design decisions, such as TTL weighting or CNAME correcting mechanisms, which cannot be readily analyzed using existing rankings and whose impact on rankings was therefore previously undocumented.

Through our evaluation, we demonstrate that the recent design decisions on bucketing and aggregation improve ranking stability, that the correcting mechanisms are necessary to avoid that certain domain types overly dominate the ranking, and that service classification caters for the different usage patterns of domain rankings. After TTL weighting, query distributions closely match those observed on the web but are more skewed, meaning that the design decision to incorporate TTL (or not) must be carefully made. Overall, our contribution confirms that the design decisions introduced in recent (DNS-based) rankings are beneficial and make these rankings suitable for

Internet and Web measurements. This improved understanding of the fundamentals of building these rankings allows our community to continue working towards rankings that meet the needs of researchers and that fulfill the soundness and validity requirements that ultimately increase trust in the resulting research findings [15].

This paper is an extension of our prior work on passive DNS-based rankings [16]. In this version, we further develop a critical analysis of passive-DNS based rankings, with a focus on influences of potential data biases and the resistance to manipulation by malicious actors. Such manipulation can make rankings misrepresent the popularity of domains and websites, which not only affects research measurements but has security implications as popularity is often used as a proxy for benignness [8]. We also improve and update our discussion of the background and related work, including to reflect recently published work on domain popularity lists as well as work on passive DNS traffic analysis.

TABLE I
EXISTING RANKINGS SHOW GREAT VARIATION ACROSS THEIR DESIGN DECISIONS, WHICH WE ANALYZE INDEPENDENTLY.

Name	Source	*Contains* websites	*Contains* Infrastr. domains	*Contains* Root domains	*Contains* FQDNs	# daily queries	Bucketed ranking	Length	Data period	Update frequency
Web-based										
Alexa	[8]	✓	✗	✓	✗	N/A	✗	1M	1d	1d
CrUX	[14]	✓	✗	✓	✓	N/A	✓	18M	30d	30d
Majestic	[8]	✓	✗	✓	✓	N/A	✗	1M	120d	1d
Quantcast	[8]	✓	✗	✓	✗	N/A	✗	0.5M	1d	1d
DNS-based										
Radar	[12]	✓	✓	✓	✗	1T	✓	1M	7d	7d
SecRank	[17]	✓	✓	✓	✓	500G	✗	1M	1d	1d
Umbrella	[11]	✓	✓	✓	✓	620G	✗	1M	2d	1d
Hybrid (aggregated)										
Tranco	[8]	✓	✓	✓	✓	N/A	✗	3.6M	30d	1d

II. MOTIVATION AND BACKGROUND

A. Data sources for rankings

Popularity rankings capture the most popular or "top" of either *websites* or *domain names*.

Web-based rankings were the earliest to appear, collecting web traffic data from participating users and websites to rank websites [8]. Web traffic represents "ground truth" of web page visits from participating users or websites, improving its accuracy. However, there may be difficulties in getting (a representative sample of) users or websites to participate, not in the least due to privacy concerns regarding the collection of users' browsing behavior. Owing to these challenges, the Web-based rankings that researchers rely on are steadily disappearing, with Quantcast and Alexa discontinuing their rankings in 2020 and 2022 respectively. Only the Chrome User Experience Report (CrUX) [14] has emerged as an alternative Web-based ranking since then, which depends on Chrome users opting in to URL sharing [6]. As an alternative web-based ranking, providers like Majestic use the "link graph", where website interlinking is used as a proxy for likely web traffic, i.e., the more links refer to a web page, the more likely it is that that page gets visited. The top websites are those most often referred to from other websites, similar to how search engines (at their core) rank search results [8]. As web graph crawling can be done independently without the need for users or websites to explicitly participate, it is easier for link graphs to have broad and representative coverage of users and websites, and link graphs do not observe data on any individual user.

DNS-based rankings by contrast use passive measurements of DNS queries for domains as observed at one or more DNS resolvers. For example, Cloudflare's *Radar* list [12] uses traffic to their 1.1.1.1 resolvers. Passive DNS-based rankings search a middle ground between the accuracy of web traffic and the coverage, relative ease of acquisition, and privacy preservation in link graph data. To better achieve this, a common thread across these DNS-based rankings is that they seek to incorporate the number of distinct entities (users, organizations) querying a domain, often to better approximate genuine human Web traffic. The techniques for this can vary from simple counting of the number of organizations contacting a domain a day, to being quite advanced: for example, Cloudflare uses a machine learning model that predicts each domain's rank [12].

B. What ranking properties do we expect?

To thoroughly evaluate whether rankings fulfill community needs, we must first understand what the community expects from these rankings. Inherently, we want the ranks to reflect how *popular* a given domain is. However, what constitutes popularity in and of itself is not necessarily well-defined. Most (passive DNS-based) rankings define popularity as the number of users or organizations accessing a domain, usually counting at most one access per day. However, this disregards how often and how regularly a domain is queried — for example, a search engine or news website might be visited very often throughout one day, yet would not have a better rank than a software update domain that is queried only once a day but by as many users. As one example of a more elaborate metric to quantify popularity, the ranking approach of SecRank [17] integrates query volume, query regularity, and IP address count.

Rankings should also exhibit properties that serve as requirements to make them more suitable for research usage. Prior work evaluated rankings across these dimensions [6–8, 17]:

- *Accuracy*: rankings should correctly capture popularity, i.e., a better ranked domain should genuinely be more popular than a worse ranked one.
- *Agreement* or *similarity*: if all rankings correctly capture popularity, they should place domains at the same ranks.
- *Manipulation resistance*: rankings should be designed such that domains cannot easily be inserted, removed, or shifted, as that would reduce the trustworthiness of measurements or affect use cases where benignness is inferred from popularity.
- *Representativeness*: rankings should correctly reflect Internet-wide distributions and have good coverage across, e.g., site categories, or countries.

- *Reproducibility*: rankings should be easily retrievable and uniquely referenceable such that others can reproduce studies with the exact rankings used in those studies.
- *Stability*: rankings should be sufficiently stable over time, while still incorporating natural changes to popularity.
- *Transparency*: rankings should publish details on the methods used in the ranking for better insight into possible biases.

C. Design decisions on composition

Certain design decisions can be made to influence the composition of a ranking and ultimately contributed to achieving the previously listed desirable properties of top lists. Some of these design decisions affect how well a ranking reflects popularity, while others impact how a ranking can be used and how it should be interpreted. Table I shows how existing rankings vary greatly on the design decisions they have taken. Our evaluation (Section V) is focused on assessing how significantly these design decisions affect a ranking's composition, and doing so in isolation to avoid confounding factors in existing rankings, by building a ranking from raw data.

- Domains represent *websites*, i.e., resources that humans tend to view in a top-level browsing context, or *infrastructure*, i.e., resources such as nameservers. Depending on its data source, a ranking may naturally tend to contain more domains of one type: Web traffic and link graph lists will likely almost exclusively contain websites, while DNS-based lists will also include infrastructure domains. This distinction will affect what data can be retrieved from a domain and is therefore important to ensure that a ranking fits a research study's purpose. For example, an infrastructure domain may not host any Web content, therefore appearing unreachable in a Web measurement (and potentially skewing its results).
- A ranking can list *root*[1] domains (e.g., Alexa), *fully qualified domain names* (e.g., Umbrella), *origins* (e.g., CrUX), or even *URLs* (e.g., Hispar [9]). This will affect the level of detail in the analysis: e.g., measuring only root domains ignores subdomain-specific properties or content.
- Each entry can have an individual rank, or entries can be 'bucketed', where only a rank bracket is given (e.g., 'top 10000'), as is done in Radar and CrUX. This is meant to better match the 'long-tail effect': human browsing patterns concentrate on a small number of very popular websites [18], while less popular websites (in the long tail) quickly have much less traffic and therefore become more difficult to accurately rank [12]. While most studies ignore individual ranks in existing lists [6], the lack of ranks can still affect the level of detail at which a certain observation can be correlated to a domain's rank.
- A ranking can be computed using data from a certain period of time. This time frame will affect the ranking's stability versus agility, as well as its update frequency. CrUX is averaged across (and updated every) 30 days, Radar across 7 days, and most other lists across 1 day. Tranco is updated every day but averages across 30 days of data.
- Each of these decisions, as well as the observed traffic volumes and (diversity of) traffic origins in a data source, affects the final length of a ranking. Commonly, rankings have been cut off at 1 million entries. The ranking length will affect how many resources a measurement can (dis)cover.

[1]Also called *eTLD+1* — one level above the effective top-level domain — or *pay-level* domains.

D. Related work

Lists of popular domain names have long been used in Internet measurement research (since at least 2005 [19, 20]), but were only thoroughly scrutinized starting in 2018. For three rankings (Alexa, Umbrella, and Majestic), Scheitle et al. [7] described the methods, research usage, characteristics such as stability, and potential impact on Internet measurement research. Le Pochat et al. [8] similarly characterized the three rankings and Quantcast from a security perspective, and showed that all four rankings could be manipulated to insert any domain. They also developed the Tranco list [8, 21], aggregating multiple existing rankings to improve properties such as stability and manipulation resistance. However, while Tranco's construction method is public, it ultimately relies on existing, opaquely constructed rankings, and therefore does not allow for independent evaluation of design decisions. Rweyemamu et al. studied aspects such as weekly patterns and domain clusters in detail [22], and refined manipulation attacks for Alexa and Umbrella [23]. Xu et al. [24] showed how certain open DNS resolvers respond to non-recursive queries in a way that can be abused to manipulate passive DNS-based rankings. Finally, Xie et al. [17] developed SecRank, a ranking method for pre-recursor passive DNS traffic. They evaluate the impact of their decision to use weighted voting across IP addresses, focusing on ranking manipulation and stability. We complement their work by evaluating post-recursor passive DNS data and decisions on TTL weighting and correcting mechanisms.

Beyond these proposals and evaluations for public rankings, several DNS-based ranking methods have been proposed but did not materialize to concrete rankings. Proposals include ranking domains on counts of querying DNS resolvers [25], unique querying clients [26], or per-client DNS query counts [27]. Other proposals estimate client query volumes and therefore popularity through active DNS cache probing [28–31]. Finally, the DNS Observatory [32] tracks top objects including domains in passive DNS post-recursor traffic, similar to our data set.

These DNS-based ranking methods rely on the rich information that can be extracted from a stream of passive-DNS queries and answers. Hao et al. observed DNS queries to newly registered domains and detected malicious domains from query patterns, domain names and entry content [33, 34]. Similarly, Fernandez et al. used DNS query patterns and SPF configurations to detect spam domains before their malicious activity [35]. Lyu et al. identified network assets and their health based on passive-DNS traffic observation [36]. These methods highlight the rich diversity of information contained in DNS traffic, providing detailed information on domain activity, popularity, and the infrastructure and assets that rely on it.

III. Post-recursor passive DNS data

For our evaluation of ranking design decisions, we use a feed of "post-recursor" ("above-resolver" [32]) passive DNS

traffic data [37] from SIE Europe [38], which is aggregated from multiple "sensors", i.e., recursive resolvers. To readers unfamiliar with (passive) DNS measurement, we recommend the tutorial by van der Toorn et al. [39] as a primer.

A. Motivation

Passive DNS data is becoming increasingly common as a data source for domain rankings, as it is a useful large-scale repository of network activity that can be used to infer domain popularity. In particular, the properties of post-recursor passive DNS data make it a good candidate both for developing new rankings and for our evaluation of ranking design decisions.

Its advantages compared to other traffic data are:

- *Increased coverage and diversity:* Passive DNS data can be easily collected from (large) recursive resolvers, increasing the user base of networks and organizations across which domain popularity is measured, as opposed to Web traffic data, which is collected from individual users. Furthermore, passive DNS data can be aggregated across a diverse range of service providers, therefore smoothing over differing usage patterns, e.g., mixing residential and corporate traffic [7, 22].
- *Better preservation of user privacy:* The raw data is aggregated over the users of all resolvers and no IP-level data is retained. For Web traffic data and pre-recursor passive DNS data, raw data can be traced back to individual users, potentially revealing their personal browsing habits.
- *Availability:* Owing to the better privacy posture, providers may be more willing to share raw passive DNS data with researchers for developing or evaluating domain rankings.
- *Record breadth:* The raw data spans all observed DNS record types, enabling their usage for applying corrections (e.g., `CNAME` redirects) or for service classification (e.g., `MX`/`NS` hinting at non-websites).

Nevertheless, such data has certain disadvantages:

- DNS traffic mixes (human) top-level browsing context visits and (automated) background DNS resolutions. This means that a passive DNS-based ranking will interleave websites and infrastructural domains. This can highlight the importance of some infrastructure domains as they will appear in the ranking, but studies that focus on website popularity may need additional classification steps, such as those described and implemented in Section IV-A3.
- The selection of recursive resolvers matters for the (global) coverage and representativeness of the ranking. The risk exists that the user base is too small or too skewed to specific user types. Simultaneously, domain popularity is country-specific [18], making the coverage of countries also important to the representativeness of the data.
- The observed per-domain counts represent cache misses at the recursive resolvers, i.e., not all user requests are visible.
- Post-recursor passive DNS data precludes the use of ranking methods that incorporate per-user statistics, e.g., IP-based voting [17].

The combination of data availability, privacy preservation, and access to raw data makes post-recursor passive DNS data well suited to use for our evaluation of the influence of design decisions, including the reliability of passive DNS-based rankings for all areas of Web and Internet measurement [7].

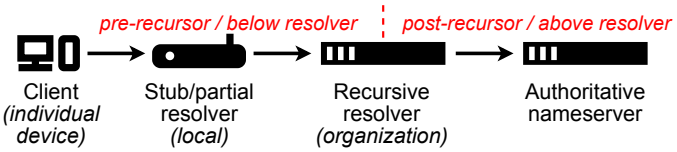

Fig. 1. Parties in a recursive DNS resolution, where we highlight the boundary between pre- and post-recursor traffic.

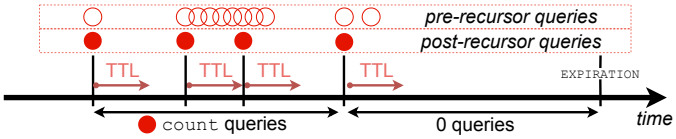

Fig. 2. Pre-/post-recursor DNS traffic have different query patterns, caused by resolver caching and TTL. We only observe the post-recursor query `count`s.

B. Data provenance and format

The input for the rankings that we generate for our evaluation is a continuous live feed of passive DNS entries in the *Passive DNS Common Output Format*, published as an IETF Internet Draft [37, 40] listing the mandatory and optional fields in each feed message. This type of feed is already available from large passive DNS aggregators such as Farsight Security's DNSDB,[2] SIE Europe [38], CIRCL,[3] or mnemonic.[4] Such a feed aggregates DNS requests from multiple 'sensors', i.e., recursive resolvers from organizations or networks that share their data with the passive DNS aggregator. Figure 1 shows how resolvers execute DNS resolution, and where the boundary between pre- and post-recursor traffic lies. The sensors only observe the post-recursor traffic, i.e., the recursive queries to the authoritative nameserver for entries missing from the recursive resolver's cache. Each sensor sends a copy of this traffic to the aggregator that will merge them into a single feed.

The passive DNS aggregator maintains its own cache with a count of observations of tuples of *requested resource* (`rrname` field, i.e., the domain), *record type* (`rrtype`, e.g., `A` or `AAAA`), and *record(s)* (`rdata`, value(s), e.g., IP address(es)). When a tuple is observed for the first time, the aggregator stores it in its cache. Subsequent observations by any sensor increment the count. Entries in this cache expire either when the cache is full or after a set time (for our SIE Europe feed, we estimate this expiration time at 6 hours), upon which an `EXPIRATION` message is emitted to the feed. This message contains the previously described tuple and a `count` field for the total number of observations across all sensors. Since the entry is being removed from the cache, this count is then implicitly reset to 0. Figure 2 shows how the `count` field relates to queries pre- and post-recursor; we observe only the latter. In addition, we rely on the `rrttl` field in this message for the (authoritative) Time-To-Live (TTL) record value. While this field is available in the data feed that we use, the field is

[2]https://www.farsightsecurity.com/solutions/dnsdb/
[3]https://www.circl.lu/services/passive-dns/
[4]https://docs.mnemonic.no/api/services/pdns/

not documented in the draft standard, so it is not guaranteed to be present. If necessary, it could be retrieved separately through an active DNS query. Finally, the draft standard has an optional `sensor_id` field for identifying the individual sensor at which the record was seen. It is not present in all passive DNS feeds (this also holds for ours), so we do not expect or use it. In addition, the use of this field could raise privacy issues as it may make it easier to trace observations of specific domains to specific sensors.

IV. Ranking generation

To conduct our evaluation of the influence of design decision, we use a feed of aggregated post-recursor passive DNS data to generate three rankings: one each for Fully Qualified Domain Names (FQDNs), root domains, and domains that likely host websites (i.e., pages visited by regular Internet users while browsing). Figure 3 schematically represents how we process the data feed to ultimately generate our three ranking types.

A. Processing resource records

1) Observation counters: For the final popularity tally, we track the counts of observations per domain (i.e., requested resource) for the `A` and `AAAA` record types. We only observe and count the requests above the recursive resolver — they represent resolver cache misses. Observation counts therefore depend on the resolver caching behavior, which is primarily fixed by the TTL value set by the authoritative nameserver and sent along with the resource records. TTL weighting may affect biases towards certain services and the representativeness of websites (Section V-B) as well as susceptibility to manipulation (Section VI-B). To evaluate TTL weighting, next to one variant ranking where we ignore TTL, we generate a second variant where we multiply all observation counts by this TTL value, extracted from the passive DNS data itself. Simply put, we expect a domain with twice the TTL value to have half as many visible requests, so we need to multiple its query count by two. In case the TTL is zero, we retain the original count.

TTL values of more than 1 day (86400 seconds) are commonly truncated by resolvers [41], and as we primarily use 1 day of data to generate rankings, we truncate all TTL values to 1 day as well. In terms of the effect this has, Figure 4 shows the distribution of TTL values over all queries (observed on 1 Sep. 2023) per record type. Almost all `MX` and `CNAME` records have a TTL below 1 day, whereas `A/AAAA` and `NS` entries have respectively 38%, 63% and 45% of their TTLs over 1 day (almost always at exactly 2 days). Because of the dominance of nameservers with 2-day TTLs (see also Section V-C), this truncation is applied for 48% of queries, although this only affects 3.2% of FQDNs.

2) CNAME reverse cache: We track the counts of `A/AAAA` record observations, i.e., query results instead of the queries themselves. Therefore, if a user resolves a domain that uses a `CNAME` record to map to the `A/AAAA` records of a second domain, we count an observation for this second domain, not for the domain the user originally visited. In particular, large CDNs would therefore be over-counted if not for correcting mechanisms (Section V-C), because CDN-hosted domains often point a `CNAME` record to a CDN (sub)domain that hosts the actual website [42, 43]. Another use case for `CNAME` redirection is mimicking first-party context for tracking purposes [44].

We therefore maintain a cache of all observed mappings in CNAME records observed in the feed of the ranking's day. We recursively reverse the resulting mapping for all domains in our final ranking. If multiple records map to the same domain, we select the reversed domain for which we observed the highest total query count. We then retain the mapped domain if it is observed often enough (`CNAME` count $> 1\%$ of the `A/AAAA` count) to ignore erroneous `CNAME` records, and if it is not a subdomain of the original domain, to prioritize root domains.

3) Service classifier: Our data feed comes from DNS queries, and therefore contains infrastructural domains such as nameservers that are not visited by humans through browsers. However, certain studies focus on user Web traffic and require website-only rankings, akin to Alexa, CrUX, or Majestic. An important design decision is therefore how websites versus infrastructural domains are accounted for, to avoid dominance of certain domain classes (Section V-C).

We therefore need to determine what *service* each domain offers, to then separate domains into service-specific rankings. In our design, we use internal and external *passive* sources to determine the service. Internally, we mark the domains appearing in `MX` and `NS` record values as mailservers and nameservers, respectively. We also specifically mark nameservers in the root zone file, as they are not included in the passive DNS feed [45]. We manually develop a service mapping for the most commonly observed subdomain labels, e.g., `www` for a web service, or `ns1` for a nameserver. Externally, we collect websites from three web-focused domain lists (BuiltWith,[5] CrUX [14], and DomainRank[6] [46]), and domains in DBpedia [47].

To resolve contradictory classifications from multiple sources (e.g., when a `NS` record mistakenly contains a web domain), we develop a scoring system that accounts for the reliability and the certainty between and within sources, e.g., awarding higher scores for records observed more often. We first classify individual FQDNs; if there is no known class, we recursively search a class of the parent domain. To determine the class of a root domain, we compare its known class (if any) with that of its subdomains, and heuristically select the most likely class based on the scores and query counts of each. If a domain offers multiple services, we assign it the most common service.

B. Computing the ranking

We generate daily rankings based on that day's query counts, CNAME reverse cache and service classifier data. To fill these counters and caches, we process 1-day slices of the data feed. For rankings across larger time periods (e.g., 7 days), which may increase ranking length (Section V-A), we combine these processed 1-day slices by summing counts and merging caches. We first compute the base FQDN ranking of fully qualified domain names, mapping each FQDN to its reversed CNAME domain if applicable. We then generate a 'root' ranking across the total counts for each root domain's subdomains. We extract

[5]https://builtwith.com/top-1m

[6]https://www.domainrank.io/download.html

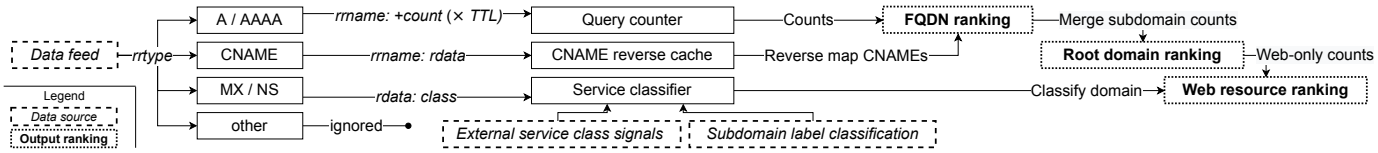

Fig. 3. We process the resource records from the passive DNS data feed to populate the counters and caches, which we use to compute three rankings.

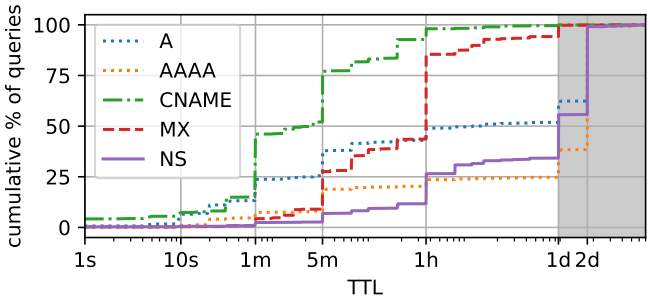

Fig. 4. The cumulative distribution of TTLs per record type shows that almost all `MX` and `CNAME` records have a TTL below 1 day, whereas `A/AAAA` and `NS` entries often have TTLs going up to 2 days.

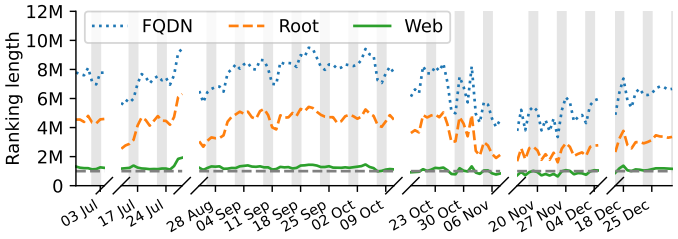

Fig. 5. 1-day FQDN rankings are longer than root and web rankings, at an average reduction of 42.7% and 70.1% respectively. (Weekends in gray.)

the root domain for each FQDN using the Public Suffix List (PSL) [48]. We use the public section of the PSL such that the root ranking contains only true eTLD+1s. This means that the root ranking has one single entry for domains listed in the private section (a user-contributed subset of domains for which subdomains are maintained by separate users, e.g., `*.blogspot.com`), and we sum the counts of all subdomains to determine the root domain's rank. A relatively large part of the FQDN ranking (4.95%) belongs to a domain in the PSL's private section, mostly owing to the listing of large CDN domains in the private section. Finally, we extract a 'web' ranking of root domains classified as a website, using only the counts of subdomains classified as a website.

V. Evaluation

To evaluate the influence of design decisions on rankings using real-world data, we apply our ranking generation method to passive DNS data sourced from SIE Europe [38], aggregated from sensors located at European-based commercial, government, and higher education organizations.[7] We generate the three ranking types (FQDN, root, and web) using the passive DNS feed for 27 June–4 July, 13–28 July, 24 August–11 October, 17 October–8 November, 15 November–5 December, and 16–30 December 2023 (gaps are due to outages; see Section VII-C). When we describe a ranking for a given day and time period, we use the data observed on that day and the days before it for the duration of the time period. We analyze the composition of the rankings, and the impact of the different steps of the resource record processing (Section IV-A) on the rankings.

A. Ranking length

We first quantify the raw lengths of the computed rankings, i.e., the total number of distinct observed domains over the given period. We do not impose any threshold on the observed query counts for inclusion in the ranking; Section V-B quantifies in detail the distribution of query counts, which could serve to select a threshold that meets requirements for accuracy and representativeness of a domain's popularity yet produces a sufficiently long ranking.

The lengths of the 1-day rankings vary over time between 3.3M and 9.5M FQDNs, 1.6M and 6.3M root domains, and 0.6M and 1.9M web domains (Figure 5); an average reduction of 42.7% and 70.1% respectively. This shows that the choices of excluding subdomains and/or prioritizing websites significantly reduces a ranking's length. In the first half of our measurement period (until 6 October), the length of the web ranking always exceeds 1 million websites, but this no longer always holds afterwards. Figure 6 shows how raw query volumes dropped from a peak of 1.4 to a low of 0.3 billion queries per day.[8]; If the query volume becomes too low, the ranking no longer achieves the threshold of 1 million websites that is commonly used in other rankings, highlighting the need to observe a sufficient traffic volume to capture diverse long-tail domains.

One way of augmenting query volumes is to aggregate traffic data over a longer timespan. Figure 7 shows how the lengths increase for 7-day rankings, yielding an even larger set of domains that can be measured. Owing to the commonality of some ranked domains across time, these lengths do not increase seven-fold. Instead, the increase in ranking length is between a factor of 2.7 and 6.0 for FQDN rankings, 2.7 and 7.7[9] for root rankings, and 2.2 and 4.7 for web rankings. As a result, the 7-day web ranking contains at least 2.5M websites, comfortably meeting the 1M threshold.

Takeaway: It is necessary to observe a sufficiently large traffic volume such that there are enough distinct domains/websites in the ranking. Traffic volume can be increased by aggregating across multiple days, as is done in, e.g., Radar (7 days).

[7]No details are available on the exact set of organizations contributing to SIE Europe. We also make no attempt at deanonymizing them.

[8]We have no immediate explanation for this variance in query volume. Possible causes are an organic (temporary) decrease in traffic, the removal or malfunctioning of sensors or the data feed, or bandwidth limitations.

[9]This outlier (above 7) corresponds to the trough in query volume, where the data for the previous days contain more domains unseen in the 1-day data.

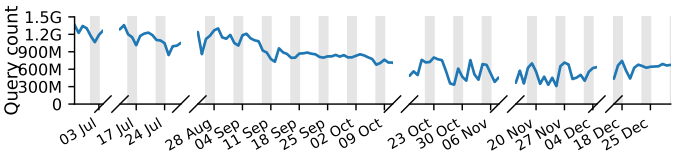

Fig. 6. Observed query counts fluctuate over time, although we do not observe a direct correlation with ranking lengths.

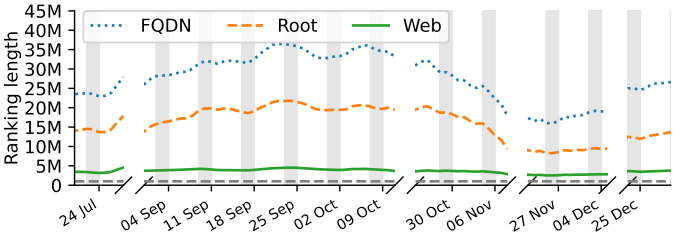

Fig. 7. 7-day rankings, as shown, are much larger than 1-day rankings. Web rankings easily surpass the common 1 million website threshold.

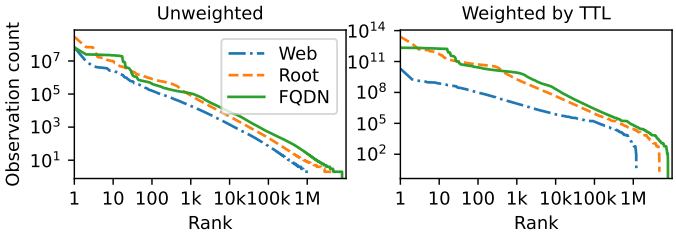

Fig. 8. The rank-size distribution of (un)weighted observation counts appear to adhere to a power law distribution (1 Sep. 2023 rankings).

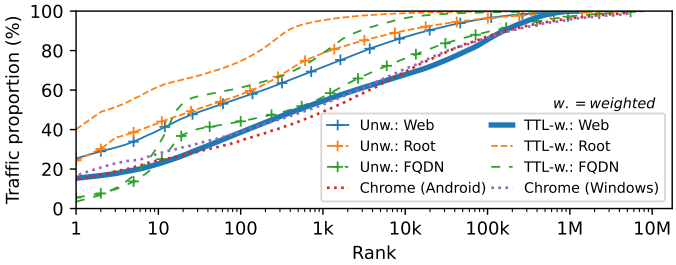

Fig. 9. The TTL-weighted cumulative traffic distribution closely follows the distribution of Google Chrome web traffic [6, Figure 1] (1 Sep. 2023 rankings).

B. Observation count distribution

Unlike for existing rankings, we have access to the raw scores that determine the final domain ranking. We use this to characterize the estimated traffic distribution across the ranking and its dependence on incorporating TTL, and to understand whether the ranking exhibits patterns that have previously been observed in web traffic.

Figure 8 shows the distribution of the observation counts, both unweighted and weighted by TTL over the domain rank, for the ranking of 1 September 2023. Distributions on other days are similar. Overall, the score distributions appear to adhere to a power law distribution, i.e., the rank and count have an inverse relation. This distribution matches previous observations in website and domain popularity distributions [8, 18, 49, 50]. For the FQDN rankings, there is a stagnation and then drop-off in the head (well-ranked domains). We conjecture that it is the effect of resolver caching: the expected traffic increase (in client queries) is dampened (in cache misses) because caches already serve these requests without going to the authoritative nameserver. There may also be remaining effects from the overcounting of nameservers (Section IV-A1). There is also a drop-off in the tails of all rankings, likely due to shared low discrete query counts. Another view on this ranking tail comes from the absolute differences in query counts from one rank to another. From a rank of around 1,000 onwards, differences sometimes drop to 0 and are generally very low. This effect has also been previously observed in existing rankings, where it was visible through alphabetically ranked clusters [22]. This seems to confirm prior observations [8] that differences between individual ranks in the tail are not (statistically) significant, suggesting that bucketing is also useful for grouping domains with very similar traffic levels.

A final way to view the traffic distribution is by charting the cumulative traffic proportion along the ranking (Figure 9). For the FQDN and root rankings, unweighted rankings attribute more traffic to the head, while for web rankings, the opposite is true. We also compare the web distribution to the Google Chrome traffic distributions traced from Ruth et al. [18]. Here, the TTL-weighted web ranking closely follows this distribution; the unweighted ranking assigns more importance to the head.

Takeaway: The observed traffic distributions have a significant effect on the resulting rankings, exacerbated by TTL. The choice to weight query counts by TTL affects the granularity at which domain query volumes can be compared, and how the resulting distributions match those observed empirically.

C. Correcting mechanisms

We develop two correcting mechanisms as part of our ranking method: CNAME reversal and service classification. On 1 September 2023, across the FQDN ranking, we mapped 261,486 (3.26%) of all subdomains to another subdomain based on CNAME reversal. 90,084 (34.5%) of them mapped to another subdomain within the same root domain. The remaining subdomains are distributed across 24,613 unique root domains and mapped to one of 95,016 other root domains. Across these other subdomains, major CDNs and managed DNS providers are concentrated at the top (e.g., *cloudflare.net*, *azure.com*), who would be overcounted without the CNAME reversal.

54% of the FQDN ranking on 1 September 2023 is classified as a specific service, of which 69.5% as websites (Table II). The rest cannot reliably be classified using our selection of passive sources, and may require additional signals such as an active crawl. Within the root ranking, 33% of domains can be classified, of which 79.7% as a website to be included in the web ranking. Given that our external sources for websites integrate a notion of popularity (for rankings) or notability (for Wiki sources), any unclassified root domains that would be websites are more likely to be unpopular or uninteresting, next to disposable [51] or algorithmically generated [52] domains. In terms of the distribution of service classes over the ranking, nameservers dominate the ranking head, in particular for the FQDN ranking and when weighting by TTL (Figure 10).

Takeaway: Using correcting mechanisms, such as CNAME reversal or a service classifier, corrects for otherwise overly dominant domains such as CDNs and nameservers.

TABLE II
SERVICE CLASS DISTRIBUTION (1 SEP. 2023 FQDN RANKING).

Class	Percentage	Class	Percentage
Unclassified	45.79	IPv4 address	0.89
Website	37.65	CDN	0.70
Nameserver	9.31	Other web service	0.44
Mailserver	3.45	Protocol (FTP, ...)	0.36
Web admin panel	1.07	UUID	0.34

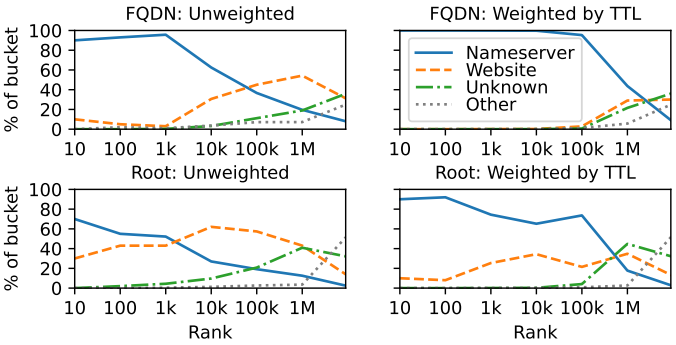

Fig. 10. The distribution of service classes over buckets shows that nameservers dominate the ranking head (1 Sep. 2023 rankings).

VI. RESISTANCE TO BIAS AND MANIPULATION

Domain rankings see frequent use in research studies and as lists of benign domains [8, 22, 23]. For example, in 2019, Le Pochat et al. [8] listed over 130 security studies that relied on data from a public popularity list like Alexa, Umbrella, Majestic and Quantcast, to either evaluate the deployment of technologies, measure the prevalence of vulnerabilities, build allowlists or as part of reputation algorithms. As a consequence, the presence of a domain in a popularity list can affect its visibility, attracting more attention from security researchers, as well as its reputation, as popular domains are often considered as more likely to be trustworthy and less likely to be malicious. For these reasons, some actors could try to manipulate popularity lists to have their domains in them, while other actors could try to have their domains not appear in these lists [53, 54].

Next to this potential targeting through active manipulation with malicious intent, the potential for "manipulation" of domain rankings also covers the effect of inadvertent misconfigurations, data inaccuracies, or other biases that may unduly influence these rankings, and any systems that rely on them. For example, Chiba et al. [54] highlight how trustworthy yet less-visited websites are often missing from popularity lists, causing their omission from allowlists and thus potentially improper blocking since the legitimate status of the website is not recognized. Popularity list designs must take those effects in consideration and identify undesirable influences on the composition and order of their lists.

A. Injection of traffic or links

Certain manipulation techniques that are based on injection of traffic or links to feign popularity heavily depend on the data sources the list relies on (Section II-A). Web traffic-based rankings are vulnerable to injection of forged visits, either as a fake user for any number of sites when targeting the reporting browser extensions, or as many fake users for a specific site when targeting embedded scripts [8, 23]. Link graph-based rankings are vulnerable to fake 'backlinks', i.e. hyperlinks injected into existing web pages through payment or abuse of a website vulnerability, which the link graph provider's crawler then follows [8, 55].

Passive DNS-based rankings that attempt to approximate user counts are vulnerable to injection of forged DNS queries that are made to appear to come from many distinct users (i.e. IP addresses) [8, 23]. This depends on the attacker's ability to direct those DNS queries to one or more DNS resolvers that contribute their traffic statistics to the popularity ranking. One attack venue requires that the attacker has access to a sufficiently large pool of IP addresses, e.g., by cycling through the IP ranges of cloud providers or by finding open traffic relays and issues the DNS queries directly from those IP addresses to the DNS resolvers. For example, Xu et al. [56] successfully manipulated passive DNS-based rankings (Umbrella, SecRank and Farsight) by sending DNS queries to open forwarders. These forwarders transmit the received queries to downstream resolvers. As a consequence, passive DNS providers using DNS traffic from one of these downstream resolvers will observe queries to the domain. In the case of Xu et al., they could manipulate Umbrella and SecRank because those rankings use data from OpenDNS and 114DNS respectively, two public resolvers that are downstream of multiple public forwarders. Similarly, even though Farsight relies on an undisclosed set of sensors deployed at multiple resolvers, the fact that Xu et al. were able to insert a domain in their top list indicates that some of Farsight's sensors are deployed on resolvers downstream of some open forwarders.

Some passive DNS nodes are located at closed resolvers (like commercial, government, and higher education organizations) and only serve queries sent by users inside their respective networks. However, in networks that do not perform Source Address Validation [57–59], it is possible to spoof the source IP of the DNS queries [60] in order to generate fake traffic to DNS resolvers that seemingly originates from inside the target network. If an attacker manages to identify the location of passive DNS nodes in networks not implementing Source Address Validation, they can send a few spoofed DNS queries to the resolvers every N seconds, where N is the `TTL` value of the `A/AAAA` record of the domain, thus triggering artificial domain resolutions from resolvers that would otherwise only serve requests from users inside their network while requiring low bandwidth usage from the attacker. The resulting post-recursor traffic will be similar to a highly popular domain.

The effects of these two traffic injection methods (open forwarders and spoofed IP) depends on the number of passive DNS sensors impacted — sensors on resolvers downstream of open forwarders or in a network vulnerable to IP spoofing. The resolver cache will limit the volume of queries each sensor can observe and the aggregation process will reduce the relative count of the domain. However, if multiple sensors are vulnerable and attacked, the resulting aggregated stream could be efficiently manipulated. As a mitigation to limit the impact of such attacks, passive-DNS providers could require the

deployment of anti-spoofing mechanisms like Source Address Validation on networks where sensors are deployed, and actively look for sensors downstream of open forwarders.

B. Attacker-controlled parameters

Every domain ranking algorithm relies, at one step of its design, on data provided by the domain owners. As a consequence, attackers may use specific configurations to deceive the ranking algorithm and increase their popularity. For example, web traffic-based designs are vulnerable to websites embedding scripts and iframes triggering automatic visits to third-party websites, often without the user's knowledge, while making it difficult for ranking algorithms to distinguish legitimate user traffic from artificial traffic. Similarly, attackers can exploit graph-based designs by artificially increasing the number of links between their website and the rest of the web: outgoing links can be added to their websites, and incoming backlinks to their websites can be inserted on popular public platforms such as blogs, social networks, wikis and forums [61].

For rankings based on passive DNS, the attacker can adapt to the design choices of the popularity list regarding `TTL` weighting (Section V-B). A list that weights each DNS record `count` by its `TTL` value, to account for record caching, is vulnerable to an attacker creating a high-`TTL` record, as this reduces the number of spoofed or forwarded queries needed to emulate high user traffic, with TTL thereby acting as an amplifier [32, 62]. On the contrary, a list that does not weight record `count` by the `TTL` is vulnerable to an attacker creating a low-`TTL` record that will quickly expire from the different caches, allowing for a new query that will be observed in the post-recursor passive DNS feed. The total unweighted observation counts (Section V-B) indicate the query volume required to insert a domain at a given rank. To limit the effect of such `TTL` manipulations, rankings could truncate high `TTL` values or integrate the `TTL` in a non-linear way by multiplying the `count` by $\log(ttl)$ or $\sqrt{(ttl)}$ to reduce the potential amplification factor of extreme values.

Section IV and Figure 3 described additional mechanisms that may be necessary to convert aggregated passive DNS traffic to rankings, such as subdomain merging, CNAME reverse caching, or service classification. All these steps rely on data partially provided by the attacker and thus can be manipulated to exploit design choices. As such, creating multiple low-`TTL` subdomains can amplify any successful traffic injection, as each subdomain will have its own `TTL` cache expiration timeout, but the observed queries to each subdomain will be merged together to generate the root domain ranking. Similarly, successful traffic injection can be amplified by having multiple CNAME entries pointing to the attacker-controlled domain, bypassing the `TTL` caching of the main domain.

Lastly, an attacker may want to manipulate the results of the service classifier to manipulate whether their domain appears or not in web-only lists by poisoning the data sources of the classifier. This kind of data poisoning was proven possible by Carlini et al. [61], who manipulated deep-learning training datasets by changing web page contents after labeling or changing the contents of crowd-sourced content temporarily during labeling. As a parallel, our service classifier relies in part on DBpedia and Wikidata, both crowd-sourced datasets, where an attacker could thus inject domains to be (falsely) classified as websites.

TABLE III
THE BEST RANK REACHED IN EACH LIST, IF AN ATTACKER SUCCESSFULLY INJECTS 1 QUERY PER `TTL` TO A SENSOR.

Target List	Unweighted `TTL` = 1 s	Weighted `TTL` = 86 400 s (1 day)
fqdn	1 404	894 625
root	968	505 724
web	203	145 544

To insert a domain in the top list through traffic injection, the attacker needs to identify how to reach passive DNS sensors, either through open forwarders or IP spoofing. We run a small-scale experiment to estimate the rank that an attacker could theoretically reach, if they are able to maximize the fake traffic detected at the passive DNS resolvers while limiting the total number of queries needed. We assume the attacker sets the `TTL` to 1 second for unweighted lists (to reduce the cache time of the domain), and to 1 day for TTL-weighted lists (though any TTL could be picked with the results staying the same, as the number of cache misses will be weighted). The attacker then injects one query per `TTL` (the highest traffic each sensor can detect) to a domain, without using `CNAME` or subdomains. We finally match the theoretical post-weighting observation counts with the empirical observation counts from Section V-B, to derive the theoretical best rank that can be achieved.

Based on this achieved rank (Table III), we observe that unweighted rankings are more vulnerable to traffic injection, because the record quickly expires from the resolver cache and can be queried again, whereas legitimate domains may use longer `TTL`s that will stay in the cache without triggering new queries. TTL-weighted rankings are more robust to traffic injection: there is no difference between a `TTL` = 1 s record queried every second and a `TTL` = 86 400 s record queried every day, both will generate a weighted observation count of 86 400. However, the number of forwarded or spoofed queries needed will be lower for a high-`TTL` record, thus consuming less bandwidth to run the attack. As a consequence, the attacker must identify multiple sensors, set up `CNAME` and subdomains to reach higher counts and have a chance to reach the top 1 000 domains.

C. Drop-catching and misconfigurations

Some manipulation techniques apply to all popularity lists, as they do not target the data sources but the domains themselves. For example, Lever et al. [63] described how malicious actors (called "drop-catchers") register recently expired benign domains to take advantage of their previous good reputation and hide their malicious activity. This manipulation technique can be hard to detect, depending on the accessible data. However, passive DNS based lists could setup defense mechanisms against these attacks. Passive DNS entries to `A/AAAA` records are used to generate the total domain traffic count, but they also contain the `rdata` field that provides the actual IP addresses

in the records. In addition to query counts, popularity lists could keep track of the IPs used by each domain and use this data to detect a change in ownership through the sudden modification of such addresses and reflect the possibility of malicious activity in the ranking.

Lastly, even without an active malicious attacker trying to manipulate the results, common domain misconfigurations could modify the ranking. For example, we observe multiple domains with extreme `TTL` values, potentially skewing the weighting of observation counts, or incoherent `NS` and `MX` records, which lead to misclassification of domains to name-server or mailservers respectively. These errors are probably due to configuration errors or a misunderstanding of the underlying protocols, but that still could have an impact on the domain ranking. In our ranking design, truncation of TTL values (Section IV-A1) and a scoring system across multiple sources for classification data (Section IV-A3) help mitigate this impact.

VII. Limitations

A. Cache Misses & Aggregation

Rankings based on post-recursor passive DNS traffic are inherently limited by the traffic representing only cache misses, heavily aggregated across large-scale recursive resolvers. We have no true count of client queries (to the recursive resolver) nor identifiers for individual clients or sensors, and can therefore not incorporate this into the ranking generation process for our evaluation as performed by [17] through their voting mechanisms; hence our goal of evaluating whether post-recursor traffic can still be useful for modeling domain popularity despite these restrictions. The aggregator cache also reduces temporal granularity, as counts are aggregated over six hours, which we deem too long to meaningfully integrate in our design, e.g., for client query modeling based on DNS request inter-arrival times [29]. Evolutions and optimizations in DNS resolution, such as QNAME minimization [64, 65] or encrypted queries [66], may affect the reliability or comprehensiveness of our traffic data. Since we observe DNS queries and not (top-level browsing context) web visits, query counts may also be inflated by website redirects [67], and by the inclusion of third-party resources.

B. Record Values & TTL

In terms of data quality, we have no reason to believe that the counts provided by the passive DNS aggregator or individual resolvers would be inaccurate. In contrast, we rely on record values set by the domain operators themselves. We sometimes observe inaccuracies that could affect our data processing and subsequent results, e.g., `google.com` incorrectly being set as a `CNAME` or `NS` record value. We account for these errors by (heuristically) setting minimum thresholds for our CNAME reverse cache, and by prioritizing data sources based on reliability for our service classifier. We also use TTL values as provided by authoritative servers, which resolvers may not always respect [17, 68] — e.g., Moura et al. found that values under 1 hour were rarely changed, but TTLs over 1 day were more frequently truncated by the resolvers [41, 62].

C. Data Source

We monitor, store, and process a live feed of passive DNS entries as the basis of our data set. Such feeds can produce up to tens of gigabytes of aggregated data per hour, requiring a stable and fast network connection with the aggregator and efficient storage systems to preserve all data. During our data collection, we experienced multiple small network disruptions that led to the ingestion of the live feed silently stopping, requiring manual intervention to restart feed processing. These outages has led to gaps in our source data and therefore our daily generated rankings. We verified for the date ranges that we retain (Section V) that the feed ingestion ran uninterrupted for the entire day; we discarded rankings of days where fewer than 24 hours of data was collected.

The lack of definitive ground truth to evaluate accuracy forms a challenge for our evaluation. Ruth et al. [6] come closest, comparing existing rankings to Cloudflare web traffic (covering around 10% of all websites), but given a lack of open data, we cannot replicate their method. We are also dependent on one specific data aggregator for our evaluation. Our method can be applied to all passive DNS data sources, but our concrete results inherently only hold for SIE Europe data, and are therefore biased to the usage patterns of the users of its European-based resolvers.

This geographical effect is apparent in the distribution of the TLDs of observed domains. We count the number of queries per TLD observed by SIE Europe (data from European resolvers) and by DomainTools[10] (data from resolvers worldwide) and compared the 20 most represented TLDs in each data source (Table IV). The difference is most visible when comparing country-code TLDs (ccTLDs): European ccTLDs (highlighted in grey) like `.de`, `.eu`, `.it`, `.fr`, `.dk`, `.lv` and `.re` (respectively for Germany, European Union, Italy, France, Denmark, Latvia and Réunion) are ranked higher in SIE Europe than in DomainTools. Similarly, non-European ccTLDs like `.cn`, `.au`, `.jp`, `.br`, `.il` and `tv` (respectively for China, Australia, Japan, Brasil, Israel and Tuvalu, which are non-European countries) are ranked lower in SIE Europe. A few notable exceptions are `.ru` and `.uk` (Russia and United Kingdom) ranked higher in Domain Tools – which may suggest that those countries are not represented in SIE Europe's European resolver base –, and `.us` (United States) being ranked higher in SIE Europe.

Due to these biases, we refrain from directly comparing to other top sites rankings, and instead focus on isolating influences within our own ranking data.

VIII. Ethics & Reproducibility

Processing DNS query data may come with a privacy risk, as it could be used to identify individual users and the domains that they access [69–72]. We only have access to count data aggregated from post-recursor requests across multiple sensors that mix cache misses originating from many users and institutions. The data does not contain any explicit fields with likely personally identifiable information (PII), such as the

[10]https://research.domaintools.com/statistics/tld-counts/

TABLE IV
THE 20 MOST POPULAR TLDS IN PASSIVE DNS TRAFFIC FROM SIE EUROPE (EU) OR DOMAINTOOLS (DT) ARE SKEWED TOWARDS THEIR GEOGRAPHICAL RESOLVER BASE (EUROPE VS. WORLDWIDE, RESPECTIVELY). EUROPEAN CCTLDS ARE HIGHLIGHTED IN GREY.

TLD	Rank in DT	Rank in EU	
com	1.	1.	
net	2.	2.	
ru	3.	5.	(-2)
arpa	4.	8.	(-4)
de	5.	3.	(+2)
cloud	6.	12.	(-6)
io	7.	9.	(-2)
cn	8.	16.	(-8)
org	9.	4.	(+5)
dev	10.	31.	(-21)
top	11.	44.	(-33)
au	12.	37.	(-25)
uk	13.	23.	(-10)
mil	14.	150.	(-136)
xyz	15.	11.	(+4)
online	16.	26.	(-10)
jp	17.	34.	(-17)
br	18.	50.	(-32)
info	19.	18.	(+1)
il	20.	155.	(-135)
tv	21.	19.	(+2)
us	22.	10.	(+17)
co	23.	20.	(+3)
eu	24.	7.	(+17)
it	35.	17.	(+18)
fr	37.	13.	(+24)
dk	49.	6.	(+43)
lv	69.	15.	(+54)
re	154.	14.	(+140)

client IP address. We already identify domain names (likely) containing IP addresses as a separate class, and would discard them for any publicly available rankings. We further aggregate the raw counts from the passive DNS feed per domain, and for a final public ranking would select a reasonable cutoff (on query count or ranking length) to retain only sufficiently popular domains and avoid publicizing private internal domains. We, therefore, consider that user privacy is preserved.

To enable reproducing our work, we make the ranking generation code, ranking files, and analysis scripts and results available. These resources are available from https://domain-ranking-design-decisions.distrinet-research.be.

IX. DISCUSSION AND CONCLUSION

Our goal was to evaluate the impact of design decisions on the development and characteristics of rankings using "post-recursor" passive DNS data, and understand how to leverage such data for generating domain rankings that are suitable for Internet and Web measurements. Based on the design of a ranking method and evaluation on the separate design decisions, we can draw several conclusions on the challenges and necessary steps that turn such data into a usable ranking.

At its base, passive DNS data is usable for generating a website-oriented ranking, primarily after the service classification step; otherwise, nameservers are dominant. This step also lends itself to further development and refinement: we deliberately chose to work only with passive service class indicators to show that this resource-limited approach is already feasible, but service classification may be improved with active indicators (i.e., web crawls that reveal the presence of useful content over HTTP), given the availability of the necessary resources. Correcting factors are necessary: in addition to service classification, we found that without CNAME reversal, CDNs and managed DNS providers would be overrepresented.

Specifically for DNS-based rankings, TTL can be an important factor in accurately assessing domain popularity, as it affects the temporal querying patterns (Figure 2). While incorporating TTL makes theoretical sense, it appears that this does not necessarily improve all ranking properties. For example, nameservers dominate TTL-weighted rankings more heavily, and the observation count distribution is more skewed at the tail. Nevertheless, TTL-weighted web rankings follow known web traffic distributions more closely, so appear more representative in this regard. The TTL policy also impacts the manipulation robustness of rankings: TTL-weighted rankings require attackers to correctly identify multiple sensors and leverage `CNAME` and subdomain amplification to reach high ranks, but the attack requires less traffic to be injected, whereas unweighted rankings are more vulnerable to manipulation, but the attacker will have to inject more queries. Incorporating TTL must therefore be carefully considered. For example, Xie et al. [17] decided to omit TTL from SecRank as it was difficult to reliably infer resolver caching behavior from TTL. More abstractly, this shows how even for the exact same data set, one design decision can result in wildly different outputs, suggesting that reliably comparing all existing rankings and their variation in data sources and methods is even more challenging. For example, Ruth et al. [6] already had to introduce several normalizations and adjustments, such as filtering on Cloudflare-only sites, to allow for ranking comparison.

Two more recent ranking design elements appear to have a positive impact: rank buckets and long-term aggregation. Buckets yield significant improvements to stability, perhaps unsurprisingly, given the low levels of traffic at the long tail of the ranking, where small differences in rank may, therefore, be meaningless (i.e., statistically insignificant). This effect had already been observed at the heyday of the Alexa ranking [8], and continues in Google Chrome traffic to this day [6]. This also serves as confirmation of Cloudflare's decision to deploy two different models for compiling the head and tail of their Radar ranking [12]. Longer data periods also lead to higher stability, and aggregation over 7 days (like Radar) or even 30 days (like CrUX and Tranco) may therefore be preferable. Also, in this light, while the finding by Ruth et al. that CrUX is more reliable [6] is likely due to Chrome's widespread visibility on Web traffic, these design decisions also play a part in improving CrUX's (and other rankings') accuracy.

In conclusion, (post-recursor) passive DNS traffic can be used to generate a representative domain or website popularity ranking. Despite its limitations, such as the lack of IP-level data, the data is sufficiently rich to generate reliable rankings, as it provides the necessary data for producing traffic volumes, correcting mechanisms, and service classes. Through our evaluation, we are able to confirm the benefits of the design decisions made by recent rankings, specifically the move towards bucketing and longer-term aggregation, although these may not fulfill all use cases, e.g., if more granular ranks are desired. Future directions for evaluating these rankings ideally involve ground-truth data [6], long-term evaluations [21], and an assessment of the impact on research results [7].

ACKNOWLEDGMENTS

We thank the TMA and TNSM reviewers for their valuable feedback. We thank the TNSM board for the opportunity to extend our TMA work. We thank SIE Europe for providing

access to the passive DNS data feed that enabled this work. This research is partially funded by the Research Fund KU Leuven, and by the Cybersecurity Research Program Flanders. This work has been partially supported by the French Ministry of Research projects PERSYVAL-Lab under contract ANR-11-LABX-0025-01, DiNS under contract ANR-19-CE25-0009-01 and Grenoble Alpes Cybersecurity Institute (ANR-15-IDEX-02). Victor Le Pochat is currently employed by the European Commission. The views and opinions expressed herein are personal and do not necessarily reflect those of the European Commission or other EU institutions.

REFERENCES

[1] N. Demir, M. Große-Kampmann, T. Urban, C. Wressnegger, T. Holz, and N. Pohlmann, "Reproducibility and Replicability of Web Measurement Studies," in *2022 ACM Web Conference*, WWW '22, 2022, pp. 533–544. DOI: 10.1145/3485447.3512214

[2] S. S. Ahmad, M. D. Dar, M. F. Zaffar, N. Vallina-Rodriguez, and R. Nithyanand, "Apophanies or Epiphanies? How Crawlers Impact Our Understanding of the Web," in *The Web Conference 2020*, WWW '20, 2020, pp. 271–280. DOI: 10.1145/3366423.3380113

[3] D. Zeber et al., "The Representativeness of Automated Web Crawls as a Surrogate for Human Browsing," in *The Web Conference 2020*, WWW '20, 2020, pp. 167–178. DOI: 10.1145/3366423.3380104

[4] J. Jueckstock et al., "Towards Realistic and Reproducible Web Crawl Measurements," in *2021 Web Conference*, WWW '21, 2021, pp. 80–91. DOI: 10.1145/3442381.3450050

[5] V. Le Pochat and W. Joosen, "Analyzing Cyber Security Research Practices through a Meta-Research Framework," in *16th Cyber Security Experimentation and Test Workshop*, CSET '23, 2023, pp. 64–74. DOI: 10.1145/3607505.3607523

[6] K. Ruth, D. Kumar, B. Wang, L. Valenta, and Z. Durumeric, "Toppling Top Lists: Evaluating the Accuracy of Popular Website Lists," in *22nd ACM Internet Measurement Conference*, IMC '22, 2022. DOI: 10.1145/3517745.3561444

[7] Q. Scheitle et al., "A Long Way to the Top: Significance, Structure, and Stability of Internet Top Lists," in *2018 Internet Measurement Conference*, IMC '18, 2018, pp. 478–493. DOI: 10.1145/3278532.3278574

[8] V. Le Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczyński, and W. Joosen, "Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation," in *26th Annual Network and Distributed System Security Symposium*, NDSS '19, 2019. DOI: 10.14722/ndss.2019.23386

[9] W. Aqeel, B. Chandrasekaran, A. Feldmann, and B. M. Maggs, "On Landing and Internal Web Pages: The Strange Case of Jekyll and Hyde in Web Performance Measurement," in *2020 ACM Internet Measurement Conference*, IMC '20, 2020, pp. 680–695. DOI: 10.1145/3419394.3423626

[10] T. Alby and R. Jäschke, "Analyzing the Web: Are Top Websites Lists a Good Choice for Research?" In *26th International Conference on Theory and Practice of Digital Libraries*, TPDL '22, 2022, pp. 11–25. DOI: 10.1007/978-3-031-16802-4_2

[11] D. Hubbard. "Cisco Umbrella 1 Million. "[Online]. Available: https://web.archive.org/web/20210502123151/https://umbrella.cisco.com/blog/cisco-umbrella-1-million

[12] C. Martinho and S. Zejnilovic. "Goodbye, Alexa. Hello, Cloudflare Radar Domain Rankings," The Cloudflare Blog. [Online]. Available: https://blog.cloudflare.com/radar-domain-rankings/

[13] A. Gee-Clough. "Mirror, Mirror, on the Wall, Who's the Fairest (website) of Them all?" DomainTools. [Online]. Available: https://www.domaintools.com/resources/blog/mirror-mirror-on-the-wall-whos-the-fairest-website-of-them-all

[14] "Chrome UX Report," Chrome Developers. [Online]. Available: https://developer.chrome.com/docs/crux/

[15] V. Le Pochat, "Reflecting on Research Practices," *Communications of the ACM*, vol. 67, no. 5, May 2024. DOI: 10.1145/3651965

[16] V. Le Pochat et al., "Evaluating the impact of design decisions on passive DNS-based domain rankings," in *8th Network Traffic Measurement and Analysis Conference*, TMA '24, 2024. DOI: 10.23919/tma62044.2024.10559182

[17] Q. Xie et al., "Building an Open, Robust, and Stable Voting-Based Domain Top List," in *31st USENIX Security Symposium*, USENIX Security '22, 2022, pp. 625–642. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/xie

[18] K. Ruth et al., "A World Wide View of Browsing the World Wide Web," in *22nd ACM Internet Measurement Conference*, IMC '22, 2022. DOI: 10.1145/3517745.3561418

[19] A. Medina, M. Allman, and S. Floyd, "Measuring the Evolution of Transport Protocols in the Internet," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 2, pp. 37–52, Apr. 2005. DOI: 10.1145/1064413.1064418

[20] B. Veal, K. Li, and D. Lowenthal, "New Methods for Passive Estimation of TCP Round-Trip Times," in *6th International Workshop on Passive and Active Network Measurement*, PAM '05, 2005, pp. 121–134. DOI: 10.1007/978-3-540-31966-5_10

[21] V. Le Pochat, T. Van Goethem, and W. Joosen, "Evaluating the Long-term Effects of Parameters on the Characteristics of the Tranco Top Sites Ranking," in *12th USENIX Workshop on Cyber Security Experimentation and Test*, CSET '19, 2019. [Online]. Available: https://www.usenix.org/conference/cset19/presentation/lepochat

[22] W. Rweyemamu, T. Lauinger, C. Wilson, W. Robertson, and E. Kirda, "Clustering and the Weekend Effect: Recommendations for the Use of Top Domain Lists in Security Research," in *20th International Conference on Passive and Active Measurement*, PAM '19, 2019, pp. 161–177. DOI: 10.1007/978-3-030-15986-3_11

[23] W. Rweyemamu, T. Lauinger, C. Wilson, W. Robertson, and E. Kirda, "Getting Under Alexa's Umbrella: Infiltration Attacks Against Internet Top Domain Lists," in *22nd International Conference on Information Security*, ISC '19, 2019, pp. 255–276. DOI: 10.1007/978-3-030-30215-3_13

[24] C. Xu, Y. Zhang, F. Shi, H. Ma, W. Ding, and H. Shan, "Gushing Resolvers: Measuring Open Resolvers' Recursive Behavior," in *4th International Conference on Advanced Information Science and System*, AISS '22, 2022. DOI: 10.1145/3573834.3574533

[25] L. Deri, S. Mainardi, M. Martinelli, and E. Gregori, "Exploiting DNS traffic to rank internet domains," in *2013 IEEE International Conference on Communications Workshops*, ICC '13, 2013, pp. 1325–1329. DOI: 10.1109/iccw.2013.6649442

[26] A. Mayrhofer, M. Braunöder, and A. Kaplan, "DNS Magnitude - A Popularity Figure for Domain Names, and its Application to L-root Traffic," nic.at GmbH, Tech. Rep., Aug. 5, 2020. [Online]. Available: https://www.nic.at/media/files/nic-report/dns-magnitude-paper-20200805.pdf

[27] J. L. García-Dorado, J. Ramos, M. Rodríguez, and J. Aracil, "DNS weighted footprints for web browsing analytics," *Journal of Network and Computer Applications*, vol. 111, pp. 35–48, Jun. 2018. DOI: 10.1016/j.jnca.2018.03.008

[28] C. E. Wills, M. Mikhailov, and H. Shang, "Inferring Relative Popularity of Internet Applications by Actively Querying DNS Caches," in *3rd ACM SIGCOMM Conference on Internet Measurement*, IMC '03, 2003, pp. 78–90. DOI: 10.1145/948205.948216

[29] M. Abu Rajab, F. Monrose, A. Terzis, and N. Provos, "Peeking through the Cloud: DNS-Based Estimation and Its Applications," in *6th International Conference on Applied Cryptography and Network Security*, ACNS '08, 2008, pp. 21–38. DOI: 10.1007/978-3-540-68914-0_2

[30] Z. Wang, "Analysis of DNS Cache Effects on Query Distribution," *The Scientific World Journal*, vol. 2013, pp. 1–8, 2013. DOI: 10.1155/2013/938418

[31] A. Shimoda et al., "Inferring Popularity of Domain Names with DNS Traffic: Exploiting Cache Timeout Heuristics," in *IEEE Global Communications Conference*, Globecom '15, 2015. DOI: 10.1109/glocom.2015.7417638

[32] P. Foremski, O. Gasser, and G. C. M. Moura, "DNS Observatory: The Big Picture of the DNS," in *2019 Internet Measurement Conference*, IMC '19, 2019, pp. 87–100. DOI: 10.1145/3355369.3355566

[33] S. Hao, A. Kantchelian, B. Miller, V. Paxson, and N. Feamster, "PREDATOR: Proactive Recognition and Elimination of Domain Abuse at Time-Of-Registration," in *2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, Oct. 24, 2016, pp. 1568–1579. DOI: 10.1145/2976749.2978317

[34] S. Hao, N. Feamster, and R. Pandrangi, "Monitoring the Initial DNS Behavior of Malicious Domains," in *2011 Internet Measurement Conference*, IMC '11, Nov. 2, 2011, pp. 269–278. DOI: 10.1145/2068816.2068842

[35] S. Fernandez, M. Korczyński, and A. Duda, "Early Detection of Spam Domains with Passive DNS and SPF," in *Passive and Active Measurement*, 2022, pp. 30–49. DOI: 10.1007/978-3-030-98785-5_2

[36] M. Lyu, H. H. Gharakheili, C. Russell, and V. Sivaraman, "Enterprise DNS Asset Mapping and Cyber-Health Tracking via Passive Traffic Analysis," *IEEE Trans. Netw. Serv. Manag.*, vol. 20, no. 3, pp. 3699–3716, 2023. DOI: 10.1109/TNSM.2022.3221981

[37] A. Dulaunoy, A. Kaplan, P. A. Vixie, and H. Stern, "Passive DNS - Common Output Format," Internet Engineering Task Force, Internet-Draft draft-dulaunoy-dnsop-passive-dns-cof-10, Jun. 2023, Work in Progress, 13 pp. [Online]. Available: https://datatracker.ietf.org/doc/draft-dulaunoy-dnsop-passive-dns-cof/10/

[38] "A Breakthrough European Data Sharing Collective to Fight Cybercrime," SIE Europe UG. [Online]. Available: https://www.sie-europe.net/

[39] O. van der Toorn, M. Müller, S. Dickinson, C. Hesselman, A. Sperotto, and R. van Rijswijk-Deij, "Addressing the challenges of modern DNS a comprehensive tutorial," *Computer Science Review*, vol. 45, p. 100 469, 2022. DOI: 10.1016/j.cosrev.2022.100469

[40] R. Edmonds, "ISC Passive DNS Architecture," Internet Systems Consortium, Inc., Tech. Rep., Mar. 2012. [Online]. Available: https://ftp.iij.ad.jp/pub/network/isc/kb-files/passive-dns-architecture.pdf

[41] G. C. M. Moura, J. Heidemann, M. Müller, R. de O. Schmidt, and M. Davids, "When the Dike Breaks: Dissecting DNS Defenses During DDoS," in *2018 Internet Measurement Conference*, IMC '18, 2018, pp. 8–21. DOI: 10.1145/3278532.3278534

[42] S. Hao, Y. Zhang, H. Wang, and A. Stavrou, "End-Users Get Maneuvered: Empirical Analysis of Redirection Hijacking in Content Delivery Networks," in *27th USENIX Security Symposium*, USENIX Security '18, 2018, pp. 1129–1145. [Online]. Available: https://www.usenix.org/system/files/conference/usenixsecurity18/sec18-hao.pdf

[43] A. Kashaf, V. Sekar, and Y. Agarwal, "Analyzing Third Party Service Dependencies in Modern Web Services: Have We Learned from the Mirai-Dyn Incident?" In *2020 Internet Measurement Conference*, IMC '20, 2020, pp. 634–647. DOI: 10.1145/3419394.3423664

[44] Y. Dimova, G. Acar, L. Olejnik, W. Joosen, and T. Van Goethem, "The CNAME of the Game: Large-scale Analysis of DNS-based Tracking Evasion," *Proceedings on Privacy Enhancing Technologies*, vol. 2021, no. 3, pp. 394–412, Apr. 2021. DOI: 10.2478/popets-2021-0053

[45] J. St Sauver. "What is a Bailiwick?" Farsight Security. [Online]. Available: https://www.farsightsecurity.com/blog/txt-record/what-is-a-bailiwick-20170321/

[46] S. Coulondre and B. Sitney, "A Stable and Open Method for Ranking Domains," Profound Networks, 2019. [Online]. Available: https://www.profound.net/pages/resources/DomainRank_Whitepaper.pdf

[47] J. Lehmann et al., "DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia," *Semantic Web*, vol. 6, no. 2, pp. 167–195, 2015. DOI: 10.3233/sw-140134

[48] "Public Suffix List," Mozilla Foundation. [Online]. Available: https://publicsuffix.org/

[49] L. A. Adamic and B. A. Huberman, "Zipf's Law and the Internet," *Glottometrics*, vol. 3, pp. 143–150, 2002. [Online]. Available: https://glottometrics.iqla.org/wp-content/uploads/2021/06/g3zeit.pdf

[50] A. Clauset, C. R. Shalizi, and M. E. J. Newman, "Power-Law Distributions in Empirical Data," *SIAM Review*, vol. 51, no. 4, pp. 661–703, Nov. 2009. DOI: 10.1137/070710111

[51] Y. Chen, M. Antonakakis, R. Perdisci, Y. Nadji, D. Dagon, and W. Lee, "DNS Noise: Measuring the Pervasiveness of Disposable Domains in Modern DNS Traffic," in *44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, DSN '14, 2014, pp. 598–609. DOI: 10.1109/dsn.2014.61

[52] V. Le Pochat et al., "A Practical Approach for Taking Down Avalanche Botnets Under Real-World Constraints," in *27th Annual Network and Distributed System Security Symposium*, NDSS '20, 2020. DOI: 10.14722/ndss.2020.24161

[53] L. Invernizzi, K. Thomas, A. Kapravelos, O. Comanescu, J. Picod, and E. Bursztein, "Cloak of Visibility: Detecting When Machines Browse a Different Web," in *IEEE Symposium on Security and Privacy*, SP '16, 2016, pp. 743–758. DOI: 10.1109/sp.2016.50

[54] D. Chiba, H. Nakano, and T. Koide, "DomainHarvester: Uncovering Trustworthy Domains Beyond Popularity Rankings," *IEEE Access*, vol. 13, pp. 28 167–28 188, 2025. DOI: 10.1109/access.2025.3539882

[55] T. Van Goethem, N. Miramirkhani, W. Joosen, and N. Nikiforakis, "Purchased Fame: Exploring the Ecosystem of Private Blog Networks," in *2019 ACM Asia Conference on Computer and Communications Security*, Asia CCS '19, 2019, pp. 366–378. DOI: 10.1145/3321705.3329830

[56] C. Xu, Y. Zhang, F. Shi, Y. Wang, J. Zheng, and M. Hu, "RaMOF: Ranking Manipulation Leveraging Open Forwarders," *Tsinghua Science and Technology*, 2024. DOI: 10.26599/TST.2024.9010211

[57] D. Senie and P. Ferguson, *Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing*, Rfc 2827, May 2000. DOI: 10.17487/rfc2827 [Online]. Available: https://www.rfc-editor.org/info/rfc2827

[58] Q. Lone, M. Korczyński, M. van Eeten, and C. H. Gañán, "SAVing the Internet: Explaining the Adoption of Source Address Validation by Internet Service Providers," in *20th Annual Workshop on the Economics of Information Security*, WEIS '20, 2020. [Online]. Available: https://weis2020.econinfosec.org/wp-content/uploads/sites/8/2020/06/weis20-final31.pdf

[59] Q. Lone, A. Frik, M. Luckie, M. Korczyński, M. van Eeten, and C. Gañán, "Deployment of Source Address Validation by Network Operators: A Randomized Control Trial," in *2022 IEEE Symposium on Security and Privacy*, SP '22, 2022, pp. 703–720. DOI: 10.1109/sp46214.2022.00041

[60] M. Korczyński, Y. Nosyk, Q. Lone, M. Skwarek, B. Jonglez, and A. Duda, "Don't Forget to Lock the Front Door! Inferring the Deployment of Source Address Validation of Inbound Traffic," in *21st International Conference on Passive and Active Measurement*, PAM '20, 2020, pp. 107–121. DOI: 10.1007/978-3-030-44081-7_7

[61] N. Carlini et al., "Poisoning Web-Scale Training Datasets Is Practical," in *2024 IEEE Symposium on Security and Privacy*, SP '24, 2024, pp. 407–425. DOI: 10.1109/sp54263.2024.00179

[62] G. C. M. Moura, J. Heidemann, R. d. O. Schmidt, and W. Hardaker, "Cache Me If You Can: Effects of DNS Time-to-Live," in *2019 Internet Measurement Conference*, IMC '19, 2019, pp. 101–115. DOI: 10.1145/3355369.3355568

[63] C. Lever, R. J. Walls, Y. Nadji, D. Dagon, P. D. McDaniel, and M. Antonakakis, "Domain-Z: 28 Registrations Later Measuring the Exploitation of Residual Trust in Domains," in *IEEE Symposium on Security and Privacy*, SP '16, 2016, pp. 691–706. DOI: 10.1109/sp.2016.47

[64] W. B. de Vries, Q. Scheitle, M. Müller, W. Toorop, R. Dolmans, and R. van Rijswijk-Deij, "A First Look at QNAME Minimization in the Domain Name System," in *20th International Conference on Passive and Active Measurement*, PAM '19, 2019, pp. 147–160. DOI: 10.1007/978-3-030-15986-3_10

[65] J. Magnusson, M. Müller, A. Brunstrom, and T. Pulls, "A Second Look at DNS QNAME Minimization," in *24th International Conference on Passive and Active Measurement*, PAM '23, 2023, pp. 496–521. DOI: 10.1007/978-3-031-28486-1_21

[66] C. Lu et al., "An End-to-End, Large-Scale Measurement of DNS-over-Encryption: How Far Have We Come?" In *2019 Internet Measurement Conference*, IMC '19, 2019, pp. 22–35. DOI: 10.1145/3355369.3355580

[67] I. Sanchez-Rola, D. Balzarotti, C. Kruegel, G. Vigna, and I. Santos, "Dirty Clicks: A Study of the Usability and Security Implications of Click-Related Behaviors on the Web," in *The Web Conference 2020*, WWW '20, 2020, pp. 395–406. DOI: 10.1145/3366423.3380124

[68] J. St Sauver. "Finding Top FQDNs Per Day in DNSDB Export MTBL Files (Part One of a Three Part Series)," Farsight Security. [Online]. Available: https://www.farsightsecurity.com/blog/txt-record/TopFQDNs-20190322/

[69] T. Wicinski, *DNS Privacy Considerations*, RFC 9076, Jul. 2021. DOI: 10.17487/rfc9076

[70] B. Imana, A. Korolova, and J. Heidemann, "Enumerating Privacy Leaks in DNS Data Collected above the Recursive," in *2018 NDSS DNS Privacy Workshop*, 2018. [Online]. Available: https://www.isi.edu/~johnh/PAPERS/Imana18a.pdf

[71] B. Imana, A. Korolova, and J. Heidemann, "Institutional Privacy Risks in Sharing DNS Data," in *2021 Applied Networking Research Workshop*, ANRW '21, 2021, pp. 69–75. DOI: 10.1145/3472305.3472324

[72] J. M. Spring and C. L. Huth, "The Impact of Passive DNS Collection on End-user Privacy," in *2nd Workshop on Securing and Trusting Internet Names*, SATIN '12, 2012. [Online]. Available: https://resources.sei.cmu.edu/asset_files/WhitePaper/2012_019_001_57023.pdf