A Deep Dive into DNS Query Failures

Donghui Yang, *Institute of Computing Technology, Chinese Academy of Sciences;* Zhenyu Li, *Institute of Computing Technology, Chinese Academy of Sciences, and Purple Mountain Laboratories;* Gareth Tyson, *Queen Mary University of London*

https://www.usenix.org/conference/atc20/presentation/yang

This paper is included in the Proceedings of the 2020 USENIX Annual Technical Conference.

July 15–17, 2020

978-1-939133-14-4

Open access to the Proceedings of the 2020 USENIX Annual Technical Conference is sponsored by USENIX.

A Deep Dive into DNS Query Failures

Donghui Yang[†] [§], Zhenyu Li[†] [§] [‡], Gareth Tyson[♭]
[†]*ICT-CAS,* [§]*University of Chinese Academy of Sciences,* [‡] *Purple Mountain Laboratories,* [♭]*QMUL*

Abstract

The Domain Name System (DNS) is fundamental to the operation of the Internet. Failures within DNS can have a dramatic impact on the wider Internet, most notably preventing access to any services dependent on domain names (*e.g.* web, mobile apps). Although there have been several studies into DNS utilization, we argue that greater focus should be placed on understanding *how* and *why* DNS queries fail in-the-wild. In this paper, we perform the largest ever study into DNS activity, covering 3B queries. We find that 13.5% of DNS queries fail, and this leads us to explore the root causes. We observe significant differences between IPv4 and IPv6 lookups. A handful of domains that have high failure rates attract a huge volume of queries, and thus dominate the failures. This is particularly the case for domains that are classified as malicious. The success rates also vary greatly across resolvers due to the differences in the domains that they serve and the infrastructure reliability.

1 Introduction

The Domain Name System (DNS) is organized as a distributed system that provides mappings between human-readable domain names (*e.g.* foo.com) and their associated DNS records [17–20]. These include A type records for IPv4 addresses, AAAA for IPv6 addresses, MX for SMTP mail exchanges, NS for name servers, PTR for pointers of reverse DNS lookups and CNAME for domain name aliases. Nearly all Internet-connected applications depend on DNS. As such, it is a critical dependency, whose failure has the potential to create global Internet outages. For example, in 2016, Dyn (a DNS operator) suffered a major Denial of Service attack against its infrastructure. This meant that DNS queries for popular domains such as Netflix and Visa began to fail, crippling access to these services (even though those services were still online).

Although there is a significant body of research into DNS behavior, we argue that *DNS failures* specifically require further investigation. Root DNS server behavior was examined in [5, 8], where negative DNS answers were analyzed including NXDOMAIN responses. Callahan *et al.* [4] also examined the DNS behavior from the perspectives of performance and response message. We differ in that our focus is on failures not caused by NXDOMAINs. In addition, we note that DNS has evolved significantly since these studies, *e.g.* the rise of new gTLDs and IDNs. Although there have been a number of studies of new gTLDs and IDNs [9, 13, 15], they mainly focus on domain registration behavior and cyber attacks, while we complement these studies with DNS query failure analysis. Other work [12, 22, 25] has leveraged NXDOMAIN responses to detect botnets or DGAs. Again, our work focuses on failures caused by DNS infrastructures instead of NXDOMAINs.

With the above in-mind, we present a large-scale analysis of DNS query failures in-the-wild. To achieve this, we gather a unique dataset containing 3B DNS queries (Section 2). We find that failed queries are, indeed, common place with 13.5% of all queries not successfully resolved. This motivates us to inspect which factors correlate most closely with failed queries (Section 3). We observe a highly skewed distribution, whereby a small number of domains are responsible for the majority of failures. AAAA queries (IPv6) are particularly unreliable, with only 1/3 of queries successfully resolved. This is perhaps understandable given the use of protocols such as Happy Eyeballs [24], although we also find that many domains lack AAAA support. We further inspect the relationship between failures and the DNS resolver used, to find a vast array of resolvers, with 13.5% of queries in our dataset being issued to public resolvers, *e.g.* OpenDNS. We observe diverse failure rates across the resolvers, confirming that they do have an impact on failures. We also note differences among the Top Level Domains (TLDs) with, for example, the new wave of generic-TLDs having higher failure rates than more traditional TLDs. Finally, we propose system implications based on our findings. To sum up, we make the following key findings:

- In spite of the promotion of IPv6 over recent years [7, 21, 26], the majority (86.2%) of DNS queries are still for A records, while only 10.4% are for AAAA records

(comparable to the proportion in 2012 [8]). The failure rate for A lookups is 6.9%, yet, to our surprise, the failure rate for AAAA queries is as high as 64.2%, almost 3 times of that in 2012 [8]. We observe that approximately 60% of domains do not support AAAA queries.

- We explore a number of factors to explore the causes of failures. We observe a heavy-tailed distribution of failures across domains, indicating that a handful of domains contribute to most of the failures. 20% of local resolvers in our dataset have never successfully resolved AAAA queries, implying they are not ready for IPv6. The use of public resolvers may also impact query failures as they show diverse failure rates, in-part due to the distinct domain sets that each serves and the differences in infrastructure reliability.
- The success rates for new gTLD domains and IDNs are 10% lower than that of well established domains. This is largely because of the prevalence of malicious domains. Certain ASes are prevalent for hosting malicious new gTLD domains, and these new gTLD domains contribute to 73.7% of the all new gTLD queries. The malicious domains are, however, volatile and change frequently. In fact, *none* are resolvable today. The malicious new gTLD domains in these ASes are of various types and have distinct network footprints.

2 Dataset

Dataset Oveview. Our dataset consists of passive DNS logs that are generated by Deep Packet Inspection (DPI) appliances in 3 ISPs in China. Each DPI appliance parses the DNS response messages from recursive resolvers to end users, and generates a log for each response. A log includes the end user's anonymized IP address, BGP prefix,[1] the ASN (Autonomous System Number), the recursive resolver's IP address, the DNS query type, all the resource records, the timestamp (in seconds) and an indicator about whether the resolver and the end user's original IP address share a common /24 prefix. In cases of CNAME responses, we follow the redirection to the final record. The dataset contains 14 samples that were collected every other day in February 2018. Each sample consists of 10-minute logs generated by all the DPI appliances of the 3 ISPs. In total, we obtain 3,085,998,589 logs. It is noteworthy that while there were IPv6 addresses in the response IP list of AAAA queries, no IPv6 addresses were seen in end users' IP addresses and recursive DNS resolvers.

Identification of Failed Queries. We next extract the set of failed queries for the four most popular types of records (A, AAAA, PTR, MX), because they constitute 99.5% of all queries. For each response, we extract the requested domain (the QNAME) from the Question portion, and check if the response contains a valid answer (*e.g.* for an A query, at least one RR in the response is an A record of the requested domain). In this paper, we are interested in failures caused by DNS infrastructures instead of NXDOMAINs (*e.g.* typos). However, we do not have the response code (*e.g.* 'NOERROR', 'NXDOMAIN' or other status) in our dataset. Therefore, we turn to a heuristic method to filter out logs that are attributed to NXDOMAINs. Specifically, for each domain requested (*i.e.,*QNAME in the log), we check if it has succeeded at least once in our logs. We then remove the logs containing domains that have never succeeded in the whole dataset, as they are likely NXDOMAINs. The subsequent analyses are based on the remaining dataset, which contains 2,811,010,890 logs issued by 37,070,965 unique IP addresses to 246,991 resolvers.

Caveats. It is important to highlight potential limitations in our data. The above heuristic method may leave some domains that were resolvable at a time but then became NXDOMAINs later. Moreover, our dataset does not allow us to inspect failed queries that did not trigger a response (*e.g.* due to packet loss). Naturally, the fact that a DNS response is returned does not necessarily mean that the web server is live and responsive. Therefore, we only inspect if a valid DNS response is returned (not if the IP address is correct). There are many possibilities that lead to incorrect mapping of domains to addresses, such as DNS manipulation [16] and on-path DNS interception [14]. Another related concern pertains to censored domains. Thus, before continuing, we test if a censored domain will return a valid IPv4/IPv6 address for an A/AAAA query. Our tests confirm that, indeed, valid addresses are returned, even when querying censored websites.[2] Another potential limitation is that our data is local to China. Nevertheless, we believe the scale of the Chinese Internet means that these findings can still have a major impact. As DNS is a globally distributed system where China and other countries are all involved, there is not much specific to China from the perspective of the DNS infrastructure. We also note that (to the best of our knowledge) this is by far the largest DNS failure dataset ever studied.

Ethical Issues. The ISPs collect the DNS logs for the purpose of improving their service quality and security. The end users' IP addresses were anonymized and we are unable nor allowed to link queries to users. Users are notified when subscribing that the ISPs may collect this information, and may share it with academics for research. Our study has not triggered the collection of any new data. All data was processed in a secure silo by the first author.

3 Exploring DNS Query Failures

3.1 A Primer on DNS Failures

We begin by simply computing the number and types of failed queries in our dataset. Table 1 shows the percentage of query

[1] IP addresses and BGP prefixes are anonymized with `Crypto-PAn` [2]

[2] Since we focus on DNS query failures, we did not check whether the returned IP address does host the queried domain or not [1].

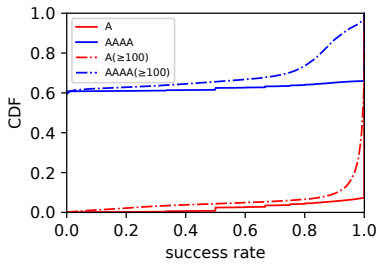

Figure 1: CDFs of success rates of domains for A and AAAA queries.

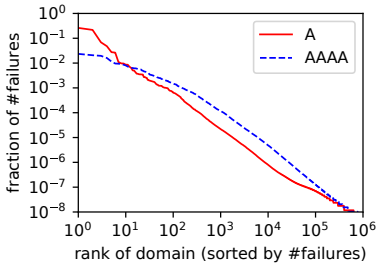

Figure 2: #failures - rank of domains(log-log).

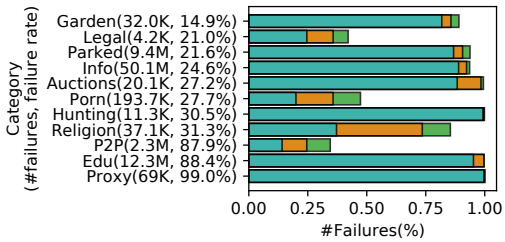

Figure 3: The top 11 domain categories sorted by failure rates with # of failures $>$ 1K.

types, alongside the *overall* success rate, which is 1 minus the ratio of failed queries to all queries of each query type. We present the four most popular query types. The A type queries account for the majority (86.2%), and AAAA queries have a smaller query volume (10.4%).[3] We observe a variety of success rates across the query types. Overall, the A queries are successfully resolved most frequently, while other query types manifest lower success rates. This confirms that a sizeable fraction of queries are *not* successfully resolved; we spend the rest of the section exploring factors influencing this trend.

Table 1: Four popular types of DNS queries: percentages of the number of queries and success rates.

Query Type	A	AAAA	PTR	MX	Others
#queries	86.2%	10.4%	2.8%	0.1%	0.5%
Success Rate	93.1%	35.8%	40.4%	82.9%	-

3.2 Failures Across Domains

We first compute the distribution of failures across domains. Due to their prominence, we focus on A and AAAA queries. Figure 1 presents the CDFs of the success rates across domains encountered within our dataset.

A Queries. A queries exhibit high success rates: overall, 93.7% of domains have a success rate exceeding 95% suggesting high reliability. There are ouliers though; for instance, the bottom 0.1% of domains have success rates below 5%. To eliminate the impact of low-frequency domains on the results, we filter out domains that issue fewer than 100 requests and plot the CDF of success rate of the remaining domains (the red dash-dot line), where 84.9% of remaining domains have a success rate exceeding 95%. Nevertheless, as many as 7% of domains experience a success rate lower than 50%. Given that we have removed the failures caused by NXDOMAINs, the result suggests that problems with the DNS infrastructure do impact users when visiting these domains.

AAAA Queries. For AAAA queries, only 34.3% of domains have a success rate exceeding 95%. When limiting to domains whose query frequency exceeds 100, only 7.8% of domains have a success rate exceeding 95%, while about 60% of domains have never been successfully resolved. Again, given that we only include domains that have been successfully resolved (considering all query types), this suggests that there are infrastructural limitations in how DNS supports IPv6.

Domain Failure Rates The above suggests that the majority of failures are the responsibility of a small set of domains, especially for A queries. To explore this further, Figure 2 presents the number of failures per domain on a log-log plot. We sort the X-axis by the rank of the domain (based on the number of failures). We see that failures are concentrated on a small number of domains. To gain a further understanding of the types of domains that have high failure rates, we utilize the Webroot Brightcloud API[4] to classify the top 50K domains (measured by failure rate). Figure 3 presents the results. The Y-axis shows the top 11 categories sorted by failure rate, where the number of failed queries ($>$ 1K) and the failure rate is shown in the parentheses. For each category, we plot the ratio of the number of failed queries of the top 3 SLDs to the total number of failures. A number of classifications which can be expected to fail frequently are present, *e.g.* proxy, porn and parked domains. This suggests that such domain types are paramount in increasing failure rates. However, it is unexpected to see the Education category is ranked second. Closer inspection reveals that *clock.cuhk.edu.hk*, which is the third most failed domain, contributes the most failures. Another interesting observation is the concentration of failures for each category on a few domains. For 8 out of the 11 categories, over 80% of the failures are attributed to the top 3 SLDs (top 1 SLD in most cases).

3.3 Failures Across Resolvers

Another explanation for failed queries is that the resolvers may not correctly handle queries.

Testing Resolvers. To explore this, we calculate the success rate of queries issued to each DNS resolver (identified by the resolver's IP address). Figure 4 presents the CDF of the success rate for the domains per resolver.[5] The majority of

[3] Note that we follow CNAME redirections, rather than reporting them here as responses.

[4] https://www.brightcloud.com/web-service

[5] We eliminate the resolvers serving fewer than 100 queries in our dataset to avoid bias.

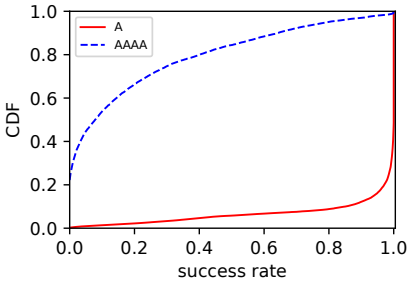

Figure 4: CDF of the success rate for individual resolvers.

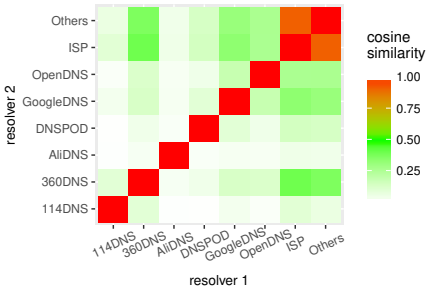

Figure 5: Cosine similarity between each pair of DNS resolver types.

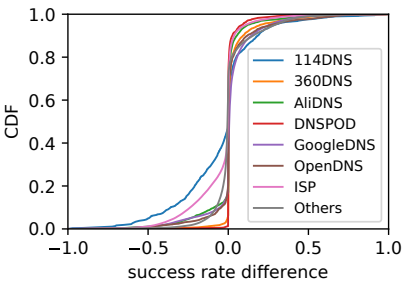

Figure 6: Different success rates of same domains handled by different resolvers.

Table 2: The number of A and AAAA queries and their success rates (shown in the parentheses) handled by each resolver.

	114DNS	360DNS	AlibabaDNS	DNSPOD	GoogleDNS	OpenDNS	ISP	Others
A	296.4K(98.5%)	831.0K(95.9%)	667.8K(94.7%)	352.5K(99.6%)	333.4M(90.7%)	467.6K(86.3%)	48.7M(95.3%)	2.1B(93.5%)
AAAA	75.4K(14.5%)	50.3K(61.8%)	112.9K(52.4%)	15.5K(54.3%)	40.6M(43.4%)	31.0K(49.2%)	9.6M(22.8%)	252.6M(35.0%)

resolvers have very high success rates when serving A queries: about half experience almost no failures. By contrast, 60% of resolvers serving AAAA queries can successfully resolve just 20% of the queries. Surprisingly, about 20% of the resolvers never succeed in resolving AAAA queries. These resolvers may not be IPv6 ready during our observation period.

Testing Public Resolvers. Closer inspection reveals that a notable set of queries are sent to *public resolvers* [3,6,10]. Hence, we also inspect the reliability of these public infrastructures, which include 114 (Chinese) DNS resolvers provided by multiple telecom operators, DNS Pai which belongs to Qihoo 360, AliDNS which belongs to Alibaba, and DNSPOD which belongs to Tencent. We also take into account GoogleDNS and OpenDNS which are widely used throughout the world. We identify these public DNS resolvers by the IP addresses offered on their official websites. Table 2 shows the number of A and AAAA queries handled by each public DNS resolver mentioned above along with their success rates. GoogleDNS dominates the most used public DNS service (even though Google is less well known in China). Others do not show much difference in terms of query volume. We observe various success rates across public DNS resolvers though. For example, DNSPOD succeeds in almost all its A queries, while OpenDNS achieves just 86.3%. There is also notably lower success rate across all resolvers for AAAA queries.

The above confirms that resolvers do seem to have an impact on success rates. A potential reason for this is that the resolvers may simply receive different queries. To explore this, we compute a vector for each resolver, where each element represents a domain, which appears in A or AAAA queries using the resolver, and the query volume of that domain handled by the resolver. Then we calculate the similarity of each pair of DNS resolver using the cosine similarity between their vectors. Figure 5 illustrates the result. The resolvers actually demonstrate a surprisingly low similarity with each other, signalling rather different request patterns. Among these resolvers, 114DNS and AliDNS are the least like the others. Indeed, 114DNS handles many requests for Akamai domains which appear less often in other resolvers, while AliDNS handles many requests of taobao.com and alipay.com which belongs to Alibaba services. This could be the reason for the variance of success rates observed from different resolvers.

Another possible explanation is the differences between resolvers' infrastructures. To explore this, we compare the success rates of the same domains handled by different resolvers. Specifically, for each resolver, we first find the domain intersection of it and each other resolver. Then for each domain in each intersection, we calculate the difference in their success rates on the two resolvers. Finally, for each type of resolver, we plot the CDF of the differences between this type to other types in Figure 6. Note, a difference below 0 indicates a lower success rate of this type of resolver, and a positive value indicates a higher success rate. We can see significant differences for some types of resolvers: domains resolved by 114DNS and ISP are more likely to fail, while DNSPOD and 360DNS have higher success rates. This observation partially explains the results in Table 2.

3.4 Failures Across TLDs

We next inspect if certain TLDs have lower query success rates. Specifically, we explore two camps of TLDs: the new generic Top Level Domains (gTLD),[6] and those that have Internationalized Domain Name (IDNs). Our dataset contains 611,769 new gTLDs and 79,705 IDNs. Table 3 summarizes our results. We see rather different rates of success across the domain and query types. The lower success rate for new gTLDs may be because such gTLDs attract certain types of domain registrant. For example, the .lol domain is well known to attract large volumes of malicious activites [11]. With this in-mind, we find a success rate of just 20.3% for

[6]Based on the list from nTLDStats https://ntldstats.com

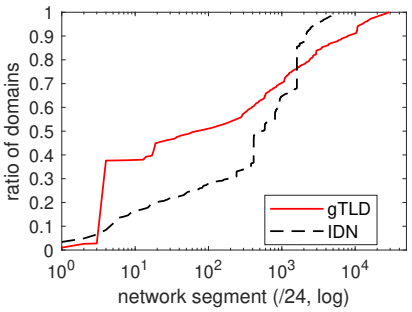

Figure 7: Distri. of new gTLD domains (gTLD in the legend for short) and IDNs seperated by /24 network segments.

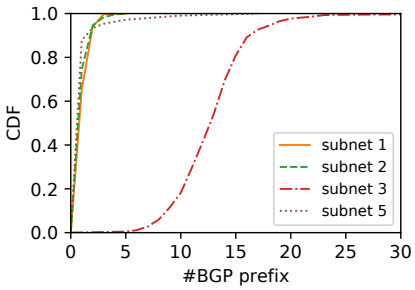

Figure 8: CDF of # BGP prefixes requesting individual malicious SLDs.

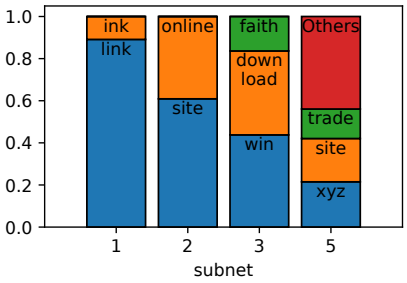

Figure 9: TLDs and their fractions of # malicious SLDs.

.lol, compared to 83.0% of .com. Another reason for failed queries is the presence of malicious domains which are unreliable. To inspect this, we extract all A record queries that were successfully resolved and investigate the network segments hosting them. Overall, we obtain 113,539 (11,729) IP addresses hosting new gTLDs (IDNs) mapping to 29,047 (5,635) /24 network segments, respectively.

Table 3: The number of queries and success rates (shown in the parentheses) of new gTLD domains and IDNs.

	new gTLD	IDN
total	4.0M (79.3%)	0.26M (66.6%)
A	3.4M (88.6%)	0.17M (86.7%)
AAAA	0.6M (25.9%)	0.09M (26.4%)

Figure 7 presents the distribution of the number of new gTLD domains and IDNs per network segment (sorted by #IPs hosting new gTLD domains and IDNs, respectively). Several surges in both lines further arouse our attention: they indicate the existence of some /24 network segments that serve a large number of new gTLD domains or IDNs. This is naturally driven by the presence of large web hosting providers.

Thus, we extract the top 5 surges for both new gTLD domains and IDNs to explore their details. Due to space limitation, we only present the results of new gTLD domains in Table 4, where the last column presents the number of domains that are resolvable on 26 Sept. 2019. Across all of these top ASes we witness an extremely low rate of successful resolutions. In the most extreme case, we observe 201K queries for 195K domains being mapped to Enzu, none of which are resolvable today. In addition, the number of queries is close to the number of FQDNs, suggesting that these domains are short-lived and change frequently. The above trends lead us to hypothesize that some of these domains may be associated with malicious activities. To explore this, we leverage two blacklists from VirusTotal and Qihoo 360 to check the domains. We label a domain as malicious if any of these two blacklists classify it as so. Due to the large volume of domains, we only check SLDs (as opposed to FQDNs). The results are listed in parentheses in Table 4. None of the IDNs are classified as malicious by the two blacklists, however, a *significant* fraction of the new gTLD domains fall into this category. For example, 80% of the SLDs hosted in 23.245.136.0/24 prefix (first row) are classified as malicious. This is common across all of the top new gTLD domains. In total, 73.7% of the queries are for the malicious domains.

Figure 8 presents the distribution of the number of end users' BGP prefixes requesting each malicious SLD in Subnet 1, 2, 3, 5 (shown in Table 4). Except the SLDs hosted in subnet 3, other malicious SLDs affect only 1 or 2 networks. In contrast, malicious SLDs hosted in subnet 3 have a footprint in dozens of networks, implying a larger impact. One possible explanation is that the subnets host different sites. Hence, Figure 9 presents the make-up of the 5 subnets, confirming that they do map to different TLDs.

4 Implications on Systems Design

In this section, we discuss about the implications of our findings on system design, *i.e.,*what systems we could build based on our observations.

Active Measurement System. Our results show that although IPv6 has been promoted in recent years, AAAA queries still fail frequently, and there exist resolvers that do not support AAAA queries. In order to understand which resolvers support AAAA queries, we can build a system to actively measure the IPv6 support of the resolvers. For instance, similar to [23], we can build a single-node measurement system for monitoring IPv6 support of DNS resolvers. The system can distinguish between resolvers that support and do not support AAAA queries by sending DNS queries of popular domains that support AAAA queries. We can also test whether a domain supports AAAA queries by sending requests of this domain to resolvers that are classified as supporting AAAA queries. Considering the differences between resolvers, we could measure the success rates of domains by sending queries to different resolvers, and use the result to help choose the better resolvers.

In addition, we could compare different resolvers from the perspective of localization, *i.e.,*whether the resolver directs

Table 4: new gTLD domains hosted by the top 5 subnets. The number of domains labeled as malicious are within the parentheses.

No.	subnet	AS num.	AS name	#IPs	#queries	#FQDN	#SLD	#resolvable
1	23.245.136.0/24	18978	Enzu Inc	252	201.9K (157.1K)	195.9K (152.2K)	483 (386)	0(0)
2	192.238.167.0/24	395954	Leaseweb	236	17.4K (14.8K)	16.3K (13.9K)	287 (243)	0 (0)
3	172.246.207.0/24	18978	Enzu Inc	236	15.7K (15.4K)	13.2K (13.0K)	443(434)	1 (1)
4	104.217.93.0/24	40676	Psychz Net	253	9.0K (1)	8.8K (1)	923 (1)	9(0)
5	47.89.58.0/24	45102	Alibaba	4	10.9K (469)	8.8K (114)	7.7K (107)	748 (7)

Table 5: The localization performance of resolvers: the proportion of queries directed to servers in the same AS as the end user when possible.

114DNS	360DNS	AlibabaDNS	DNSPOD	GoogleDNS	OpenDNS	ISP	Others
91.2%	98.0%	95.9%	94.3%	64.6%	43.8%	71.7%	69.9%

users to remote servers. This additional function is motivated by our observation presented in Table 5: We calculate for each resolver the fraction of queries that redirect clients to servers in the same AS (when possible). For each domain i, we count all its response IP addresses in the entire dataset and these IP addresses form a set S_i. Then for each log whose QNAME equals i, if at least one IP address in S_i is in the same AS as the end user, we label this query as "possible to be served locally"; if at least one IP address in the response IP addresses of this log is in the same AS as the end user, we label this query as "served locally". These two labels are independent and a log can have both labels, one of the labels, or no label. We aggregate the logs according to the resolvers, and calculate the ratio of *the number of logs labeled as served locally* to *the number of logs labeled as possible to be served locally* for each resolver. We observe that the obtained ratio differs significantly across the resolvers, which indicates that users can choose appropriate resolvers for better network performance. Therefore, it is useful to develop a system for end users to measure the localization performance of different resolvers.

Such an active measurement system is useful for content publishers, ISPs and end users. Many CDNs are being upgraded for better IPv6 support, however, if AAAA queries frequently fail, then the content publisher should be careful to use such CDN IPv6 service. Therefore, it is useful for publishers to locate their content if they can understand in advance which resolvers do not support AAAA queries. In addition, ISPs could also benefit when considering IPv6 network expansion, because understanding which domains support AAAA queries is useful for estimating the IPv6 traffic. For users, the measurement of different resolvers can help them to choose more suitable resolvers considering both IPv6 support and localization performance.

Malicious New gTLD Domain Detection System. We have found that malicious SLDs (of new gTLD domains) hosted by particular ASes contribute to higher failure rates. Manual inspection further reveals that the length of the SLDs tend to be short. Table 6 presents the fraction of malicious SLDs of different length. In more traditional TLDs, malicious domains are usually long because registering a short domain name would cost too much for an attacker. However, registering short new gTLD domains is much easier. Therefore, extracting features from domain names may not work well for detecting malicious new gTLD domains. We could use features like DNS query frequency, the number of FQDNs of an SLD, the resolved IP addresses and the corresponding ASes to build a system for detecting malicious SLDs of new gTLD domains.

Table 6: Fraction of malicious SLDs of different lengths.

Length	3	4	5	≥6
% of SLDs	0.1%	93.0%	6.1%	0.8%

5 Conclusion

The paper has presented a deep dive into DNS query failures using over 3B queries. We have identified high failure rates: 6.9% of A and 64.2% of AAAA queries. IPv6 is far from ready as over half of the domains and 20% of local resolvers do not support AAAA queries. Upgrading these resolvers and popular domains for IPv6 is the first step towards the wider usage of IPv6. Internet users, on the other hand, should be aware of the impact of using public resolvers, from both the perspectives of query failures and mapping inaccuracy [6]. We also found that the volatility of malicious domains (particularly new gTLD domains and IDNs) contributes to higher failure rates because they change frequently and accesses to them results in failures. The corresponding SLDs of malicious new gTLD domains and IDNs, however, are limited, and they are likely hosted by particular ASes. We finally proposed two potential systems that could build on our findings.

Acknowledgement

This work was supported in part by National Key RD Program of China: 2018YFB1800201, the NSF of China (NSFC): 61725206, the Youth Innovation Promotion Association CAS. The corresponding author is Zhenyu Li.

References

[1] Towards a comprehensive picture of the great firewall's DNS censorship. In *4th USENIX Workshop on Free and Open Communications on the Internet (FOCI 14)*, San Diego, CA, 2014. USENIX Association.

[2] Crypto-pan. https://www.cc.gatech.edu/computing/Networking/projects/cryptopan/, 2018.

[3] Bernhard Ager, Wolfgang Mühlbauer, Georgios Smaragdakis, and Steve Uhlig. Comparing DNS resolvers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 15–21. ACM, 2010.

[4] Thomas Callahan, Mark Allman, and Michael Rabinovich. On modern DNS behavior and properties. *ACM SIGCOMM Computer Communication Review*, 43(3):7–15, 2013.

[5] Sebastian Castro, Duane Wessels, Marina Fomenkov, and Kimberly Claffy. A day at the root of the internet. *ACM SIGCOMM Computer Communication Review*, 38(5):41–46, September 2008.

[6] Fangfei Chen, Ramesh K Sitaraman, and Marcelo Torres. End-user mapping: Next generation request routing for content delivery. *ACM SIGCOMM Computer Communication Review*, 45(4):167–181, 2015.

[7] Jakub Czyz, Mark Allman, Jing Zhang, Scott Iekel-Johnson, Eric Osterweil, and Michael Bailey. Measuring IPv6 adoption. In *ACM SIGCOMM Computer Communication Review*, volume 44, pages 87–98. ACM, 2014.

[8] Hongyu Gao, Vinod Yegneswaran, Yan Chen, Phillip Porras, Shalini Ghosh, Jian Jiang, and Haixin Duan. An empirical reexamination of global DNS behavior. In *ACM SIGCOMM Computer Communication Review*, volume 43, pages 267–278. ACM, 2013.

[9] Tristan Halvorson, Matthew F Der, Ian Foster, Stefan Savage, Lawrence K Saul, and Geoffrey M Voelker. From. academy to. zone: An analysis of the new TLD land rush. In *Proceedings of the 2015 Internet Measurement Conference*, pages 381–394. ACM, 2015.

[10] Cheng Huang, David A Maltz, Jin Li, and Albert Greenberg. Public DNS system and global traffic management. In *2011 Proceedings IEEE INFOCOM*, pages 2615–2623. IEEE, 2011.

[11] Damilola Ibosiola, Ignacio Castro, Gianluca Stringhini, Steve Uhlig, and Gareth Tyson. Who watches the watchmen: Exploring complaints on the web. *Web Conference*, 2019.

[12] N. Jiang, J. Cao, Y. Jin, L. E. Li, and Z. Zhang. Identifying suspicious activities through DNS failure graph analysis. In *The 18th IEEE International Conference on Network Protocols*, pages 144–153, Oct 2010.

[13] Maciej Korczynski, Maarten Wullink, Samaneh Tajalizadehkhoob, Giovane Moura, Arman Noroozian, Drew Bagley, and Cristian Hesselman. Cybercrime after the sunrise: A statistical analysis of DNS abuse in new gTLDs. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pages 609–623. ACM, 2018.

[14] Baojun Liu, Chaoyi Lu, Haixin Duan, Ying Liu, Zhou Li, Shuang Hao, and Min Yang. Who is answering my queries: Understanding and characterizing interception of the DNS resolution path. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 1113–1128, 2018.

[15] Baojun Liu, Chaoyi Lu, Zhou Li, Ying Liu, Hai-Xin Duan, Shuang Hao, and Zaifeng Zhang. A reexamination of internationalized domain names: The good, the bad and the ugly. 2018.

[16] Paul Pearce, Ben Jones, Frank Li, Roya Ensafi, Nick Feamster, Nick Weaver, and Vern Paxson. Global measurement of DNS manipulation. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 307–323, 2017.

[17] P.Mockapetris. Domain names–concepts and facilities, rfc 882. http://www.ietf.org/rfc/rfc882.txt, 1983.

[18] P.Mockapetris. Domain names–implementation and specification, rfc 883. http://www.ietf.org/rfc/rfc883.txt, 1983.

[19] P.Mockapetris. Domain names–concepts and facilities, rfc 1034. http://www.ietf.org/rfc/rfc1034.txt, 1987.

[20] P.Mockapetris. Domain names–implementation and specification, rfc 1035. http://www.ietf.org/rfc/rfc1035.txt, 1987.

[21] Enric Pujol, Philipp Richter, and Anja Feldmann. Understanding the share of IPv6 traffic in a dual-stack ISP. In Mohamed Ali Kaafar, Steve Uhlig, and Johanna Amann, editors, *Passive and Active Measurement*, pages 3–16, Cham, 2017. Springer International Publishing.

[22] Samuel Schüppen, Dominik Teubert, Patrick Herrmann, and Ulrike Meyer. FANCI : Feature-based automated nx-domain classification and intelligence. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 1165–1181, Baltimore, MD, August 2018. USENIX Association.

[23] Will Scott, Thomas Anderson, Tadayoshi Kohno, and Arvind Krishnamurthy. Satellite: Joint analysis of CDNs and network-level interference. In *2016 USENIX Annual Technical Conference (USENIX ATC 16)*, pages 195–208, Denver, CO, June 2016. USENIX Association.

[24] Dan Wing and Andrew Yourtchenko. Happy eyeballs: Success with dual-stack hosts. Technical report, 2012.

[25] Sandeep Yadav and A. L. Narasimha Reddy. Winning with DNS failures: Strategies for faster botnet detection. In Muttukrishnan Rajarajan, Fred Piper, Haining Wang, and George Kesidis, editors, *Security and Privacy in Communication Networks*, pages 446–459, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[26] Sebastian Zander and Xuequn Wang. Are we there yet? IPv6 in Australia and China. *ACM Transactions on Internet Technology*, 18(3), February 2018.