Comments on DNS Robustness

Mark Allman
International Computer Science Institute
mallman@icir.org

ABSTRACT

The Domain Name System (DNS) maps human-friendly names into the network addresses necessary for network communication. Therefore, the robustness of the DNS is crucial to the general operation of the Internet. As such, the DNS protocol and architecture were designed to facilitate structural robustness within system. For instance, a domain can depend on authoritative nameservers in several topologically disparate datacenters to aid robustness. However, the actual operation of the system need not utilize these robustness tools. In this paper we provide an initial analysis of the structural robustness of the DNS ecosystem over the last nine years.

CCS CONCEPTS

• **Networks** → **Network security**; **Network structure**; **Network reliability**;

KEYWORDS

DNS; nameserver; robustness; structure

ACM Reference Format:
Mark Allman. 2018. Comments on DNS Robustness. In *2018 Internet Measurement Conference (IMC '18), October 31-November 2, 2018, Boston, MA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3278532.3278541

1 INTRODUCTION

The Domain Name System's (DNS) basic job is to map human-friendly hostnames into network layer addresses [8]. As such, DNS transactions represent a prerequisite for a large amount of Internet transactions. It is hard to over-state the reliance we have on the DNS infrastructure for the operation of the Internet. Therefore, the robustness of the DNS ecosystem is of crucial importance. This paper provides an initial assessment of various robustness properties of the modern DNS ecosystem.

The DNS provides a hierarchical namespace and is structured as a distributed system. One of the key building blocks within the DNS is the ability to *delegate* portions of the namespace to particular authoritative nameservers. Delegation facilitates both flexibility and robustness, as follows:

Flexibility: Delegating to name owners allows the owners to configure the name resolution process according to their own specific needs. This flexibility manifests in myriad ways, including allowing for highly dynamic bindings that facilitate traffic management in content distribution networks, outsourcing DNS services to third party providers, controlling the caching of resolutions via varying time-to-live setting, etc.

Robustness: Delegating various portions of the namespace to a variety of nameservers allows for independent operation of different parts of the system. In other words, the overall DNS system gains robustness because any specific issue will only impact a relatively small amount of the namespace. Further, delegations can be made to multiple replica nameservers that all provide the same bindings. This allows the system to function even when a specific replica becomes unavailable, as clients can consult one of the other nameservers to obtain the required information.

DNS was designed to facilitate robustness. For instance, RFC 1034 requires each DNS zone to maintain two nameservers [8] and RFC 2182 further requires a zone's nameservers to have geographic and topological diversity [4]. However, this is insufficient to achieve *robust operation.* Rather, robustness requires careful configuration and operation. For instance, while DNS allows for nameserver replicas to avoid single points of failure, a DNS operator can configure only a single nameserver for their portion of the namespace, hence, negating the robustness potential in DNS's design.

One approach to gain robustness that has been gaining popularity is to out-source DNS operation to a DNS provider. These providers have expertise and resources—e.g., geographically disparate datacenters for hosting nameservers—that typical organizations either do not have or do not wish to build. While leveraging a DNS provider can improve robustness, this choice can also hurt robustness by concentrating the DNS ecosystem. For instance, consider the DDoS attack on the nameserver provider Dyn in October 2016 [2]. In this case, Dyn's nameservers were flooded by attack traffic from the Mirai botnet. This flood interfered with legitimate DNS requests. Given that Dyn runs nameservers on behalf of many organizations, this attack had indirect impact on these organizations, as well, since clients were not able to retrieve name-to-address mappings from Dyn. The organizations served by Dyn effectively all shared fate via their reliance on Dyn's DNS infrastructure.

Some of DNS's concentration is systemic and obvious. For instance, given the hierarchical namespace, all names depend on the root nameservers. To make the system robust we use 13 named and well known root replicas. Since the nameservers are well known, requesters have recourse when a particular server is unreachable. Further robustness at the root comes as most of the named replicas are in fact multiple nameservers that are reachable via anycast addresses. A similar situation manifests at the top-level domain (TLD) level—at least for popular TLDs such as *.com* and *.net*.

In this paper we focus on the robustness of second-level domains, or SLDs (e.g., *icir.org*). At this point in the name hierarchy, well known, broadly shared and community-driven infrastructure gives

IMC '18, October 31-November 2, 2018, Boston, MA, USA
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5619-0/18/10…$15.00
https://doi.org/10.1145/3278532.3278541

way to the individual decisions of millions of organizations. Further, at this level, the number of named replicas is generally smaller than at the root and TLD levels. In a snapshot of the *.com*, *.net* and *.org* zone files from April, 2018 we find that 80.6% of the SLDs have at most two named authoritative nameservers and 97.5% of the SLDs have at most four named authoritative servers. This provides less robustness and gives requesters less recourse when problems arise compared to the 13 named root nameservers. In this paper we study SLD robustness from a variety of angles.

This paper represents a modest initial effort. The DNS generally works well at providing users with the hostname-to-address mappings they need to accomplish their work. However, this paper describes a number of unhealthy habits that impinge on the envisioned robustness of the system.

2 RELATED WORK

Previous studies investigated the connectedness of the DNS nameserver ecosystem—which is one aspect of robustness. [9] uses data gathered in July, 2004 to study the "trusted computing base" (TCB) for nearly 600K names. In this context, the TCB is the set of nameservers on which a fully-qualified domain name (partially) depends. This investigation highlights the breadth of dependency in the system as the median TCB size was found to be 26 nameservers. The TCB notion was refined and an analytic model of the connections in the system is developed in [3].

A second study primarily aims to investigate the degree to which web server infrastructure is shared using data from June, 2007 [12]. This work also contains a preliminary investigation of shared DNS infrastructure. As with [9] and [3], this work starts with hostnames at the bottom of the hierarchy. This work shows sharing DNS nameservers is not rare. Further, this work moves beyond studying just the concentration of nameservers on shared infrastructure and into how many nameservers DNS zones utilize. We conduct a similar, but more in-depth characterization in § 4.1.

Our work is complimentary to the three previous studies we sketch above. First, we use a longitudinal and up-to-date dataset that spans from 2009–2018 to not only update our understanding, but also study how DNS robustness has evolved over time. Second, rather than starting at the bottom of the DNS hierarchy and trying to understand the breadth of influence on individual hostnames, we start closer to the top of the hierarchy and strive to understand the robustness of popular second-level domains (SLDs). Both approaches have their merits and together form a stronger understanding than either on its own.

Finally, there are many studies on specific DNS vulnerabilities (e.g., the Kaminsky attack [7], the fragmentation attack [6], the Preplay attack [11]). These investigate issues in the DNS protocol and/or implementations. While these issues are important, they are orthogonal to our work. In this paper we concentrate on structural robustness—or lack thereof—that stems from the way the DNS ecosystem has been intentionally configured and operated.

3 DATA AND METHODOLOGY

We leverage the following datasets and methodology in this study.

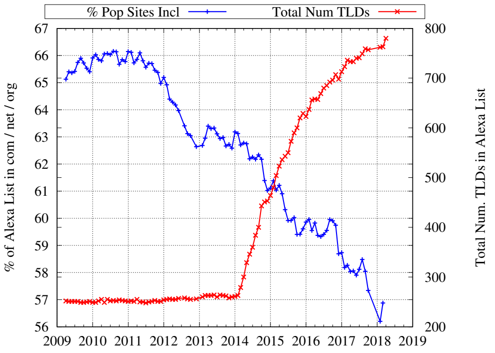

Figure 1: Overview of Alexa top 1M list.

3.1 Data

Dataset $\mathcal{A}$: Alexa Popularity Lists: Not all SLDs are equal, and therefore in this study we focus only on popular SLDs. The robustness of unpopular domains is certainly of crucial import to those leveraging these domains. However, in this study our goal is to understand robustness as it applies to domains on which large numbers of people depend. As such, we concentrate on the SLDs for the sites on Alexa's list of 1M most popular sites [1].[1] We have gathered Alexa's list regularly since 2009. We are missing Alexa lists for three months during our collection and therefore we do not consider these months in the analysis in this paper.

In this study we focus on the *.com*, *.net* and *.org* DNS zones (due to our zone file dataset, which we describe below). Not all entries in Alexa's list of popular sites are found in the three zone files we analyze. The blue line on Figure 1 shows the percentage of the Alexa list we analyze for each month of our dataset. While *.com*, *.net* and *.org* together constitute at least 56% of the Alexa list in each month we analyze, their contribution is decreasing over time. While we use roughly 63% or more of the Alexa list in the first half of the dataset, the percentage drops off in the second half. The red line on the plot shows the number of TLDs in the Alexa list for each month in the dataset. For the first half of our dataset, the original set of global TLDs (gTLDS; *.com*, *.net*, *.edu*, etc.) and the country-code TLDs (ccTLDs) make up the entire list. However, in 2014 new gTLDs started to be introduced and the number of TLDs in Alexa's list has therefore grown. With the exception of the *.de* and *.ru* TLDs, the TLDs we do not consider each make up less than 2% of the Alexa list. While our study encompasses the majority of the popular sites, we note that popular SLDs in unpopular TLDs may well have different robustness characteristics than we present here. We do not currently have the zone files to investigate these SLDs and therefore leave this to future work.

Dataset $\mathcal{Z}$: TLD Zone Files: Our second dataset is the zone files for the *.com*, *.net* and *.org* TLDs. We have daily snapshots of these zone files since April 2009. In this study we use the first snapshot from each month that correctly incorporates all three zones.[2] We

[1] We do not consider Alexa's list of popular sites to be definitive (see [10]). However, we consider the list to be a reasonable set of popular SLDs and use it as such.

[2] Generally, we use the snapshot from the first day of each month, but due to data gathering glitches at various times we may use an alternate snapshot. Since these glitches—e.g., full disks—have nothing to do with the contents of the zones, we do not believe these minor deviations bias our analysis.

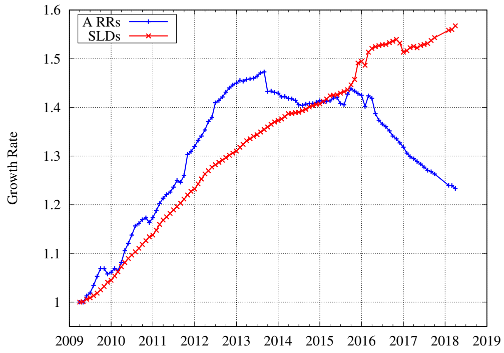

Figure 2: Overview of *.com*, *.net* and *.org* zone files.

are missing *.net* snapshots for three months during our collection period and do not consider those months in our analysis.[3] The red line in Figure 2 shows the growth rate of the number of SLDs in the three zones across our dataset relative to April 2009. The number of SLDs is generally increasing over time, as expected. The number of SLDs in the three TLDs increases from about 100M in April, 2009 to roughly 157M in April, 2018. While not shown on the figure, we find a corresponding increase in the number of NS records contained in the zone files—which makes sense since each SLD must have NS records. The blue line on the figure shows the growth in the number of A records in the three TLDs across time—again, relative to April 2009. The number of A records grows from 2.1M in April, 2009 to a high of roughly 3.1M in September, 2013. The number is then fairly stable until early 2016 at which point we observe a downward trend until the end of our dataset. This simple count cannot conclusively show anything because the count only covers three TLDs, but the plot does give us our first *objective inkling* that the DNS ecosystem is using fewer nameservers for more SLDs and therefore is becoming more concentrated.

Dataset $\mathcal{T}$: Traceroutes: From a host at ICSI in Berkeley, CA in April, 2018 we ran *traceroute* to every /24 we find in the April, 2018 zone files via step 2. Additionally, for /24s that we find to be highly utilized (as we discuss in § 5), we run *traceroutes* from a handful of looking glass servers located on each continent of the globe.

3.2 Methodology

Our analysis methodology is as follows.

Step 1: Mining NS Records: As we discuss above, we use one snapshot of the zone files for each month. Our first step is to gather the NS records from $\mathcal{Z}$ for all sites in the corresponding list in $\mathcal{A}$.

Step 2: Mining A Records: Next, we "resolve" the NS hostnames we find in step 1 to IP addresses using the A records in the given zone file snapshot in $\mathcal{Z}$. Since we only use three zone files, we are missing some A records that correspond to NS records in our zones (e.g., an NS record may point to *ns1.example.info* and since we do not have the *.info* zone file we are unable to determine the corresponding IP address).[4] Therefore, below we frame analyses in terms of (*i*) *fully resolved SLDs*, where we have A records for all NS records in each SLD; (*ii*) *partially resolved SLDs*, where we have A records for some, but not all, of the NS records in each SLD; and (*iii*) *unresolved SLDs*, where we have no A records for the NS records in each SLD. We pair the NS records found in step 1 with the A records found in this step to form Winnowed Zone Files (WZFs), which are the foundation of our analyses in this paper.

Step 3: Topological Determination: A final step in many of the analyses we describe in § 4 and § 5 is determining the topological location of authoritative servers. There are a number of possible approaches to determine the topological location of an IP address—e.g., via historical BGP or *traceroute* routing data. In this initial study, we generally delineate networks using /24 address blocks. While crude, this approach is conservative since we know that smaller address blocks cannot be confidently advertised in the Internet. Therefore, if two hosts share a /24 we can confidently assume they are routed to the same location. However, the opposite is not true: we cannot say that just because two hosts do not share a /24 that they will be routed to different locations in the network. We use the *traceroute* measurements from $\mathcal{T}$ to refine this crude notion for the last month of our analysis. Future work will include bringing better historical routing information to bear to refine the analyses we present in this paper.

A complication in understanding the modern DNS structure is the prevalent use of anycast. Large DNS providers often rely on anycast routing to direct requests to a nearby replica of a named authoritative nameserver (e.g., most of the 13 root nameservers leverage anycast) [5]. This helps performance by connecting DNS clients to a close replica. Further, using anycast increases the global robustness of the system by effectively breaking the network into regions that operate independently. Therefore, when issues arise—e.g., a replica becomes unreachable—the impact is to a topologically localized region rather than felt globally. As we sketch above, in our analysis we label authoritative nameservers by their IP addresses. Without taking into account whether anycast is in use we may underestimate the amount of global robustness present in the system. However, our robustness findings are at least germane to regions of the Internet. Consider the Dyn attack we sketch in § 1 [2]. Dyn's use of anycast in their infrastructure meant that instead of the attack being uniformly felt throughout the network it was more pronounced in certain regions than in others. This illustrates that while anycast is a beneficial technique for increasing global robustness, it does not cure all issues. Our future work includes taking anycast into consideration in our analysis.

4 ZONE ROBUSTNESS

First, we tackle the robustness of each SLD individually.

4.1 Nameserver Replicas

As we sketch in § 1, RFC 1034 requires each DNS zone to maintain two nameservers for robustness [8], while RFC 2182 requires these nameservers to be geographically and topologically diverse [4]. In other words, we should eschew single points of failure—whether that be a single host or a single network. We use the WZFs to study the extent to which these robustness requirements are followed in the wild. Figure 3 shows the percentage of the fully resolved SLDs

[3] The three months of missing *.net* zone files and the three months of missing Alexa lists do not overlap.

[4] Note, we could simply issue a DNS lookup to get the current IP addresses, but since our analysis spans the last nine years, this approach will not work.

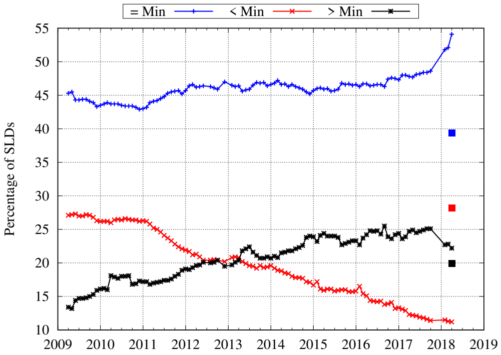

Figure 3: Percentage of SLDs that meet (blue), do not meet (red) and exceed (black) the nameserver requirements.

that meet (blue line), do not meet (red line) and exceed (black line) the minimum zone-level robustness requirements over time. For this analysis we use the crude notion that nameservers in different /24 blocks are topologically diverse, as we discuss in § 3.2. Therefore, the percentage of SLDs that do not meet the minimum requirements is a lower bound in that these SLDs cannot meet the requirements since all their nameservers fall within the same /24 address block. The percentage of SLDs that meet and exceed the requirements are upper bounds. I.e., while the nameservers span /24 address blocks, those blocks may in fact be routed to the same place. Finally, note the percentages for a given month do not sum to 100% because we elide partially resolved and unresolved SLDs, which together constitute 12–15% of the SLDs over time.

The plot shows the DNS ecosystem becoming more robust over time. The percentage of SLDs that do not meet the basic—and lenient in our /24-based analysis—requirements is decreasing over time. Meanwhile, the percentage of SLDs that meet or exceed the requirements is increasing. A more precise accounting requires historical routing data and is beyond the scope of this initial paper. However, the single points on the plot—color coded to match the lines—represent a refinement of our analysis for the last month of our dataset based on the *traceoute* data we describe in § 3. The single points show an analysis that is based on using the last hop router to determine network diversity.[5] The percentage of SLDs that exceed the requirements is only overestimated by about 2% using the crude /24-based analysis. This gives us confidence that the upward trend of SLDs increasing their diversity is in fact likely correct. However, the other two points show a 15% decrease in percentage of SLDs that exactly meet the requirements and an increase of 17% of SLDs that do not meet the requirements. This shows that while the red line provides a solid bound, the downward trend of SLDs failing to meet the minimum requirements may be misleading.

Recommendation 1: Roughly 28% of the SLDs currently do not meet the minimum requirements for the number and diversity of authoritative nameservers. The owners of these SLDs—or their chosen providers—should take steps to remedy this precarious situation.

Recommendation 2: Because routing is a dynamic process, it is generally difficult for a TLD operator to understand whether an SLD is in compliance with the topological diversity requirement without continuous auditing. However, we recommend that TLDs enforce a requirement that SLDs have nameservers in at least two distinct /24 address blocks. While this is not a guarantee of topological diversity, it is an easy step that has little downside.

4.2 Glue Location

We next turn to the process of resolving NS records to IP addresses. Resolving *www.example.com* involves a query to the *.com* TLD nameserver. The answer from the *.com* nameserver is a series of NS records (e.g., for *ns1.example.com*). These records must then be resolved into IP addresses before being useful. In this case, the *.com* TLD will have corresponding A records since the NS record falls within *.com*. The NS and A records—the so-called "glue"—will be served in the same DNS response. For instance, a response containing an NS record pointing to *ns1.example.com* will also contain an A record that maps *ns1.example.com* to 1.2.3.4.

Consider an NS record pointing to a different TLD—e.g., an NS record for *example.com* pointing to *ns1.dns-provider.net*. In this case, the response from the *.com* TLD will not include an A record. Rather, the requester must go through the process of querying for the A record of *ns1.dns-provider.net*. Compared to the case where the NS and A records and are in the same TLD, this process incurs both (*i*) additional lookup delay and (*ii*) additional failure modes. That is, now looking up a name within *example.com* can fail when the *.net* and *dns-provider.net* networks have issues, whereas in-zone glue has no such failure cases.

We find that 69–73% of the popular SLDs have at least one in-zone NS record across the span of our dataset. When no in-zone glue is available, the most robust approach is to use nameservers in multiple TLDs such that there is no single point of failure. Across our dataset we find at most 0.2% of the popular SLDs leverage multiple outside TLDs to provide the addresses for NS records. The remaining 27–31% of the SLDs (over time) rely on a single out-of-zone TLD to resolve NS records.

Recommendation 3: Even though TLD servers are robust—as we discuss in § 1—the cost of avoiding reliance on a second TLD in the lookup process is small. Mitigating this issue for the 27–31% of SLDs without in-zone glue does not require added resources, but simply adding in-zone glue to aid resolution of NS records. Therefore, we recommend all SLDs have at least some in-zone glue.

Recommendation 4: If no in-zone glue is available, we recommend SLDs use nameservers from multiple TLDs to avoid a single point of failure. As with the previous recommendation, the cost of this is only additional entries in zone files.

5 SHARED INFRASTRUCTURE

We now turn our attention from individual SLDs to building an understanding of how the inter-dependence between SLDs impacts robustness. As concrete motivation we note that over our nine year dataset 91–93% of the SLDs share at least one nameserver (by IP) with at least one other SLD. In the era of out-sourced DNS hosting, content-distribution networks and centralized cloud computing infrastructure this level of sharing is unsurprising. Our aim is to better understand the scope of shared DNS infrastructure.

[5] Note, *traceroute* cannot always find the last hop router and we fall back to the /24-based analysis when this occurs.

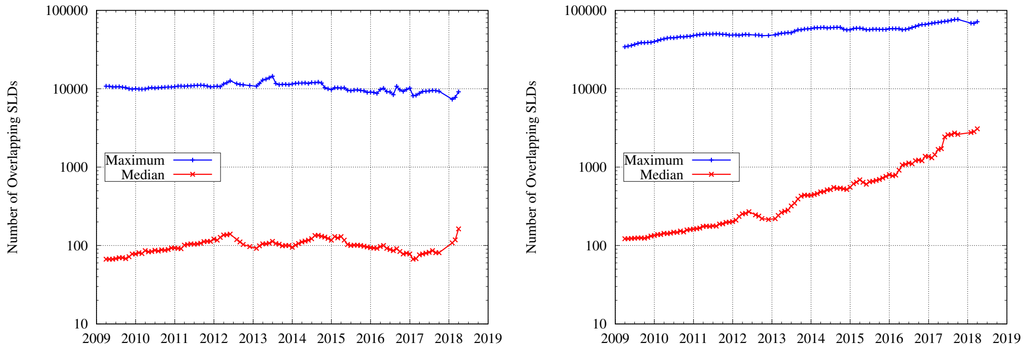

Figure 4: Median (red) and maximum (blue) SLD group size based on shared nameservers for IPs (left) and /24s (right).

Rank	Full SLDs	Partial SLDs	Num. IPs
1	9,135	15	2
2	8,347	3	2
3	5,568	375	3
4	5,076	69	2
5	3,938	47	10
6	3,657	31	5
7	3,144	1,043	5
8	3,069	19	2
9	2,967	336	3
10	2,610	1,225	6
	47,511	–	40

Table 1: Top ten SLD groups based on IP address.

Rank	Full SLDs	Partial SLDs	/24s	Same Last Hop
1	71,472	3,066	2	✓
2	69,637	328	2	
3	15,421	17	2	✓
4	13,044	3,727	2	✓
5	8,347	3	2	
6	6,111	631	2	✓
7	5,568	375	3	✗
8	5,076	69	2	
9	4,788	648	2	
10	4,611	4,820	4	
	204,075	–	23	–

Table 2: Top ten SLD groups based on /24 address prefix.

Our first exploration is via a host-based analysis of nameserver sharing. For each fully resolved SLD in our dataset we compute the number of other SLDs that use precisely the same set of IP addresses as authoritative nameservers. From these per-SLD counts we obtain a distribution across all SLDs for each month of our dataset. The left-hand plot of Figure 4 shows the median and maximum of this distribution over time. In April 2018 we find that half the SLDs exactly share a set of nameservers with at least 163 other SLDs. Further, we find the largest group contains 9K SLDs that share the exact same set of nameservers. Additionally, the plot shows nameserver sharing at the IP level is relatively stable over time, with small variations but no large trends.

Table 1 lists the ten largest SLD groups that rely on the exact same set of nameservers within the group in April 2018. The second column shows the group size, with the largest group containing 9K SLDs. In sum, the top ten groups contain over 47K SLDs—or, 4.7% of Alexa's list of top sites. The third column shows the number of additional SLDs that have some—but not complete—overlap with the group's nameserver set. The range of partial overlap is large—from only three SLDs in Group 2 to over 1K SLDs in Groups 7 and 10. The final column shows the size of the nameserver set for each group. Four of the groups use two nameservers—the minimum requirement prescribed in RFC 1034 [8]. However, most of the groups exceed the minimum requirements, with Group 5 utilizing ten nameservers. Each additional nameserver an SLD leverages decreases the chances that the SLD's names will become unresolvable due to host level issues. Across the ten largest groups we find 40 nameservers (out of 96K nameservers in the WZFs). In summary, we find 0.04% of the nameservers in our dataset are responsible for 4.7% of the popular SLDs.

We next turn to a network-based analysis of sharing. The analysis is similar to the host-based analysis, whereby we first determine the /24 address blocks that contain each fully resolved SLD's nameservers.[6] Next, for each SLD we determine the set of other SLDs that leverage nameservers in precisely the same /24 blocks. From these per-SLD counts we obtain a distribution across all SLDs for each month of our dataset. The right-hand plot in Figure 4 shows the median and maximum of this distribution over time. We find there is more shared infrastructure when viewed from a network perspective than from a host perspective. Further, we find that sharing network-level infrastructure is becoming more common over time. The plot shows that half the SLDs belong to groups with at least 3K other SLDs in April 2018. This is an increase of more than 25 times when compared to April 2009. Over our dataset we find that the maximum group size has more than doubled from 34K SLDs in April 2009 to 71K SLDs in April 2018.

Table 2 gives information about the largest ten SLD groups in April 2018. The largest group includes over 71K SLDs that have outsourced their DNS services to CloudFlare. The second largest group has nearly 70K SLDs and is run by another DNS provider, GoDaddy. We find a dramatic drop in the group size starting with Group 3—which is only 22% as big as Group 2. We find nine DNS providers across the ten groups—with the only repeat being Groups 1 and 4, consisting of distinct sets of CloudFlare customers. In total, the ten groups cover more than 20% of the popular SLDs.

The third column of the table shows the number of SLDs that have some, but not all, of their nameservers in the group's /24 blocks.

[6] As in § 4, using /24 prefixes as the basis of topological location provides a conservative bound since smaller address blocks cannot be confidently routed in the Internet.

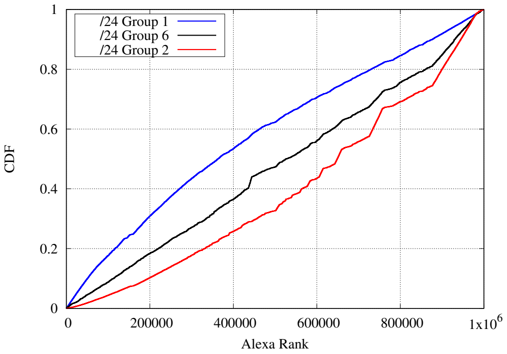

Figure 5: Dist. of Alexa ranks for 3 groups from Table 2.

Similar to our host-based results, we find this partial overlap to in some cases be small—e.g., Group 5 has only three SLDs that partially overlap. On the other hand, three groups each have more than 3K SLDs that partially share infrastructure with the group.

The fourth column of the table shows the number of /24 blocks that contain all nameservers for the group. Only Groups 7 and 10 employ more than two /24 address blocks, which represents the minimum topological diversity assuming the prefixes are routed to different edge networks. Across the ten groups we find all nameservers reside in 23 /24 address blocks.

We refine our analysis by consulting our *traceroute* data for all /24 blocks containing nameservers in the top ten groups. The fifth column shows whether the nameservers in the given group all use the same last hop router. We find four groups do in fact concentrate all their nameservers at a single edge network. Meanwhile, for Group 7 *traceroute* indicates there is in fact topological diversity. For the groups with no indication in the last column, the last hop router did not answer *traceroute* queries. However, in all these cases, inspection of the *traceroute* output indicates that the groups likely do not use the same last hop router for all their nameservers. In addition, while the /24 blocks used by Groups 1 and 4—both representing CloudFlare customers, as we note above—are distinct, all nameservers used by both groups share a last hop router. Therefore, over 86K SLDs rely on a single edge network. While we find nameservers in 23 /24 blocks, we determine that at most this represents 18 edge networks when digging more deeply into the network topology.[7] In other words, over 20% of the popular SLDs depend on 18 edge networks.

An additional note is that we find that three of the four groups in Table 2 that map to a single last hop router likely leverage anycast. Specifically, as we discuss in § 3, we use *traceroute* looking glasses on each continent to determine the route to each /24 used by the four groups that share a last hop router. We find that regardless of vantage point from which we run *traceroute* the nameservers within a group share a last hop router. However, the specific last hop router varies by where we run *traceroute*. This indicates that anycast is routing traffic to different edge networks for different source locations. As we sketch in § 3, anycast is beneficial for global robustness, but does not solve all robustness issues.

Finally, we analyze the data to determine whether shared infrastructure occurs more frequently in higher or lower ranked domains. Our analysis does not point to any general results or particular trends. Figure 5 shows the distribution of Alexa ranks for three exemplar groups from Table 2. A group with uniform distribution across the Alexa list would show as a straight diagonal line. The plot shows that Group 6 (black line) is approximately uniformly distributed. Meanwhile, Group 1 (blue line) skews towards more popular SLDs and Group 2 (red line) skews to less popular SLDs. Our initial analysis indicates that there is no predominant behavior across groups.

Recommendation 5: DNS providers should increase the topological diversity of their nameservers in terms of both servers and edge networks.

Recommendation 6: While it is tempting to simply delegate all responsibility to a single DNS provider, SLD owners should remain vigilant as to the robustness of the provider. Leveraging multiple providers or retaining a small bit of in-house DNS capability would increase an SLD's robustness. As future work we intend to build a web-based tool to aid SLD owners in understanding the robustness and connectedness of their domains.

Recommendation 7: Piggybacking on the previous suggestion, the DNS could benefit from the notion of "backup" records. In other words, an SLD owner could list a set of nameservers that are to be used only when the primary nameservers are unreachable. This would facilitate using multiple DNS providers. I.e., an SLD's general operation could stay as it is now—based on a single DNS provider—but the SLD could also have a backup provider (perhaps self-hosted) that can serve DNS responses only when problems arise.

6 SUMMARY & FUTURE WORK

While the analysis in this paper is admittedly initial and can clearly be improved in a number of ways—e.g., using historical routing data or a systematic understanding of anycast routing—we stress that the analysis is conservative. We add to an understanding of how DNS operates in the wild. Further, our goal is not to suggest that the sky is falling within the DNS ecosystem. However, we do highlight a number of places where the robustness of DNS could be improved—and in some cases at little cost.

Our future work includes a deeper analysis based on more and different data (e.g., more zone files, historical routing data). This will naturally include a deeper understanding of how anycast routing factors into DNS robustness. Also, we intend to tackle additional aspects of robustness that this initial work does not consider—e.g., issues arising from the intersection of structural concentration and software homogeneity.

ACKNOWLEDGMENTS

This paper benefits from the contributions of a number of people and organizations. Verisign provided the *.com*, *.net* and *.org* zone files. Emile Aben and Quirin Scheitle generously shared Alexa lists that filled in gaps in our own collection. The IMC reviewers and in particular our shepherd, Dina Papagiannaki, provided excellent guidance that improved the paper. This work was partially funded by NSF grants 1237265 and 1815876. Our thanks to all!

[7] This is a conservative analysis that assumes all last hop routers that *traceroute* cannot determine are distinct.

REFERENCES

[1] [n. d.]. Alexa Web Ranking. http://alexa.com.

[2] 2016. Dyn Analysis Summary Of Friday October 21 Attack. https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/.

[3] Casey Deccio, Chao-Chih Chen, Prasant Mohapatra, Jeff Sedayao, and Krishna Kant. 2009. Quality of Name Resolution in the Domain Name System. In *Intl. Conference on Network Protocols*.

[4] R. Elz, R. Bush, S. Bradner, and M. Patton. 1997. Selection And Operation of Secondary DNS Servers. RFC 2182.

[5] T. Hardie. 2002. Distributing Authoritative Name Servers Via Shared Unicast Addresses. RFC 3258.

[6] Amir Herzberg and Haya Shulman. 2013. Fragmentation Considered Poisonous, or: One-Domain-to-Rule-Them-All. In *IEEE Conf. on Communications and Network Security (CNS)*. 224–232.

[7] D. Kaminsky. 2008. Black Ops 2008: It’s the End of the Cache As We Know It. *Black Hat USA* (2008).

[8] P.V. Mockapetris. 1987. Domain Names - Concepts And Facilities. RFC 1034.

[9] Venugopalan Ramasubramanian and Emin GÃijn Sirer. 2005. Perils of Transitive Trust in the Domain Name System. In *ACM Internet Measurement Conference*.

[10] Quirin Scheitle, Oliver Hohlfeld, Julien Gamba, Jonas Jelten, Torsten Zimmermann, Stephen D. Strowes, and Narseo Vallina-Rodriguez. 2018. A Long Way to the Top: Significance, Structure, and Stability of Internet Top Lists. In *ACM Internet Measurement Conference*.

[11] Kyle Schomp, Tom Callahan, Michael Rabinovich, and Mark Allman. 2014. Assessing DNS Vulnerability to Record Injection. In *Passive and Active Measurement Conference*.

[12] Craig Shue, Andrew Kalafut, and Minaxi Gupta. 2007. The Web is Smaller than it Seems. In *ACM Internet Measurement Conference*.