Collecting “Typical” Domain Names for Web Servers

ICANN Office of the Chief Technology Officer

Paul Hoffman

OCTO-023
24 February 2021

ICANN

TABLE OF CONTENTS

This document is part of ICANN’s Office of the Chief Technical Officer (OCTO) document series. Please see the OCTO publication page for a list of documents in the series. If you have questions or suggestions on any of these documents, please send them to octo@icann.org.

This document supports ICANN’s strategic goal to improve assessment of, and responsiveness to, new technologies which impact the security, stability, and resiliency of the Internet’s unique identifier systems by greater engagement with relevant parties. It is part of ICANN’s strategic objective to evolve the unique identifier systems in coordination and collaboration with relevant parties to continue to serve the needs of the global Internet user base.

1 Introduction

When researchers measure the properties of the authoritative Domain Name System (DNS) servers on the Internet, they first need to define the types of authoritative servers they are sampling. The authoritative servers might be for domain names used for websites, for mail servers, for Internet infrastructure, and so on. Collecting domain names used for web servers is seen by many researchers as being fairly easy, and is thus the basis of much research on authoritative name servers.

However, the current collections of domain names against which one can do research are not that good for making assessments about “typical” domain names. The most popular websites are usually better managed than average websites, so lists of the most popular websites are not terribly representative of the web itself. Extracts from generic top-level domain (gTLD) zone files have many inactive names that are parked or are abandoned, so they too are not representative of the web. Dumps from passive DNS collection systems are inherently regional, and also skewed strongly against websites that are real but not popular.

One source of URLs for typical websites is the collection of the Wikipedias for all the languages of the world. Wikipedia pages often have links to sites that other sources would not have, such as the governments of small cities, colleges and universities of all sizes, obscure sports teams, small regional music and movie studios, personal sites of people who wrote just one popular blog article, and so on.

Wikimedia, the parent organization for all the Wikipedia sites, makes it easy to cull all the outward-facing URLs from the pages from all the Wikipedias. With that set of URLs, it is simple to reduce them to just domain names, and from there to create a set of unique instances of each of those names. This paper shows a methodology for creating a list of unique names, how a sample of those names was used to determine how many domain names for websites have IPv6 addresses, and how many are signed with Domain Name System Security Extensions (DNSSEC).

Note that the dataset here is derived from Wikipedia data, it is in no way associated with Wikipedia itself.

Although the dataset described here cannot be considered fully “typical” of the web, it addresses the drawbacks of many more commonly used lists. This document also discusses the properties of the dataset that would make it less than “typical” for the web, and also compares it with datasets of the most popular websites.

2 How the Dataset Is Collected

Wikipedia is a large collection language-specific encyclopedias, collections of books, collections of images, and so on. Wikimedia makes backups of the entire corpus of Wikipedia sites available for download, as described here.

As part of the backup sets, there are files that contain databases of all the external links from all the pages of every Wikipedia site, where “external” means to resources that are not other Wikipedia pages. The databases are updated at least twice a month.

To create a dataset of domain names used in these external links, the following steps are used:

1. Retrieve the database of external links for each language Wikipedia from a mirror of the main Wikipedia site.
2. From each database file, extract all the external links.
3. Clean up the list of external links, such as by removing those that are for URL schemes other than “http:” and “https:”, those with no URL scheme, those with bad syntax, those that use non-standard port numbers, and so on.
4. For each remaining URL, strip off the “http:” and “https:” scheme, and strip off everything after the host part, leaving just the domain name.
5. Cull the list of domain names so that only one copy of each domain name remains.

The database from 1 January 2021 was used for the current dataset. There were about 750 Wiki editions (that is, a Wiki in a particular language) in that database. After the processing, there were about 7.35 million unique domain names in the dataset.

For the data analysis described here, a set of 100,000 domain names was desired. During test runs of the data collection, it became clear that some of the domain names in the corpus looked valid but could not be resolved, so a random sample of 150,000 names were pulled from the full dataset with the intention of using 100,000 whose resolution for addresses were successful.

3 Results from Tests with this Data

The “getdns_query” tool from the getdns project was used to do the name resolution. This tool is available from package managers such as apt, yum, and brew. Queries for A records were sent with a request for DNSSEC status, and AAAA records were requested for any name that had an A record. The timeout for both queries was set to 4 seconds.

3.1 IPv6 and DNSSEC

Of the over 100,000 domain names tested that have IPv4 addresses, 17 percent also had IPv6 addresses. There was no attempt to measure domain names in this dataset that only had IPv6 addresses.

Of the over 100,000 domain names tested that have IPv4 addresses, 4 percent of those names were signed with DNSSEC.

3.2 TLS Startup Time

The dataset described here was originally created to answer a question from the IETF’s DPRIVE Working Group about how long a DNS recursive resolver should wait when probing a DNS authoritative server to see if the server supports DNS-over-TLS (DoT). To answer this question, assume that a typical web server that supports TLS would respond about as fast as a “typical” authoritative server that supports TLS. Also assume that a typical recursive resolver has a reasonably fast connection to the Internet, but might be geographically far away from the authoritative servers it speaks to.

To estimate how long TLS takes to set up with these assumptions, 100,000 IP addresses were sampled from the dataset of 100,000 domain names. A TLS connection was started to each of the addresses in the sample. The tests were all run from four systems in data centers in four locations around the world (Bangalore, San Francisco, Singapore, and Toronto) using the popular "curl" tool.

The tool measures the time from initiation to do DNS resolution (which is essentially zero in this test, because IP addresses were used), the further time to finish the TCP handshake, and the further time to finish the TLS handshake. The measurement for TCP startup time was subtracted from the TLS startup time so that results reflect purely the TLS delay. The 50th, 95th, and 99th percentiles of the startup time were measured with a cutoff of 4 seconds.

The results of those tests, in seconds, are:

	50th percentile	95th percentile	99th percentile
Bangalore TCP setup	0.151	0.289	0.397
Bangalore TLS setup	0.230	0.576	0.820
San Francisco TCP setup	0.071	0.196	0.273
San Francisco TLS setup	0.098	0.386	0.573
Singapore TCP setup	0.175	0.427	1.275
Singapore TLS setup	0.208	0.677	1.078
Toronto TCP setup	0.064	0.201	0.269
Toronto TLS setup	0.079	0.358	0.556

The test from Singapore had the longest latencies, and thus were used to determine how long to wait for the startup. The data suggests that setting a timeout of 2.4 seconds for a probe should allow about 99 percent of possible TCP+TLS setup times. Because the DoT protocol runs on its own port (853), one could instead use just the TCP setup time because there would be no reason to run TCP without TLS on port 853; with this assumption, a probe could just wait 1.3 seconds for TCP to be set up.

4 Assessing the Value of This Dataset

When creating a new dataset for research, it is always worth asking whether the new dataset is actually more suitable for a particular purpose than the databases already in use. This section looks at the datasets that are currently in high use (those of "most popular" websites), as well as problems with thinking that this new Wikipedia-based dataset meets its objective of representing "typical" websites.

4.1 Collections of "Most Popular" Web Sites

The Alexa top sites dataset, sold by Amazon, is probably the most commonly referenced dataset of "most popular" websites. It has commercial competitors that use different sources and different metrics, but Alexa is likely the best marketed of such datasets.

These “most popular” datasets have significant faults, however, the most notable being that it is surprisingly easy to manipulate the rankings. In 2019, a group of security researchers published a paper that showed how the various lists could be gamed in order to make it easier for attackers to lure people to distribute malware and perform phishing. As an antidote, the researchers created the TRANCO dataset which combines multiple “most popular” datasets and uses openly-described dampening mechanisms to prevent such gaming. The TRANCO dataset is now widely used in academic research where the commercial datasets were used earlier.

4.2 Comparing the TRANCO Dataset with the New Dataset

There is a surprising lack of overlap between the TRANCO and the Wikipedia-derived dataset. Even though the two datasets have 6.01 million and 7.35 million domain names, respectively, there are only about 0.42 million names in common. Note, however, that the TRANCO dataset has no names that begin with “www.”; if those labels are stripped from names in the Wikipedia dataset, there still are around only 0.90 million names in common. This shows that the two datasets have extremely different views of the web domain name space.

Running the metrics over a sample of 100,000 names from the TRANCO dataset showed similar results for percentage of names that had IPv6 addresses (16 percent for TRANCO versus 17 percent for the Wikipedia-derived dataset), but a noticeably smaller percentage of DNSSEC signing (2 percent for TRANCO versus 4 percent for the Wikipedia-derived dataset). The timings for setting up TCP and TLS were nearly identical with the Wikipedia-derived dataset.

4.3 Issues with the Wikipedia-Derived Dataset

Although it is clear that the dataset derived from the entire set of Wikipedia URLs covers many active websites that the “most popular” lists do not, it does not completely represent the “typical” web. Many types of sites that are accessed by individual users on a daily basis are probably badly under-represented in the Wikipedia-derived dataset, such as:

- Small restaurants
- Non-chain hotels and similar lodgings
- Construction companies
- Automobile dealerships and repair shops
- Local stores
- Small online shops

There are probably other large sources of links that might help make the new dataset more representative, but it is not clear if doing so would be worth the effort. There will always be gaps in any definition of “typical” for websites.

The dataset also fails at covering “typical” domain names because certainly not all domain names are for websites. There are millions of mail servers, many with their own domain names. There are also millions of pieces of Internet infrastructure, and many of those have their own domain names. A broader definition of a “typical” domain name would need to take these into consideration as well.

5 Conclusion

This paper documents a methodology for the creation of a dataset of domain names that aims to represent more “typical” websites than the more commonly used datasets developed by researchers. The dataset is derived from Wikipedia sites and includes the domain names from properly formed external URLs that are referenced on those sites.
Preliminary use of this dataset has provided useful information about how these more “typical” domain names are deployed.

When there are many types of datasets that can be analyzed, researchers will often want to compare them. This document compares two datasets, but there are certainly other comparisons that could be made, such as against just the most popular dataset (the Alexa top sites). However, it is not clear what the value of the comparisons might be, given that the primary value of having these datasets is to simply see what percentage of domain names might have particular properties, such as IPv6 and DNSSEC adoption.