Authoritatives at the Second Level: Testing SLD Nameservers

Paul Hoffman, ICANN

OARC 25, Dallas, October 2016

v3

Overview

- Looking at the **second level** via TLD zone files
- How many nameservers are likely related?
- How many do EDNS0? What MTU? Do they do NSID?
- How many are acting as open recursive resolvers?
- Notes on creating this testbed
- **Request** for more tests to run

Purpose of this research

- ICANN is interested in the **infrastructure** that supports the identifiers which help coordinate
  - There is lots of research about the root servers and TLD servers, but the DNS is served well beyond that
- This testbed gives us a view for how the overall nameserver system is working now, and how it can work in the future
- This might lead to better server **fingerprinting**
- This is **not** about naming and shaming or forcing fixes, even though there are some authoritative servers that do really weird things

There’s a lot at the second level

- You can find a lot of authoritative name servers by looking in the zone files of TLDs
- We wanted to test servers, so we went from NS records to **glue or lookups**, then **collapsed** by IP address
  - Test by IP address, not by NS name
- For the current run, we’re **only using the gTLDs**, but will add in ccTLDs in cooperation/collaboration with the ccTLD managers when we have a set of tools that ccTLD admins can use to give us the data themselves

Start from the gTLD zones

```
186,089,856 zones in the zone files
  3,468,129 NS names
  2,7601,99 NS names with glue records
    707,930 NS names without glue
    382,180 orphan glue names
```

- More than one tenth of glue records are **orphan glue**
- Many glue records have **questionable** addresses (127/8, private addresses, badly-formed IPv6 addresses, ...)

Reminder: NS names are infrastructure

- Many are not meant to be **typed**
  - zq708vote6hqo5uvbi2pult2dutvjq0u5evd075o9n214m3e15fltha0.skyedns.com
  - dns1.brinaldi.com.dns-not-in-service-ev1.com.dns-not-in-service-ev1.com.
- Some of the domains that nameservers are in are **not as stable** as some might think
  - For example, there are more than 60,000 NS names that are rooted at <elided>, which is for sale at Sedo with a US$90 minimum bid

Fill in the missing IP addresses

- First, query from **five different places** for A and AAAA of the NS records for which there was no glue, then combine the results with glue record data
- **Combine** with the names that had glue; the combined set has **1,481,301 IP addresses**
- Remove **private network and loopback** addresses (there were 405)
- 97.5% are IPv4 and 2.5% are IPv6

Mapping SLD nameservers to IPs

- Many nameserver names point to the **same IP**
- Try to associate **two names** with each IP address for which that address is **supposedly authoritative**

```
Number of NS names per IP address, by bucket:
1         : 941,867
2-9       : 508,358
10-19     : 17,579
20-99     : 11,684
100-999   : 1,265
1000-9999 : 117
10000+    : 26
```

Nameservers that are probably related

- In 299,116 /24s of X.Y.Z that had at least one nameserver, there were 241,117 **series of length 2 or greater**
- Of those series, 190,756 are length 2, 22,328 are length 3, and 16,220 are length 4
- The rest of the series lengths progress down, with **blips at length 16 and 50**
- There are 5 that have length 255, which probably means some series are >255

A bit of trivia about NS addresses

- Looking at the fourth octet in the IPv4 address of all the nameservers
- .0 and .255 are each appear one tenth as often as the typical octet value does
- .2 .3 .4 .5 .10 and .11 each appear more than twice as often as the typical octet value does

Testing for EDNS0 support

- Send one or two messages (different QNAMEs) with an **NSID extension** to each of the nameserver IP addresses from the five locations
  - Some nameserver addresses were only authoritative for one name
- There were 1,611,412 total responses to the 2,311,556 queries, **about 70%**
- Of those responses, 1,552,692 had EDNS0 in the Additional section, **about 95%**
- Total of 1,333,453 addresses

UDP size responses

- The values were completely **scattered**
- Popular announced sizes were 512, 1280, 1680, 2800, 4000, 4096, 65235, and "**reflect the size in query**"
- 4096 was by far the most popular, but "reflect" was second
- On the other hand, the size announced in responses appear to be **irrelevant** except to clients using UPDATE

RCODEs returned in the EDNS0 response

- **Almost all** returned 0, which is what we would want
- 600 of the 1.3 million returned the **DO bit set on**, even though RFC 3225 says to copy the DO bit into the response
- A small number returned 0x0000000f, 0x00000010, ...

NSID support

- 11,632 of the 1.3 million servers gave an **NSID response of some type**
- There were **19,253 unique NSIDs**
- Unsurprisingly, many IP addresses give different NSID responses to queries from different parts of the world

Acting as open recursive resolvers?

- Sent each authoritative IP a query with a real QNAME (**for which they are not authoritative**), type A, **RD=1**
- QNAME was AREALNAME.ORG, in all-caps
- The results were quite varied, and can probably be used for **fingerprinting**
- Of 1,333,453 addresses, **there were responses** from 84,421 servers
- From those servers, there was a total of 89,847 different responses (mostly due to different ordering in the Additional sections)

Interesting answers (1)

- Of the 89,847 replies, there were 47,059 Answer sections, and 233 had multiple answer records
- Of these answers, 5452 **changed the QNAME**
  - 3497 were lower.lower
  - 1549 were UPPER.lower
  - 5 were lower.UPPER
  - 3 were UPPER.mixedcase
  - 5 were “*.UPPER.UPPER”
  - Rest were unrelated to the QNAME

Interesting answers (2)

- Of these Answer sections, 7993 IPs gave 2450 different **wrong answers**
  - Only a few were CNAMES
  - Lots of just wrong IP addresses
  - Lots of junk
- Of the 89,847 replies, 57,636 had **Additional sections**
- 14,251 replies had both Answer and Additional sections

Notes on the testbed

- Processing of **zone files and responses** done on a hefty box at ICANN
- **Sending queries** to the authoritatives done from five Digital Ocean VMs
  - Located in **AMS, BLR, NYC, SFO, SGP**
  - 450 simultaneous tasks sending queries
  - 8Gb of RAM because 4Gb would sometimes die
  - $80/month each if we kept them up, which we don’t

How **not** to send out a zillion queries

- Send queries and collect answers in Python
- Like other languages, Python has libraries for doing **multiprocessing** and **async I/O**
- If you run too many workers on either type, the errors are **unpredictable** (dropped responses, lost threads, out of memory, ...)
- Even when you get the responses, you have to parse the DNS responses
  - **dnspython** is nice, but it is (apparently) not thread-safe and is also somewhat slow for parsing a million responses

How to send out a zillion queries

- Run **`tcpdump`** on port 53, writing out to a file (`-n -U src port 53`)
- Open UDP socket, send the queries, and **ignore the answers**
- **Timeout** after 10 seconds
- Stop tcpdump
- Parse the .pcap with **`dns_parse`** (https://github.com/pflarr/dns_parse)
- Parse the text output of `dns_parse` (tab and space separated) with Python

What to do next?

- Will re-run the tests after mixing in some **ccTLD** zones, once we figure out which zones that is OK with
- Will re-run with **more gTLDs** as they appear
- Will look at nameservers that are not in glue for some of their names
- I really want to **hear suggestions**, either here or afterwards