Looking at DNS traces: What do we know about resolvers?

Ólafur Guðmundsson

Shinkuro

Ogud@shinkuro.com

Motivation: Open Questions

- What is the "market share" for resolver X
- What can we deduct from traces to a subset of authorative servers ?
- Can we predict the effect of adding/deleting a server in location Y ?
- Do resolvers behave according to the RFC's?
  - Or more generally how do they behave?

Case #1: CCTTL adds foreign DNS servers by a commercial operator

- The new operator is supposed to answer questions from outside the country.
  - The new operator charges per query answered.
    - The CCTLD operator did their homework and predicted what percentage of queries are "foreign"
- The bills were higher than predicted
  - Is the operator over charging?
  - Is the operator "stealing" traffic from inside the country?
  - Was the model wrong?

Case #2: DNS operator change

I was asked to document a procedure that allowed transfer of a DNSSEC signed domain to a new operator

- Questions:
  - In what sequence to perform operations
    - How long to wait for information to propagate before next step
  - How do resolvers treat repeated information, such as NS set in authority section
    - Do they modify the TTL of the stored NS set ?
  - Which NS set do resolvers use ?
    - Important during change and when Parent or Child differ
    - Resolver that uses the Child set
    - and "stretches" the TTL
      - And asks question to the domain often enough
        - ➔ may not discover an operator change

Case #3: What % of resolvers support DNAME ?

- This question was raised in the context of D+C proposal
  - Quick survey showed less deployment of DNAME support than expected by looking at well known implementations
    - supported DNAME,
      - BIND, Unbound, WindowsDNS, Nominum
    - did not
      - Power Recursor, DJBdns/OpenDNS, Google,
    - How about all the other resolvers?

Case #4: DNSSEC validation in the wild

- I have been working with traces from .ORG to try to find out how much validation is going on.
  - Org. DNSKEY TTL is 15 minutes
  - Org. DS TTL is 1 day
  - 50 minute long traces
- I look for DNSKEY and DS queries from an resolver.
  - If a resolver asks for
    - DNSKEY twice (at least 15 minutes apart)
    - or for a DS for a signed domain
      - it is validating

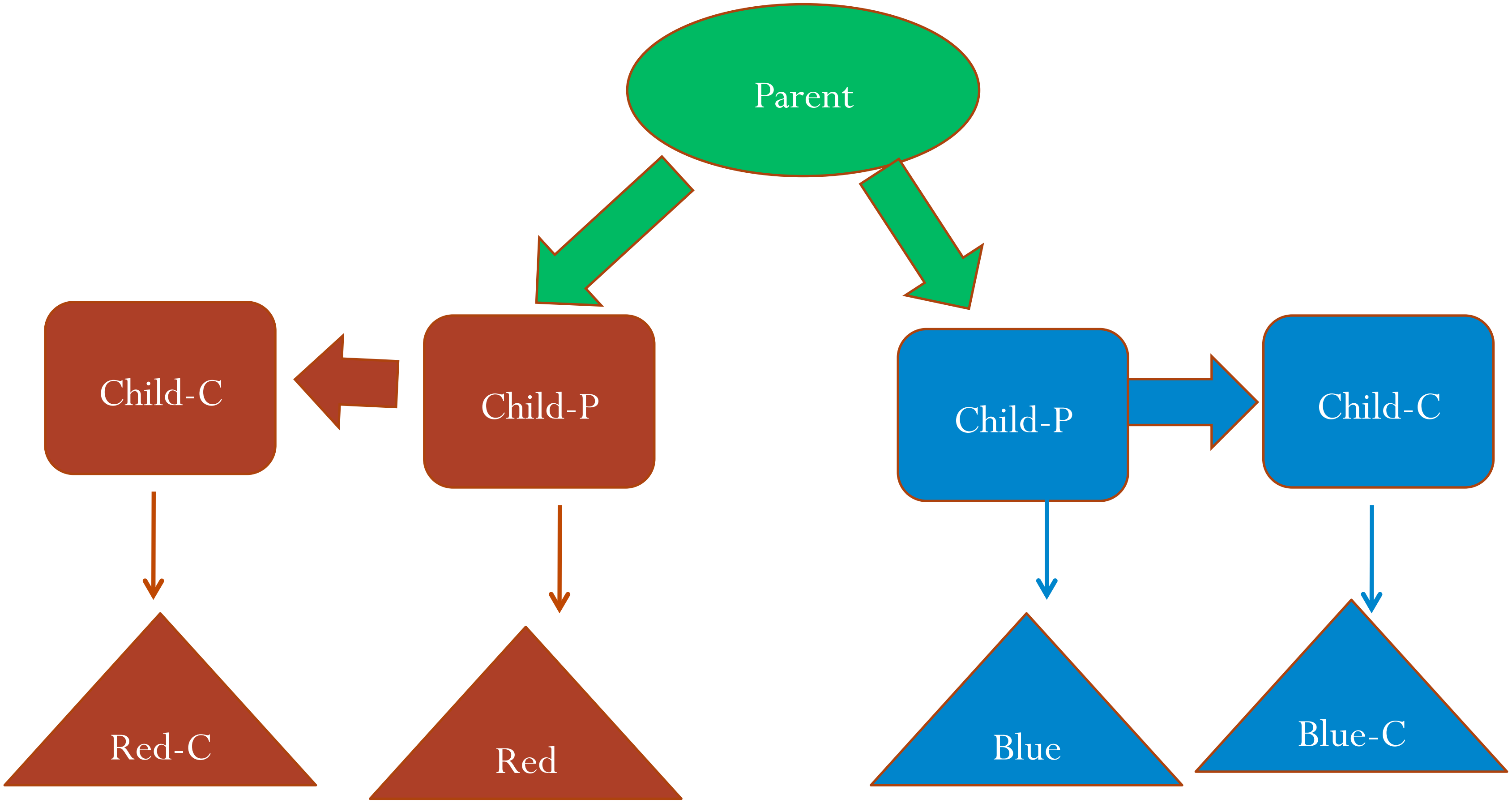
ORG DNSSEC validation study issues

- I only have traces for the DNS server that Afilias operates for .org.
  - Afilias operates 2/3 of NS records
  - PCH operates 1/3 of NS records
  - PCH has more sites
  - Afilias sees about 50% of query traffic
- My first questions:
  - what is the probably that my traces see a DNSKEY or DS query?
  - What kind resolvers do I see and why ?
  - Does the probably of seeing an **interesting** query depend on how busy the resolver is ?

Transfer work:

- Are resolvers parent or child centric?
- Do sticky resolvers exist?
- What is the percentage of each ?
- Simple experiments:
  - Set up a zone and with one server is only in one NS set
    - Works if all resolvers are queried with identical probability
  - Set up zone with two sets of authoritative server(s)
    - One set referenced in parent the other one in children

Sticky Resolver detection Zone setup



How good a sample of resolvers are open recursive resolvers ?

- I got a list of 856 open recursive resolvers
- 136 did not answer queries (16%)
- I probed the other 720 servers every 3 seconds 20 times for the record and recorded answer and TTL.
  - NS records TTL 17
  - TXT record TTL 7
- I asked the servers a final question:
  - "version.bind. TXT CH"

What should answers look like

- I process answers based on where answer comes from
  - P == Parent referenced server
  - C == Child referenced server.

- Child Centric non sticky:
  - PPPCCCPPPCCCPPPCCCPP
- Child Centric sticky
  - PPPCCCCCCCCCCCCCCCCC
- Parent Centric
  - PPPPPPPPPPPPPPPPPPPP

What answers look like

- **130** different patterns
- Top 10 are 536 or **74.4** %
  - 172 PPPCCCPPPCCCPPPCCCP ← Child Centric
  - 108 PPPCCCCCCCCCCCCCCCC ← Child Sticky 15%
  - 49 PPPCCCPPPCCCCCPPPCC
  - 42 PPPCCCCCPPPCCCPPPCC
  - 38 PPCCCCCCPPPCCCPPPCC
  - 34 PPPCCCPPPCCCPPCCCCC
  - 33 PPPCCCPPPCCCPPPCCCC
  - 26 PPPCCCPPCCCCCCPPPCC
  - 23 PPPPPPPPPPPPPPPPPPP ← Parent Centric 3%
  - 11 PPPCCCCCPPPCCCCCPPP

Closer look

- Most of resolvers are child centric and do not stretch, but some stretch some of the time
  - Child Sticky 15 % (108)
  - Parent Centric 3% (23)
  - Child Centric (517) 72%
    - 345 do not follow exact pattern 48%
    - 172 exact pattern 24%
  - Mostly Child Centric 10%   (72)
    - Partial sticky, relaxed handling of TTL, answer caching and reuse, ……

Behavior by implementations?

- 151 different version strings
- All Bind releases 9.3 and later are child centric
  - 9.2.3 and up are Child Centric
  - Older are Child sticky
- Bind 9.x shows up 305 times or 42%
  - 18 Parent Centric or mostly PC
  - 74 Child Sticky but only 7 say 9.2.1 or 9.2.1
  - 186 Mostly Child Centric
  - 25 Other
  - 62 Bind-9.6…. Only 12 behave according to my freshly compiled copy. But:

```
1 PPPCCCPPPPPPCCCPPPP
1 PPPCPCPCPPPPCCPCCPC
1 PPPPCCCCCCCCCCCCCCC
1 PPPPCPCPPPCPPPCCPPP
1 PPPPPCPCPCCCCPPCPPP
1 PPPPPPCCCPPPCCCPPPP
1 PPPPPPPCPCPCCCPPPPC
1 PPPPPPPCPPPCPPPPPPC
1 PPPPPPPPCCPPPPPCPPP
1 PPPPPPPPPPPPCCCPPPP
```

- Only 8 claim to be Bind 8.x
- none Bind 4.x

Concerns ?

- Nominum and Google DNS are Parent centric
- DNSCache and OpenDNS are Child Sticky
- There seem to be many cases where resolvers are not going to authority as expected:
  - Going through forwarder
  - Out of band synchronization
  - Minimum TTL enforcement
  - Or I'm asking any cast server cluster

Resolver query patterns

- How do Resolvers scatter queries ?
  - If a resolver discovers a domain and does not know about the name servers: it will ask a random resolver,
    - second query will go to DIFFERENT address
    - After all are probed queries will be concentrated inside a "band" of distance.
- How often do resolvers "forget" about closest authorative server.
- What is the market share of busy resolvers vs sporadic ones
  - Depends on the domain being queried e.g. ISP resolver in Brazil will likely show up as **sporadic** at *a.nic.cz* but **busy** at *a.dns.br*
- Does Address == resolver
  - We see many cases of multiple recursive resolvers behind NAT's

Resolver query patterns: effects

- **Sporadic** resolver will send DNSKEY query to the servers I see 66.7% of the time
- **Busy** resolver is likely to send **none** or **all** DNSKEY or DS query to servers I see
- **Busy** resolver that is “tied” to PCH site(s) will show up as **Sporadic** at Afilias sites and vice versa.
- How different are the query patterns at different sites?
  - Does prevalence of DNSSEC validation depend on regions?

Depressing Summary

- We do not know a lot about
  - Resolver NS set usage
  - Resolver query scattering
  - Resolver Market share,
  - Geographical differences
- We can not build reliable models of using DNS samples to answer basic questions
- What can be done to improve things?
  - Can we look at big collections and build models.
  - How can models evolve over time
    - For example: Bind-9.8 changed RTT banding from prior versions.

Positive ways forward

- What Models are needed:
  - List of models
- Can we find answers in existing traces
  - Documentation as what to look for and how
- Experiments to expose resolvers from the consumer side.
  - What do we want to know and how can we “trick” resolvers to expose their behavior ?
- Revisit the design and implementation choices for query scattering ?