Compact Denial of Existence in DNSSEC

Shumon Huque, Christian Elmerot, Ólafur Guðmundsson
July 24th 2023
DNS Operations Working Group
Internet Engineering Task Force (IETF) 117 Meeting
San Francisco, California, USA
Background

- DNSOP WG presentation from IETF116 (Yokohama), March 2023:
Working group version of draft

- Versioned draft:
- Datatracker link:
- Adopted May 2023
- Replaced individual draft from March 2023
- Added Olafur G (Cloudflare) as co-author.
- New section (work in progress) on RCODE substitution schemes.
- Need to update “Implementation Status” section.
Issue 1: NXDOMAIN or ENT signal or both?

- Should we support signals for both NXDOMAIN and Empty Non-Terminals?
  - NS1 today supports ENT sentinel (with RR type 65281)
  - Cloudflare supports NXNAME (with RR type 65283)

- Backwards compatibility because they are both deployed in the field?
  - Some folks were familiar with the ENT sentinel (and use it), and that a common bitmap could be used to identify NXDOMAIN across implementations earlier; and were surprised by the introduction of the Cloudflare NXNAME type.
  - See dnsop email thread that starts with the following message:
  - [https://lists.dns-oarc.net/pipermail/dns-operations/2023-March/021984.html](https://lists.dns-oarc.net/pipermail/dns-operations/2023-March/021984.html)
Issue 2: Official RR type allocation?

- Should we request official IANA RR type allocation(s), via the IANA early allocation process?
  - To avoid more changes by implementers down the road
- Or proceed in the field with the current private RR types?
- [from Olafur: defer; let’s get details of spec right first]
Issue 3: Explicit queries for NXNAME?

- This is a Pseudo RR type.
- Although nothing should be explicitly querying it, we should clarify what the response should be if such queries are received.
- 3 cases:
  - Query at name that exists:
    - Standard NODATA response, enumerating types that exist in the NSEC bitmap
  - Query at Empty Non-Terminal:
    - NODATA response (with ENT sentinel, if spec decides to include it)
  - Query at name that does not exist:
    - NODATA response with the standard NXNAME enhanced NSEC type bitmap
- (Repeat for ENT, if we decide to include it - result will be same)
Issue 4: RCODE 3 restoration

- Is it necessary to support mechanisms to restore NXDOMAIN (3) in the RCODE field?
- Opinions divergent on this issue, so perhaps this part of the spec, if developed, should be optional.
- Rationale:
  - Lots of security, diagnostic, and traffic characterization tools only example the RCODE field today. (In theory, they could be enhanced to recognize the NXNAME signal to deduce NXDOMAIN, but will they ever be?)
  - Most Name resolution APIs also know nothing about this new signal, so are induced to repeat queries unnecessarily at non-existent names for additional types.
Issue 4a: RCODE 3 restoration: DO=0 queries

- **Authoritative Servers**
  - Could just supply a normal NXDOMAIN response.
  - Is it worth it though, since most modern resolver always send DO=1 queries

- **Iterative Resolvers/Forwarders etc**
  - Recognize the NXNAME signal and restore NXDOMAIN in the RCODE field of responses it sends back. (Draft already mentions this possibility)
  - Additional cache management measures may be needed (tagging NXNAME responses for differential treatment to downstream queriers).
Issue 4b: RCODE 3 restoration: DO=1 queries

- Can the DNS server just set RCODE=3 for a Compact Denial answer, and assume that validators will only authenticate signed data in the response?
  - No, based on some empirical tests with deployed software.
- Hence, for validating queries (DO=1), signaling would be needed.
Sketch of a possible signaling scheme

- Define “Compact Answers OK” EDNS flag (“CO”?)
- If a DO=1 querier also sets CO=1, then if the DNS server implements Compact Denial, it can send the NXNAME enhanced NODATA response, and additionally set RCODE=NXDOMAIN
  - For an authority server this is fairly straightforward.
  - For an iterative resolver, this would need more complex cache management. It would have to tag Compact Denial answers appropriately (based on presence of the NXNAME signal), and then:
    - if downstream querier sets Compact Answers OK, return signed NODATA with RCODE=3
    - if downstream querier does not set it, returned signed NODATA with RCODE=0 (NOERROR) - basically the same answer as today without signaling.
Issue 5: Recommend or not?

- This spec standardizes a deployed existing practice.
- For new online signing implementers, should this mechanism be recommended? Or should this spec advise them to only consider this if they have the specific requirements that necessitate it?
- If they deploy RFC4470 (Minimally covering NSEC), they can avoid all the issues associated with missing or alternate NXDOMAIN signals.
  - And this too is deployed successfully in the field (both for NSEC and NSEC3)