Delegation Revalidation

draft-ietf-dnsop-ns-revalidation-01

IETF DNS Operations Working Group Interim
December 15th 2021
Shumon Huque, Paul Vixie, Puneet Sood

Revalidation algorithm (Section 4)

- Simple vs. more detailed algorithm
  - There appears to be a preference to keep only the simpler one.
  - Is anyone likely to implement the more detailed algorithm?
  - What corner cases does it better deal with?
  - Paul V is proposing some more streamlined text for the more detailed algorithm [TBD]

DS TTL

- DS TTL discussion
  - By spec, the delegating NS and DS TTL “SHOULD” match. In practice they don’t.
  - **If DS is present, resolvers MAY use DS TTL as the revalidation interval instead.**

Current text:

`If a secure delegation is present, resolvers may use the DS RRset's TTL as the revalidation interval in preference to to the delegating NS RRSet TTL.`

Proposal: use lower of DS and NS RRset TTL.

Delegation Changes

- Delegation changes, re-delegations, removals
  - If delegation is removed, ideally prune cache according to RFC8020
    - “prune” is not the only possible implementation; upward cache search at query time can expunge stale data as queried for
    - “SHOULD” (stale data may be dangerous - e.g. domain takedowns etc)
  - If zone has been re-delegated to entirely new set of child nameservers, then do the same.
  - If only a subset of NS entries have been re-delegated, then no cache cleanup is needed or recommended (avoid churn)

Lame Delegations

- Behaviour if entire NS set is lame: perform revalidation, with hold down timer to avoid DoS loop (what value for hold down timer?)

Proposal: perhaps no need to discuss this. This is behavior that resolvers have to deal with today even if they don’t implement delegation revalidation.

Resolver optimization

- Resolvers can cache whether authorities do minimal-responses and selectively forego subsequent child NS RRset fetches for those zones
  - Additional implementation complexity for currently unknown gain
  - How to detect state changes in a timely manner

Proposal: Don’t discuss. Remove.

Authoritative Server optimization

- Authorities: if employing minimal-responses, populate NS set in authority only for DNSKEY queries.
  - Additional implementation complexity for currently unknown gain
  - Moves the draft away from resolver behavior and moves into authoritative server behavior, which is not really the subject of the draft

Proposal: Don’t discuss. Remove.

Prevent abuse by others

- Resolvers should bound the amount of work they are willing to do (as a general principle)
- To avoid extremely frequent re-validations caused by very low TTL at the parent or child side, resolvers should place a lower bound on how frequently they will re-validate.
  - Should we recommend a specific (default?) value for that lower bound?
  - It should not be too high otherwise child zone operators cannot ensure quick migration and backout of new nameservers when they need to. Maybe 5 minutes, 15 minutes?

Q&A / Discussion

-