Network Working Group
Internet-Draft
Intended status: Informational
Expires: May 18, 2017

W. Kumari
Google

November 14, 2016

# Stretching DNS TTLs
## draft-wkumari-dnsop-ttl-stretching-00

## Abstract

The TTL of a DNS Resource Record expresses how long a record may be cached before it should be discarded. This document discusses the possibility of "stretching TTLS" (using them past their expiration) if they cannot be refreshed. This works on the assumption that stale data may be better than no data.

PLEASE NOTE: This document is a strawman to drive discussion. It may or may not be a good idea; this document documents the idea so that there is something concrete to throw tomatoes at.

[ Ed note: Text inside square brackets ([]) is additional background information, answers to frequently asked questions, general musings, etc. They will be removed before publication. This document is being collaborated on in Github at: https://github.com/wkumari/draft-wkumari-dnsop-ttl-stretching. The most recent version of the document, open issues, etc should all be available here. The authors (gratefully) accept pull requests ]

## Status of This Memo

This Internet-Draft will expire on May 18, 2017.

Copyright Notice

Table of Contents

## 1. Introduction

DNS Resource Records (RR) have an associated TTL. This is how long the record may be cached before it should be expired and new information fetched. This is based upon the assumption that the authoritative servers will be reachable when they are needed, and that records expire and are immediately evicted from the cache.

There are a number of reasons why an authoritative server may become unreachable, including, unfortunately, Denial of Service (DoS) attacks. Recent proposals, for example "Highly Automated Method for Maintaining Expiring Records" ([I-D.wkumari-dnsop-hammer]) propose refreshing records in the cache before they expire and are evicted. This means that the recursive server still has information in its cache when it attempts to contact the authoritative server.

This document suggests that, if the recursive server is unable to contact the authoritative server, it simply extends the existing

records TTL, on the assumption that "stale bread if better than no bread".

[Ed: This is the primary point of the document / question -- if you cannot reach the authoritative nameservers (perhaps they being DoS-ed, perhaps they were unplugged, you cannot really tell) it is better to use the last known (and perhaps outdated) information, or is it better for the domain to go dark? I think the former, but this is a significant change to the meaning / semantics of TTLs).

### 1.1. Requirements notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

## 2. Proposal

If a recursive nameserver is unable to contact any of the authoritative nameservers for a zone, and it still has the resource record cached, it MAY "stretch" the TTL by simply increasing it it by the original TTL. It may do this N times, where N should be configurable.

[ Ed: I was going to say "by doubling the TTL", but then if we allow implementations to do this e.g 3 times, is that 4 times the original TTL, or is it 2^3 the original TTL].

## 3. IANA Considerations

This document contains no IANA considerations.Template: Fill this in!

## 4. Security Considerations

TODO: Fill this out!

## 5. Acknowledgements

The authors wish to thank some folk.

## 6. References

### 6.1. Normative References

[IANA.AS_Numbers]
IANA, "Autonomous System (AS) Numbers",
<http://www.iana.org/assignments/as-numbers>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

## 6.2. Informative References

[I-D.ietf-sidr-iana-objects]
Manderson, T., Vegoda, L., and S. Kent, "RPKI Objects issued by IANA", draft-ietf-sidr-iana-objects-03 (work in progress), May 2011.

[I-D.wkumari-dnsop-hammer]
Kumari, W., Arends, R., and S. Woolf, "Highly Automated Method for Maintaining Expiring Records", draft-wkumari-dnsop-hammer-00 (work in progress), July 2013.

## Appendix A. Changes / Author Notes.

[RFC Editor: Please remove this section before publication ]

From -00 to -01

- Nothing changed in the template!

Author's Address

Warren Kumari
Google
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: warren@kumari.net