DNS TIMEOUT Resource Record
draft-pusateri-dnsop-update-timeout-03

Abstract

This specification defines a new DNS TIMEOUT resource record (RR) that associates a lifetime with one or more zone resource records with the same owner name, type, and class. It is intended to be used to transfer resource record lifetime state between a zone’s primary and secondary servers and to store lifetime state during server software restarts.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 7, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of
DNS Update [RFC2136] provides a mechanism to dynamically add/remove DNS resource records to/from a zone. When a resource record is dynamically added, it remains in the zone until it is removed manually or via a subsequent DNS Update. The context of a dynamic update may provide lifetime hints for the updated records (such as the EDNS(0) Update Lease option [I-D.sekar-dns-ul]), however, this lifetime is not communicated to secondary servers and will not endure through server software restarts. This specification defines a new DNS TIMEOUT resource record that associates lifetimes with one or more resource records with the same owner name, type, and class that can be transferred to secondary servers through normal AXFR [RFC5936], IXFR [RFC1995] transfer mechanisms.

An UPDATE lifetime could be stored in a proprietary database on an authoritative primary server but there is an advantage to saving it as a resource record: redundant master servers and secondary servers capable of taking over as the primary server for a zone automatically can benefit from the existing database synchronization of resource records.
records. In addition, primary and secondary servers from multiple vendors can synchronize the lifetimes through the open format provided by a resource record.

TIMEOUT records can be installed via policy by a primary server, manually, or via an external UPDATE from a client. If TIMEOUT records are being managed by an UPDATE client, the client should be aware of server software policy with respect to TIMEOUT records to prevent the TIMEOUT records from being rejected. The primary server has ultimate responsibility for the records in the database and the client must work within the restrictions of the policy of the primary server.

TIMEOUT records can be thought of as a universal method for removing stale dynamic DNS records. Clients such as DHCP lease servers who best know the lease lifetimes can include individual TIMEOUT records in the dynamic UPDATE messages specific for each lease lifetime. These TIMEOUT records can be refreshed when the lease is refreshed and will timeout the A, AAAA, and PTR records if they are not refreshed by the DHCP server. Additional use cases include service discovery resource records installed in unicast DNS servers via UPDATE described in [RFC6763], Active Directory Controllers publishing SRV records, and DNS TXT validation resource records supporting ACME certificate management as described in Section 8.4 of [RFC8555].

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here. These words may also appear in this document in lower case as plain English words, absent their normative meanings.

3. Sources of TIMEOUT Expiry Time

The expire time may come from many different sources. A few are listed here however, this list is not considered complete. TIMEOUT records may be included along side the records they represent in the UPDATE message or they be synthesized by the primary server receiving the UPDATE.

1. Via DHCP Dynamic Lease Lifetimes.

2. Via EDNS(0) Update Lease option [I-D.sekar-dns-ul] communicated in DNS Update.
3. Via an administrative default value such as one day (86400 seconds).

4. Resource Record Composition

TIMEOUT resource records provide expiry times for a mixed variety of resource record types with the same owner name, type, and class. Since there could exist multiple records of the same record type with the same owner name and class, the TIMEOUT resource record must be able to identify each of these records individually with only different RDATA. As an example, PTR records for service discovery [RFC6763] provide a level of indirection to SRV and TXT records by instance name. The instance name is stored in the PTR RDATA and multiple PTR records with the same owner name but only differing RDATA often exist.

In order to distinguish each individual record with potentially different expiry times, the TIMEOUT resource record contains an expiry time, the record type, a method to identify the actual records for which the expiry time applies, and a count of the number of records represented. Multiple TIMEOUT records with the same owner name and class are created for each expiry time, record type, and resource record representation. If the expiry time is the same, multiple records can be combined into a single TIMEOUT record with the same owner name, class, and record type but this is not required.

The fields and their values in a TIMEOUT record are defined as:

4.1. Represented Record Type

A 16-bit field containing the resource record type to which the TIMEOUT record applies. Multiple TIMEOUT records for the same owner name, class, and represented type can exist. Any resource record type can be specified in the Represented Record Type including another TIMEOUT record. This specification does not put any restrictions on the record type but implementations in authoritative servers will likely do so for policy and security reasons.

4.2. Represented Record Count

The Represented Record Count is a 8-bit value that specifies the number of records of the specified record type with this expiry time.

An RR Count of zero indicates that it is not necessary to represent any records in the list. This is a shortcut notation meaning all resource records with the same owner name, class, and record type use the same Expiry Time. When the Represented Record Count is 0, the
Method Identifier is set to NO METHOD (0) on transmission and ignored on reception.

In the unlikely event that the Represented Record Count exceeds 255 which is the largest number representable in 8 bits, multiple instances of the same Expiry Time can exist.

4.3. Method Identifiers

The Method Identifier is a 8-bit value that specifies an identifier for the algorithm used to distinguish between resource records. The identifiers are declared in a registry maintained by IANA for the purpose of listing acceptable methods for this purpose. In addition to the method and the index, the registry MAY contain a fixed output length in bits of the method to be used or the term "variable" to denote a variable length output per record. It is conceivable, though not likely, that the same method could be used with different fixed output lengths. In this case, each fixed output length would require a different identifier in the registry. Additions to this registry will be approved with additional documentation under expert review. At the time that the registry is created by IANA, a group of expert reviewers will be established.

Additional methods of representing records such as hashes or other algorithms may be defined in the future. If such methods are defined, a primary server could create TIMEOUT record using a new method that is not understood by a secondary server that could take over as the primary in the event of an outage or administrative change. In this case, the new primary would not be able to identify the records it is supposed to TIMEOUT. This is a misconfiguration and it is the responsibility of the administrator to ensure that secondary servers in a position to become primary understand the TIMEOUT record methods of the primary server.

4.3.1. Method Identifier 0: NO METHOD

The method identifier of 0 is defined as "NO METHOD" and MUST NOT be used if the represented record count is greater than 0. The value of 0 is to be included in the IANA registry of method identifier values.

4.3.2. Method Identifier 1: RDATA

The method identifier of 1 is defined as "RDATA". It begins with the RDATA length as a 16-bit value containing the length of the RDATA in bytes followed by the number of bytes of RDATA as appears in the record being represented. The record MUST be in canonical DNSSEC form as described in Section 6 of [RFC4034]. Any comparisons of RDATA in an actual record covered by the TIMEOUT against the
Represented RDATA contained in a TIMEOUT record must be compared in canonical form.

4.4. Expiry Time

The expiry time is a 64-bit number expressed as the number of seconds since the UNIX epoch (00:00:00 UTC on January 1, 1970). This value is an absolute time at which the record will expire. Lease times must be converted to an absolute expiry time when received.

5. TIMEOUT RDATA Wire Format

The TIMEOUT resource record follows the same pattern as other DNS resource records as defined in Section 3.2.1 of [RFC1035] including owner name, type, class, TTL, RDATA length, and RDATA.

The RDATA section of the resource record with method identifier RDATA (1) is illustrated in Figure 1:

```
 0                   1                   2                   3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Represented RR Type      |   Count (n)   |   Method (1)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Expiry Time (64-bit)                    |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.    Represented RDATA LEN 1    |                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.  Represented RDATA 1        .
.                               .
.                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.    Represented RDATA LEN n    |                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.  Represented RDATA n        .
.                               .
.                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Method (1) RDATA Wire Format

Figure 1 represents an arbitrary number of represented records with the same owner name, class, and represented type. For each expiry
time, a list of RDATA length and RDATA pairs are attached. The overall RDATA length of the TIMEOUT record indicates when the last represented record is contained in the record.

The RDATA section of the resource record with method identifier NO METHOD (0) is illustrated in Figure 2:

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Represented RR Type      |   Count (0)   |   Method (0)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Expiry Time (64-bit)                    |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Method (0) RDATA Wire Format

Figure 2 represents the TIMEOUT RDATA field of all matching records of the represented type for the same owner name and class.

6. Primary Server Behavior

A TIMEOUT resource record MUST be removed when the last resource record it covers has been removed. This may be due to the record expiring (reaching the expiry time) or due to a subsequent DNS Update or administrative action. If the TIMEOUT record is being managed by the primary server, it has the responsibility to remove it. If the TIMEOUT record was installed by an external UPDATE client, the UPDATE client has the responsibility to remove it. The primary server is the ultimate source of the database and policy established by the server may overrule the actions of external clients. The primary server is ultimately responsible for ensuring the database is consistent but until TIMEOUT record management is built-in to authoritative server software, external UPDATE clients will likely manage the records.

Upon receiving any DNS UPDATE deleting resource records that might have been covered by a TIMEOUT RR, a primary server MUST remove all represented records in all of the TIMEOUT records with the same owner name, class, and represented type.

The TIMEOUT record TTL should use the default TTL for the zone like any other record. The TTL values of the records covered by a TIMEOUT are not affected by the TIMEOUT expiry time and may be longer than the expiry time. The TIMEOUT RR is mostly for the benefit of the
authoritative server to know when to remove the records. The fact that some records might live longer in the cache of a resolver is no different than other records that might get removed while still in a remote resolver cache.

7. Secondary Server Behavior

A secondary server may or may not understand TIMEOUT resource records. If a secondary server does not understand them, they are treated like any other resource record that the server may not understand [RFC3597].

A secondary server MUST NOT expire the records in a zone it maintains covered by the TIMEOUT resource record and it MUST NOT expire the TIMEOUT resource record itself when the last record it covers has expired. The secondary server MUST always wait for the records to be removed or updated by the primary server.

8. TIMEOUT RDATA Presentation Format

Record Type:
resource record type mnemonics. When the mnemonic is unknown, the TYPE representation described in Section 5 of [RFC3597]

Represented Record Count:
unsigned decimal integer (0-255)

Method Identifier:
unsigned decimal integer (0-255)

Expiry Time:
The Expiry Time is displayed as a compact numeric-only representation of ISO 8601. All punctuation is removed. This form is slightly different than the recommendation in [RFC3339] but is common for DNS protocols. It is defined in Section 3.2 of [RFC4034] as YYYYMMDDHHmmSS in UTC. This form will always be exactly 14 digits since no component is optional.

YYYY is the year;
MM is the month number (01-12);
DD is the day of the month (01-31);
HH is the hour, in 24 hour notation (00-23);
mm is the minute (00-59); and
SS is the second (00-60) where 60 is only possible as a leap second.

RDATA Length:
unsigned decimal integer
9. IANA Considerations

This document defines a new DNS Resource Record Type named TIMEOUT to be exchanged between authoritative primary and secondary DNS servers. It is assigned out of the DNS Parameters Resource Record (RR) Type registry. The value for the TIMEOUT resource record type is TBA.

This document establishes a new registry of DNS TIMEOUT Resource Record Method Identifier values. The registry shall include a numeric identifier, a method name, a description of the method, and the length of the output function in bits or the keyword "variable". The identifier is to be used in the RDATA section of the TIMEOUT resource record.

Initially, there are two values defined in the registry. Values from 240 (0xF0) through 255 (0xFF) are reserved for experimental use.

<table>
<thead>
<tr>
<th>ID</th>
<th>Method Name</th>
<th>Description</th>
<th>Length (bits)</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>NO METHOD</td>
<td>All records match</td>
<td>0</td>
<td>Section 4.3.1</td>
</tr>
<tr>
<td>1</td>
<td>RDATA</td>
<td>Actual RDATA of represented records</td>
<td>variable</td>
<td>Section 4.3.2</td>
</tr>
<tr>
<td>240-255</td>
<td>EXPERIMENTAL</td>
<td>Reserved for Experimental Use</td>
<td>variable</td>
<td>Section 9</td>
</tr>
</tbody>
</table>

Table 1: TIMEOUT RR Method Identifier values

10. Security Considerations

There is no secure relationship between a TIMEOUT resource record and the represented resource records it applies to. TIMEOUT records should typically only apply to resource records created through the UPDATE mechanism. Protection for permanent resource records in a zone is advisable.

Authenticated UPDATE operations MUST be REQUIRED at authoritative name servers supporting TIMEOUT resource records.
11. Acknowledgments

This idea was motivated through conversations with Mark Andrews. Thanks to Mark as well as Paul Vixie, Joe Abley, Ted Lemon, Tony Finch, Robert Story, Paul Wouters, Dick Franks, JINMEI, Tatuya, and Timothe Litt for their suggestions, review, and comments.

12. References

12.1. Normative References


12.2. Informative References


Appendix A. Example TIMEOUT resource records

The following example shows sample TIMEOUT resource records based on DNS UPDATEs containing A and AAAA address records plus the corresponding PTR records.

A host sending a name registration at time $T_n$ for "A" and "AAAA" records with lease lifetime $L_n$ would have a series of UPDATEs (one for each zone) that contain:

<table>
<thead>
<tr>
<th>Name</th>
<th>RR Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>s.example.com.</td>
<td>A</td>
<td>192.0.2.5</td>
</tr>
<tr>
<td>s.example.com.</td>
<td>AAAA</td>
<td>12001:db8::5</td>
</tr>
<tr>
<td>5.2.0.192.in-addr.arpa.</td>
<td>PTR</td>
<td>s.example.com.</td>
</tr>
<tr>
<td>5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.b8.0d.01.20.ip6.arpa. (bytes)</td>
<td>PTR</td>
<td>s.example.com.</td>
</tr>
</tbody>
</table>

Table 2: Example Address Records Update

Next, consider the TIMEOUT resource records that would be generated for the records in Table 2.
Table 3: Address TIMEOUT records

Next, assume there are two hosts advertising the same service type (different service types will have different owner names). We will use _ipp._tcp.example.com as an example.

Host A sends an UPDATE at time Ta with lease life La for PTR, SRV, A, AAAA, and TXT records. Host B sends an UPDATE at time Tb with lease life Lb for PTR, SRV, A, and TXT records.

Table 4: DNS UPDATE from Host A

Table 5: DNS UPDATE from Host B

For these printer registrations, the TIMEOUT records on the server would look like the following:
| Owner Name                | For Type | C | Met hod | Expire / RDLEN RDATA |
|---------------------------+----------+---+---------+---------------------|
| _ipp.tcp.example.com.     | PTR      | 1 | 1       | Ta + La 25 p1._ipp._|
|                           |          |   |         | tcp.example.com.     |
| _ipp.tcp.example.com.     | PTR      | 1 | 1       | Tb + Lb 25 p2._ipp._|
|                           |          |   |         | tcp.example.com.     |
| p1._ipp._tcp.example.com. | SRV      | 0 | 0       | Ta + La              |
| p1._ipp._tcp.example.com. | TXT      | 0 | 0       | Ta + La              |
| p2._ipp._tcp.example.com. | SRV      | 0 | 0       | Tb + Lb              |
| p2._ipp._tcp.example.com. | TXT      | 0 | 0       | Tb + Lb              |
| p1.example.com.           | A        | 0 | 0       | Ta + La              |
| p1.example.com.           | AAAA     | 0 | 0       | Ta + La              |
| p2.example.com.           | A        | 0 | 0       | Tb + Lb              |

Table 6: Service TIMEOUT records

Authors’ Addresses

Tom Pusateri
Unaffiliated
Raleigh, NC
USA

Email: pusateri@bangj.com

Tim Wattenberg
Unaffiliated
Cologne
Germany

Email: mail@timwattenberg.de