Workgroup: Internet Engineering Task Force
Internet-Draft: draft-dnsop-update-timeout-00
Published: 24 May 2020
Intended Status: Standards Track
Expires: 25 November 2020
T.J. Pusateri
Unaffiliated
T. Wattenberg
Unaffiliated

DNS TIMEOUT Resource Record

Abstract

This specification defines a new DNS TIMEOUT resource record (RR) that associates a lifetime with one or more zone resource records. 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 25 November 2020.

Table of Contents

1. Introduction

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 necessarily 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. 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 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, DNS TXT resource records supporting ACME certificate management challenges as described in [RFC8555], Section 8.4, and the limited lifetime certificate representations produced by ACME that are stored in DANE TLSA resource records [RFC6698].

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.

*Via DHCP Lease Lifetimes.

*Via EDNS(0) Update Lease option [I-D.sekar-dns-ul] communicated in DNS Update.

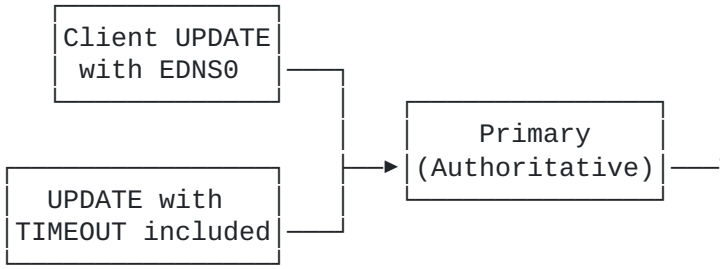
*Via an administrative default value such as one day (86400 seconds).

4. Common Usage Patterns

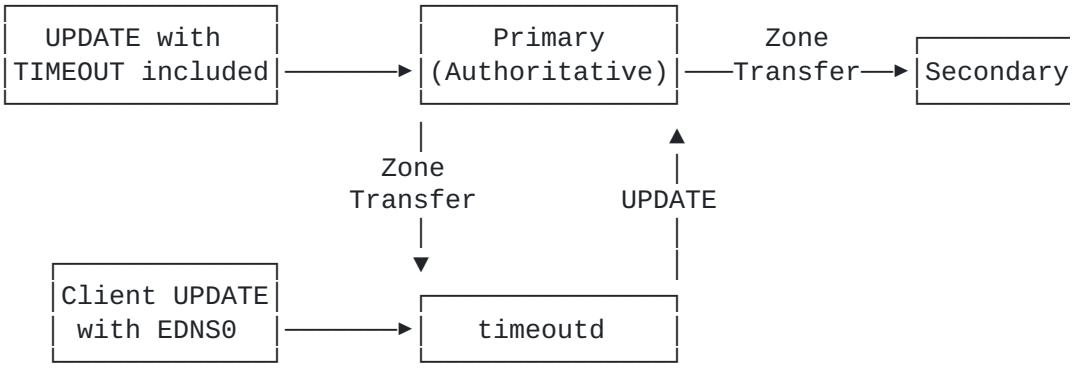
TIMEOUT resource records are just one tool in the toolbox for cleaning up stale resource records. They provide a failsafe in case other mechanisms meant to clean up records fail. It might be useful to think of them similar to Garbage Collection (GC) or Automatic Reference Counting (ARC) used by programming languages for memory management. The model in which the TIMEOUT resource records are used depends on the support provided for them by the primary DNS server.

As it cannot be presumed that all primary authoritative servers will manage TIMEOUT resource records internally, an external management of the TIMEOUT records and the resource records they represent might be necessary. The client may perform external management of TIMEOUT records it creates through an UPDATE or a third party with appropriate permission may manage the records.

If the primary server understands TIMEOUT records and manages them based on resource record updates, it will likely know when to remove the resource records referenced by the TIMEOUT records. This is similar to ARC.



If the primary server does not understand TIMEOUT records, then an external manager (client) will need to use DNS UPDATE to manage TIMEOUT records and the resource records they reference. Garbage Collectors run periodically looking for memory no longer being used to reclaim. In a similar way, external TIMEOUT record managers need to periodically scan the TIMEOUT records and send DNS UPDATE messages to add/remove records when the server doesn't manage them automatically.



It should be noted that similar to many instances of Garbage Collection, the precision with which TIMEOUT records and the resource records they reference are removed is not critical. Gross timers and/or scanning mechanisms are perfectly appropriate and should not consume additional resources for the purpose of being precise. As described in Section 5.4 below, expiry times use one second resolution.

4.1. TIMEOUT records vs. Update Leases

Each application will have to determine when it is better to use TIMEOUT resource records, EDNS(0) Update Lease options, or a combination of the two. In some cases, either will serve the same purpose. A differentiating factor is that TIMEOUT resource records use absolute time so that the records may be more easily synchronized across secondary servers whereas Update Leases are specified in relative time offsets.

If your primary DNS server supports TIMEOUT records directly, it may be simpler to just provide an Update Lease lifetime in the DNS UPDATE message that the server will use to create the TIMEOUT records internally. If your primary DNS server does not support TIMEOUT records and your application uses sources that have real-time clocks that are synchronized with standard time sources, TIMEOUT records are an available option to the client. However, if your clients are using low-cost hardware without real-time clocks, they should send Update Leases to the primary server or an intermediate proxy with a synchronized real-time clock.

4.2. Testing for TIMEOUT

There is no more reliable mechanism to determine if the primary DNS server supports the management of TIMEOUT records than explicitly trying it. Before relying on a server to expire TIMEOUT records, the application should send test records and test if they are handled as expected. If the preferred mode of operation is not supported, another mode can be attempted. For example, if sending a DNS UPDATE with a EDNS(0) Update Lease of 1 second doesn't cause the record to be expired within 6 seconds (1 + 5 fuzz), then the application can try including a TIMEOUT record in the DNS UPDATE. If that doesn't

automatically expire, TIMEOUT records will need to be managed externally.

5. 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 and 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:

5.1. 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.

QTYPEs and Meta-TYPEs MUST NOT be used as the represented record type. For more information, refer to [RFC6895], Section 3.1.

5.2. 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.

A 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. A primary server MUST NOT install a TIMEOUT record with No Method/No Count at the same time that a TIMEOUT record exists for the same owner name, class, and type with a non-zero record count. Either all records MUST match the No Method/No Count shorthand syntax or they MUST all be included in the list of matching records.

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.

5.3. 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 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.

5.3.1. 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.

5.3.2. Method Identifier 1: MD-SHA256-128

The method identifier of 1 is defined as MD-SHA256-128. Following the expiry time is a list of 128-bit values. Each of these values is the first 128-bits of a message digest of the RDATA of a represented record in canonical DNSSEC form calculated using the 256-bit SHA-256 hash algorithm defined in [FIPS180-4]. The canonical DNSSEC form is described in [RFC4034], Section 6. The input length of RDATA for the message digest is the RDLEN of the represented record.

5.4. 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. An absolute time is necessary so the TIMEOUT records do not have to change during zone transfers.

There are circumstances when a relative expiry time would be convenient due to limited resources for clock synchronization in

constrained devices. In this case, DNS UPDATE messages should not contain precomputed TIMEOUT records but convey the relative expiry time using the EDNS(0) Update Lease Option defined in [I-D.sekar-dns-ul]. The relative time is then converted to an absolute expiry time when received by the primary server which will create the TIMEOUT resource records.

6. TIMEOUT RDATA Wire Format

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

The RDATA section of the resource record with method identifier NO METHOD (0) is illustrated in Figure 1:

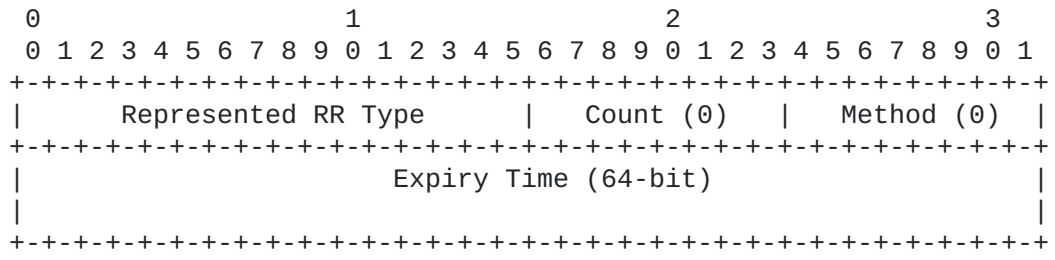


Figure 1: Method (0) RDATA Wire Format

Figure 1 represents the TIMEOUT RDATA field of all matching records of the represented type for the same owner name and class.

The RDATA section of the resource record with method identifier MD-SHA256-128 (1) is illustrated in Figure 2:

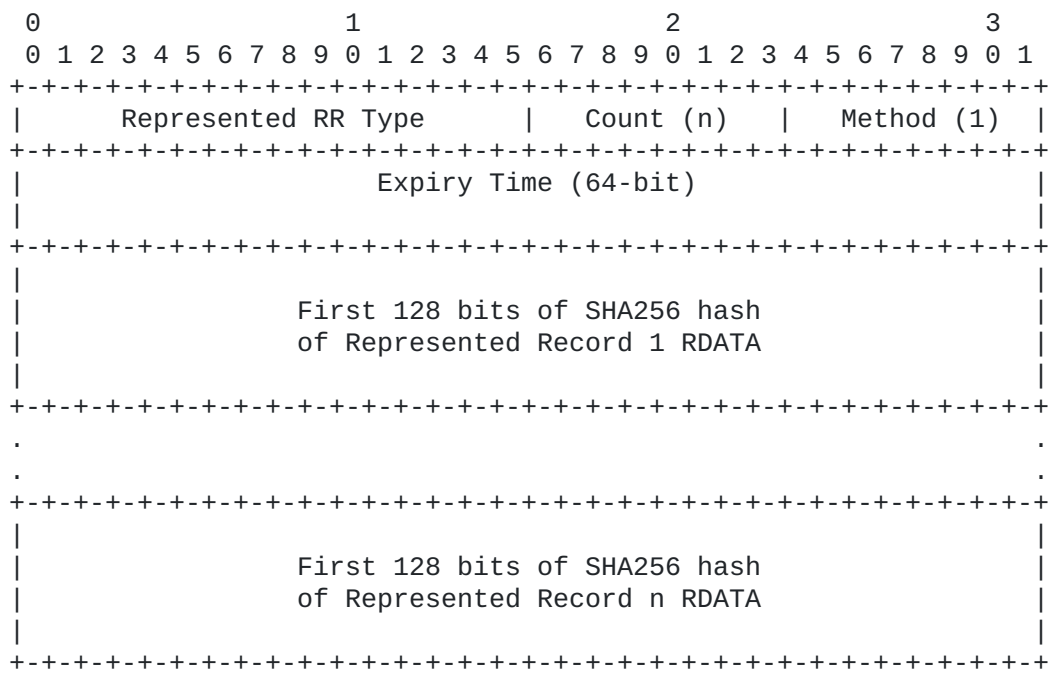


Figure 2: Method (1) MD-SHA256-128 Wire Format

Figure 2 represents an arbitrary number of represented records with the same owner name, class, and represented type. For each expiry time, a list of the first 128-bits of a SHA256 hash are appended.

7. Server Behavior

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

7.1. Primary Server Behavior

The primary server is the ultimate source of the database and policies 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.

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.

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.

7.2. Secondary Server Behavior

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 is represented by the word "TYPE" immediately followed by the decimal RR type number, with no intervening whitespace as described in [RFC3597], Section 5

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 [RFC4034], Section 3.2 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.

List of 0 or more hashes depending on Method Identifier: (hash-1 hash2 ...)

hash values shown as upper case hexadecimal string;

some type of white space MUST exist between hash values but MUST NOT exist within hash value;

MUST only display parentheses for one or more hash values;

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.

Type	Value	Meaning	Definition
TIMEOUT	TBA	expire represented records	Section 5

Table 1: DNS Parameters Resource Record Registry

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.

ID	Method Name	Description	Length (bits)	Definition
0	NO METHOD	All records match	0	Section 5.3.1
1	MD-SHA256-128	List of 128-bit hashes of represented records RDATA	128 bits	Section 5.3.2
240-255	EXPERIMENTAL	Reserved for Experimental Use	variable	Section 9

Table 2: 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, Timothe Litt, and Stuart Cheshire for their suggestions, review, and comments.

12. References

12.1. Normative References

[FIPS180-4] National Institute of Standards and Technology, U.S. Department of Commerce, "Secure Hash Standard (SHS) FIPS 180-4", August 2015, <http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

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

[RFC3339] Klyne, G. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, DOI 10.17487/RFC3339, July 2002, <https://www.rfc-editor.org/info/rfc3339>.

[RFC3597] Gustafsson, A., "Handling of Unknown DNS Resource Record (RR) Types", RFC 3597, DOI 10.17487/RFC3597, September 2003, <https://www.rfc-editor.org/info/rfc3597>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC6895] Eastlake 3rd, D., "Domain Name System (DNS) IANA Considerations", BCP 42, RFC 6895, DOI 10.17487/RFC6895, April 2013, <https://www.rfc-editor.org/info/rfc6895>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

12.2. Informative References

[I-D.sekar-dns-ul] Cheshire, S. and T. Lemon, "Dynamic DNS Update Leases", Work in Progress, Internet-Draft, draft-sekar-dns-ul-02, 2 August 2018, <https://tools.ietf.org/html/draft-sekar-dns-ul-02>.

[RFC1995] Ohta, M., "Incremental Zone Transfer in DNS", RFC 1995, DOI 10.17487/RFC1995, August 1996, <https://www.rfc-editor.org/info/rfc1995>.

[RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)",

RFC 2136, DOI 10.17487/RFC2136, April 1997, <https://www.rfc-editor.org/info/rfc2136>.

[RFC5936] Lewis, E. and A. Hoenes, Ed., "DNS Zone Transfer Protocol (AXFR)", RFC 5936, DOI 10.17487/RFC5936, June 2010, <https://www.rfc-editor.org/info/rfc5936>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <https://www.rfc-editor.org/info/rfc6698>.

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, DOI 10.17487/RFC6763, February 2013, <https://www.rfc-editor.org/info/rfc6763>.

[RFC8555] Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", RFC 8555, DOI 10.17487/RFC8555, March 2019, <https://www.rfc-editor.org/info/rfc8555>.

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 Tn for A and AAAA records with lease lifetime Ln would have a series of UPDATEs (one for each zone) that contain:

Name	RR Type	Value
s.example.com.	A	192.0.2.5
s.example.com.	AAAA	2001:db8::5
5.2.0.192.in-addr.arpa.	PTR	s.example.com.
5.0.0.0.0.0.0.0.0.0.0.0.0.0.b8.d.1.20.ip6.arpa.	PTR	s.example.com.

Table 3: Example Address Records Update

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

Owner Name	Type	Cnt	Mth	Expire
s.example.com.	A	0	0	Tn+Ln
s.example.com.	AAAA	0	0	Tn+Ln
5.2.0.192.in-addr.arpa.	PTR	0	0	Tn+Ln
5.0.0.0.0.0.0.0.0.0.0.0.0.0.b8.d.1.20.ip6.arpa.	PTR	0	0	Tn+Ln

Table 4: 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.

Owner name	RR Type	Value
_ipp._tcp.example.com.	PTR	p1._ipp._tcp.example.com.
p1._ipp._tcp.example.com.	SRV	0 0 631 p1.example.com.
p1._ipp._tcp.example.com.	TXT	paper=A4
p1.example.com.	A	192.0.2.1
p1.example.com.	AAAA	2001:db8::1

Table 5: DNS UPDATE from Host A

Owner name	RR Type	Value
_ipp._tcp.example.com.	PTR	p2._ipp._tcp.example.com.
p2._ipp._tcp.example.com.	SRV	0 0 631 p2.example.com.
p2._ipp._tcp.example.com.	TXT	paper=B4
p2.example.com.	A	192.0.2.2

Table 6: DNS UPDATE from Host B

For these printer registrations, the TIMEOUT records on the server would look like the following:

Owner Name	Type	C	M	Expire / Hash
_ipp.tcp.example.com.	PTR	1	1	Ta+La 69D67BCB98E8809702B9DFCA6B865558
_ipp.tcp.example.com.	PTR	1	1	Tb+Lb 7EBE34BC8B3E7306F8FCF1D6805331E1
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 7: Service TIMEOUT records

Authors' Addresses

Tom Pusateri
Unaffiliated
Raleigh, NC
United States of America

Email: pusateri@bangj.com

Tim Wattenberg
Unaffiliated
Cologne
Germany

Email: mail@timwattenberg.de