Registration Extensions (REGEXT)
Internet-Draft
Intended status: Standards Track
Expires: 28 November 2026

G. Brown
ICANN

27 May 2026

RDAP Extension for DNS Time-To-Live (TTL Values)
draft-ietf-regext-rdap-ttl-extension-12

Abstract

This document specifies an extension to the Registration Data Access Protocol which allows the Time-To-Live (TTL) values for relevant DNS record types to be included in RDAP responses.

About this draft

This note is to be removed before publishing as an RFC.

The source for this draft, and an issue tracker, may can be found at https://github.com/gbxyz/rdap-ttl-extension.

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 28 November 2026.

Copyright Notice

Table of Contents

1. Introduction 3
2. Conventions used in this document 3
3. RDAP Response Specification 3
3.1. DNS record Types and TTL Values 5
3.2. RDAP Conformance 5
4. Operational Considerations 5
4.1. RDAP Servers 5
4.2. RDAP Clients 5
5. IANA Considerations 6
6. Security Considerations 6
7. Implementation status 6
7.1. ICANN RDAP 6
7.2. Net::RDAP 7
7.3. rdapper 7
8. Change Log 7
8.1. Changes from 11 to 12 7
8.2. Changes from 10 to 11 8
8.3. Changes from 09 to 10 8
8.4. Changes from 08 to 09 8
8.5. Changes from 07 to 08 9
8.6. Changes from 06 to 07 9
8.7. Changes from 05 to 06 9
8.8. Changes from 04 to 05 9
8.9. Changes from 03 to 04 9
8.10. Changes from 02 to 03 9
8.11. Changes from 01 to 02 10
8.12. Changes from 00 to 01 10
8.13. Changes from draft-brown-rdap-ttl-extension-03 to draft-ietf-regext-rdap-ttl-extension-00 10
8.14. Changes from 02 to 03 10
8.15. Changes from 01 to 02 10
8.16. Changes from 00 to 01 10
9. Acknowledgements 11
10. References 11
10.1. Normative References 11
10.2. Informative References 12
Author’s Address 12

1. Introduction

The Registration Data Access Protocol (RDAP, [STD95]) provides access to information about Internet resources (domain names, autonomous system numbers, and IP addresses). While RFC9083 allows RDAP server operators to provide information about the content of the NS, DS, A and AAAA RRset(s) (see Section 5 of [RFC9499]) which are published in the DNS for a given registry object (domain or host object), it does not provide a mechanism to allow the Time-To-Live (TTL) (see Section 5 of [RFC9499]) values of those RRsets to be included in responses. Inclusion of these values in RDAP responses (in addition to nameservers, glue IP addresses, and DS records) allows out-of-band debugging of the DNS configuration of troublesome domain names.

This document describes how TTL information can be included in domain and nameserver objects in RDAP responses. As per Section 5.2 of [RFC2181], TTL values are applicable to RRSets rather than individual records.

2. Conventions used in this document

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.

This document makes use of the terms defined in Section 1.1 of [RFC9083].

3. RDAP Response Specification

Servers that support this extension MAY include a "ttl0_data" property in any domain (Section 5.3 of [RFC9083]) and nameserver (Section 5.2 of [RFC9083]) objects included in RDAP responses. As per Section 2.1 of [RFC9083], clients which do not implement this specification SHOULD ignore the "ttl0_data" member.

The ttl0_data property is an object which has the following properties:

* A "values" property, which is an object which maps DNS record type mnemonics to TTL values;

* An OPTIONAL "remarks" property, which is an array of remarks (see Section 4.3 of [RFC9083]).

As specified in Section 8 of [RFC2181], a TTL value is "an unsigned number, with a minimum value of 0, and a maximum value of 2147483647. That is, a maximum of 2^31 - 1". TTL values MUST be represented as JSON numbers with no fractional component and no exponent notation.

The TTL values included in "ttl0_data" properties MUST reflect the TTL values as provisioned in the registry database, not the remaining TTL of DNS records as observed from live DNS queries.

An example domain object with valid "ttl0_data" attribute is provided below. Readers should refer to RFC9083 for a description of the other objects listed in the example.

```
{
  "objectClassName": "domain",
  "rdapConformance": ["rdap_level_0", "ttl0"],
  "ldhName": "domain.example",
  "ttl0_data": {
    "values": {
      "NS": 3600,
      "DS": 300
    },
    "remarks": [
      {
        "description": [
          "For more information about the .example",
          " registry policy relating to DS record TTL changes,",
          "see https://domain.example"
        ],
        "links": [
          {
            "rel": "related",
            "title": ".Example Registry DNS TTL Policy",
            "href": "https://domain.example"
          }
        ]
      }
    ]
  }
}
```

An example nameserver object with valid "ttl0_data" attribute is provided below.

```
{
  "objectClassName": "nameserver",
  "rdapConformance": ["rdap_level_0", "ttl0"],
  "ldhName": "ns1.domain.example",
  "ttl0_data": {
    "values": {
      "A": 86400,
      "AAAA": 86400
    },
    "remarks": [
      {
        "description": [
          "The .example registry does not permit TTL ",
          "values for nameservers to be changed."
        ]
      }
    ]
  }
}
```

3.1. DNS record Types and TTL Values

The DNS record type mnemonics that appear as the property names in values objects MUST be in all capitals and MUST be registered with IANA in [IANA-RRTYPES]. TTL values MUST be unsigned integers in the range 0-2147483647 as per Section 8 of [RFC2181].

3.2. RDAP Conformance

Servers returning responses containing TTL values MUST include the string "ttl0" in the rdapConformance array.

4. Operational Considerations

4.1. RDAP Servers

This specification is complementary to the Extensible Provisioning Protocol [RFC5730] (EPP) Mapping for DNS Time-to-Live (TTL) Values [RFC9803], but registry operators do not need to implement that extension in their EPP server in order to implement this RDAP extension.

4.2. RDAP Clients

Many RDAP clients make use of frameworks which automatically "hydrate" objects using JSON data received in RDAP responses. As a result, RDAP clients which use these frameworks should explicitly carve out the "values" property of "ttl0_data" elements.

Since the list of record types appearing in "ttl0_data" elements may change with time, clients which implement this extension MUST accept responses containing values for all valid DNS record types, and SHOULD periodically update the list of valid DNS record types to align with [IANA-RRTYPES], to avoid discarding a recently-added record type.

5. IANA Considerations

IANA has registered the following value in the RDAP Extensions Registry [IANA-RDAP-EXTENSIONS]:

Extension identifier: ttl0

Registry operator: Any

Published specification: this document

Contact: IETF <iesg@ietf.org (mailto:iesg@ietf.org)>

Intended usage: this extension describes how DNS TTL values can be included in RDAP responses.

6. Security Considerations

Security services for the extension specified in this document are described in RFC7481.

This document only concerns itself with the representation of configured TTL values for domain and host objects. The security implications of how those TTL values are determined, assigned, or modified within a registry system are out of scope. Readers are referred to Section 6 of [RFC9803] for further discussion.

7. Implementation status

This section is to be removed before publishing as an RFC.

7.1. ICANN RDAP

Organization: ICANN

Name: ICANN RDAP

Description: ICANN RDAP contains an RDAP client, server, and common libraries.

Level of maturity: Mature

Coverage: All aspects of the protocol are implemented.

Licensing: Apache License, Version 2.0 or MIT License.

Contact: globalsupport@icann.org

URL: https://github.com/icann/icann-rdap

7.2. Net::RDAP

Name: Net::RDAP

Description: An RDAP library for Perl.

Level of maturity: Mature

Coverage: All aspects of the protocol are implemented.

Licensing: Perl (Artistic License or GPL).

Contact: gavin.brown@fastmail.uk

URL: https://metacpan.org/pod/Net::RDAP

7.3. rdapper

Name: rdapper

Description: A command-line RDAP client that uses Net::RDAP.

Level of maturity: Mature

Coverage: All aspects of the protocol are implemented.

Licensing: Perl (Artistic License or GPL).

Contact: gavin.brown@fastmail.uk

URL: https://metacpan.org/pod/App::rdapper

8. Change Log

This section is to be removed before publishing as an RFC.

8.1. Changes from 11 to 12

Updates as per IESG evaluation and advice from the RFC Editor:

1. Use a <referencegroup> element to reference [STD95].

2. Fix the examples by adding rdapConformacne properties.

3. Add type attributes to <sourcecode> elements.

4. Changed Section 5 to past tense.

8.2. Changes from 10 to 11

Updates as per IESG evaluation:

1. Add some context in Section 1 and reference [STD95] (thanks Mike Bishop).

2. Minor wording changes to Section 6 (thanks Ralf Weber and Mahesh Jethanandani).

3. Improve readability (thanks Mike Bishop).

4. Add sentence to Section 3 clarifying that the TTL values must be those stored in the registry database (thanks Mahesh Jethanandani).

5. Clarify (in Section 3) that JSON numbers representing TTL values cannot have fractional or exponent components (thanks Mahesh Jethanandani).

6. Downgrade one SHOULD to "should" and qualify a SHOULD in Section 4.2 (thanks テ詠ic Vyncke).

8.3. Changes from 09 to 10

Updates as per IESG evaluation:

1. Minor typographic edits.

2. Made [RFC9499] a normative reference (thanks Ketan Talaulikar).

8.4. Changes from 08 to 09

Updates as per directorate review:

1. Improved wording in the last sentence of the first paragraph of Section 1.

2. Add SHOULD keyword in Section 4.2 to clarify the need to carve out the "values" property of "ttl0_data" elements.

3. Updated Section 9.

8.5. Changes from 07 to 08

Updates as per AD review:

1. Added Section 4.
2. Move the last paragraph of Section 1 to Section 4.
3. Remove most of Section 3.1, moved some of it to Section 4.
4. Backfilled the change log immediately below.

8.6. Changes from 06 to 07

1. Correct reference to [RFC9499].

8.7. Changes from 05 to 06

1. Add missing change log entries to assist in IESG review.
2. Added Section 7.

8.8. Changes from 04 to 05

1. Clarify that TTL values are represented using JSON numbers.

8.9. Changes from 03 to 04

Updates based on feedback during WGLC:

1. Remove the xref in the Abstract to avoid issues when rendered as plaintext;
2. Refer to RRsets instead of resource records since [RFC2181] requires the TTL of all the records in an RRset to be the same;
3. Reference [RFC2181] which defines what a TTL is.
4. Clarify that clients need to handle (even if to ignore) unexpected record types.

8.10. Changes from 02 to 03

1. Switch to array model to object model, based on WG feedback.
2. Added Section 9.

8.11. Changes from 01 to 02

1. Include reference to [IANA-RRTYPES] in Section 3 (thanks Jasdip Singh).

2. Removed the value member of the link object in the example domain object so that it conforms with web linking practice (also thanks Jasdip Singh).

8.12. Changes from 00 to 01

1. Updated the extension identifier and extension property name to align with the current best practices in [I-D.ietf-regext-rdap-extensions].

2. Added Section 6.

3. Changed MUST to MAY in the first paragraph of Section 3.

4. Reduce ambiguity around the repetition of DNS record types in responses.

8.13. Changes from draft-brown-rdap-ttl-extension-03 to draft-ietf-regext-rdap-ttl-extension-00

1. Name change only.

8.14. Changes from 02 to 03

1. Update reference to [RFC9803].

8.15. Changes from 01 to 02

1. Update reference to the EPP extension.

8.16. Changes from 00 to 01

1. Extension property name renamed to ttl.

2. The extension data structure is now an array allowing common TTL values, remarks and events to be mapped to multiple DNS record types.

3. The extension data structure may now include remarks and events.

4. Added normative text regarding the value of DNS record mnemonics and TTL values.

9. Acknowledgements

The author wishes to thank the following for their constructive feedback and advice during the development of this document:

Andy Newton, Pawel Kowalik, Maarten Wullink, Mohamed Boucadair, Vijay K. Gurbani, Di Ma, Nabeel Cocker, Ketan Talaulikar, Ralf Weber, Mike Bishop, Mahesh Jethanandani, テ詠ic Vyncke, Sarah Tarrant.

10. References

10.1. Normative References

[IANA-RDAP-EXTENSIONS]
IANA, "RDAP Extensions", <https://www.iana.org/assignments/rdap-extensions/rdap-extensions.xhtml>.

[IANA-RRTYPES]
IANA, "Resource Record (RR) TYPEs", <https://www.iana.org/assignments/dns-parameters#dns-parameters-4>.

[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>.

[RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, DOI 10.17487/RFC2181, July 1997, <https://www.rfc-editor.org/info/rfc2181>.

[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>.

[RFC9499] Hoffman, P. and K. Fujiwara, "DNS Terminology", BCP 219, RFC 9499, DOI 10.17487/RFC9499, March 2024, <https://www.rfc-editor.org/info/rfc9499>.

[STD95] Internet Standard 95, <https://www.rfc-editor.org/info/std95/>.
At the time of writing, this STD comprises the following:

Newton, A., Ellacott, B., and N. Kong, "HTTP Usage in the Registration Data Access Protocol (RDAP)", STD 95, RFC 7480, DOI 10.17487/RFC7480, March 2015, <https://www.rfc-editor.org/info/rfc7480>.

Hollenbeck, S. and N. Kong, "Security Services for the Registration Data Access Protocol (RDAP)", STD 95, RFC 7481, DOI 10.17487/RFC7481, March 2015, <https://www.rfc-editor.org/info/rfc7481>.

Hollenbeck, S. and A. Newton, "Registration Data Access Protocol (RDAP) Query Format", STD 95, RFC 9082, DOI 10.17487/RFC9082, June 2021, <https://www.rfc-editor.org/info/rfc9082>.

Hollenbeck, S. and A. Newton, "JSON Responses for the Registration Data Access Protocol (RDAP)", STD 95, RFC 9083, DOI 10.17487/RFC9083, June 2021, <https://www.rfc-editor.org/info/rfc9083>.

Blanchet, M., "Finding the Authoritative Registration Data Access Protocol (RDAP) Service", STD 95, RFC 9224, DOI 10.17487/RFC9224, March 2022, <https://www.rfc-editor.org/info/rfc9224>.

10.2. Informative References

[I-D.ietf-regext-rdap-extensions]
Newton, A., Singh, J., and T. Harrison, "RDAP Extensions", Work in Progress, Internet-Draft, draft-ietf-regext-rdap-extensions-12, 18 February 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-regext-rdap-extensions-12>.

[RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)", STD 69, RFC 5730, DOI 10.17487/RFC5730, August 2009, <https://www.rfc-editor.org/info/rfc5730>.

[RFC9803] Brown, G., "Extensible Provisioning Protocol (EPP) Mapping for DNS Time-to-Live (TTL) Values", RFC 9803, DOI 10.17487/RFC9803, June 2025, <https://www.rfc-editor.org/info/rfc9803>.

Author’s Address

Gavin Brown
ICANN
12025 Waterfront Drive, Suite 300
Los Angeles, CA 90094-2536
United States of America
Email: gavin.brown@icann.org