

DNSOP
Internet-Draft
Intended status: Standards Track
Expires: 30 September 2024

D. Eastlake
H. Song
Futurewei Technologies, Inc.
29 March 2024

Expressing Quality of Service Requirements (QoS) in Domain Name System
(DNS) Queries
draft-eastlake-dnsop-expressing-qos-requirements-04

Abstract

A method of encoding quality of communication service (QoS) requirements in a Domain Name System (DNS) query is specified through inclusion of the requirements in one or more labels of the name being queried. This enables DNS responses including addressing and packet labeling information that is dependent on such requirements without changes in the format of DNS protocol messages or DNS application program interfaces (APIs).

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 30 September 2024.

Copyright Notice

Copyright (c) 2024 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
1.1. Terminology and Acronyms	3
2. Including Service Requirements in DNS Queries	4
2.1. Including Information in DNS Queries	4
2.2. Encoding Service Requirements in DNS Names	5
2.2.1. Service Requirement TLV Encoding	5
2.2.2. Requirements Types and Value Encoding	6
2.2.3. Complete QoS DNS Names	7
3. Security Considerations	8
4. IANA Considerations	8
4.1. Restricted LDH Label Prefixes	8
4.1.1. R-LDH Registry	9
4.1.2. R-LDH Expert Guidance	9
4.2. Requirements Label Type Codes	10
4.2.1. Coarse Requirements Label Values	11
5. Acknowledgments	11
6. Normative References	11
7. Informative References	12
Authors' Addresses	13

1. Introduction

The Domain Name System (DNS, [RFC1034] [RFC1035]) is a distributed database that stores data under hierarchical domain names and supports redundant servers, data caching, and security features. The data is formatted into resource records (RRs) whose content type and structure are indicated by the RR Type field. A typical use of DNS is that, by implementing the DNS protocol, a host can retrieve IP addresses stored at a domain name from DNS servers through that host's DNS resolver. Many other types of data besides IP addresses can be stored in and returned by the DNS.

There are instances where different DNS answers are desired depending on the type of destination service to be connected to and/or the communication protocol to be used for that communication. This can be indicated in a query through the use of designated initial labels beginning with the underscore codepoint ("_", 0x5F). This was

initially specified for the SRV RR Type [RFC2782]. For example, a query for type SRV to DNS name `_ldap._tcp.example.com` requests information on connecting to the `example.com` LDAP service with the TCP transport. This underscore label prefix method has been extended with additional types of leading-underscore labels for use with the TLSA, URI, TXT, and other RR Types [RFC8552].

Similarly, there is a need to encode different communication service quality requirements in DNS queries. Then different DNS answers can be returned depending, for example, on whether high bandwidth or low delay is the most important factor in the communication. Different answers could cause packets to be handled, constructed, or addressed differently which in turn could affect the path taken and/or the behavior of network switches along the communications path so as to be more likely to satisfy the desired communication service requirements.

Such encoding into the name being queried ensures that requirements will be forwarded by any recursive DNS servers between the querying resolver and the responding authoritative server. It also avoids any change in DNS protocol messages or application program interfaces (APIs).

This document specifies how quality of communication service requirements may be encoded in DNS queries through inclusion of the requirements in one or more labels of the name being queried enabling an authoritative server to take such requirements into account in determining its answers.

1.1. Terminology and Acronyms

The following terminology and acronyms are used in this document. General familiarity with DNS terminology [RFC8499] is assumed.

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.

ABNF - Augmented Backus-Naur Form [RFC5234].

API - Application Program Interface.

DNS - Domain Name System.

LDH - Letters, Digits, and Hyphen (DNS label) [RFC5890].

R-LDH - Restricted LDH (DNS label) [RFC5890].

RR - Resource Record [RFC8499]. This unit of data stored in the DNS.

TLV - Type, Length, Value.

2. Including Service Requirements in DNS Queries

This section specifies how to encode quality of communication service requirements in one or more domain name labels and discusses why some alternatives methods of including requirements in a DNS query are less desirable.

2.1. Including Information in DNS Queries

There exist methods to include information in a DNS request that are conveyed only from a resolver to a server, that is, one DNS hop. These are primarily through the inclusion of "meta-RRs" in the Additional Information section of a DNS request [RFC1035] including the OPT meta-RR [RFC6891] which can carry an extensible set of options. These methods are generally not suitable to use for the inclusion of QoS requirements for two reasons:

- * Typical APIs do not provide for meta-RRs to be specified on a query or retrieved from a response.
- * Because meta-RRs designate transient data associated with a particular DNS message. Thus, if a query is forwarded by a recursive DNS server, such requirements will be lost.

Other methods of including information in a DNS query that are preserved when a query is forwarded are the Name, Class, and RR Type.

Class is an additional dimension of DNS data besides Name and RR Type. However, only the "IN" or Internet Class has significant deployment or utilization and DNS messages specifying other Classes are frequently blocked by middle-boxes. Thus this dimension is not useful in practice.

RR Type is only 16-bits and is already used to indicate the type of RRs being requested.

This leaves only the name being queried for the encoding of service requirement as specified below.

2.2. Encoding Service Requirements in DNS Names

Domain names consist of a sequence of labels, with labels further to the right being a higher level in the name hierarchy and labels to the left of a particular label identifying nodes in the hierarchical tree below that particular label. Each label is limited to 63 octets in length and the zero length null label is reserved to identify the root node. In a complete, valid domain name, the sum of the length of each label in the name plus one octet of overhead per label (including the terminating null label) cannot exceed 255 octets.

Communication service requirements are encoded into names being queried. This is done by including a QoS label, constructed as described below, in the name, usually as the left most label. A QoS label consists of a special prefix followed by a sequence of one or more encoded TLVs indicating the QoS requirements. The use of such a special prefix, which affects the interpretation of the remainder of the label, is similar to the use of the "xn--" prefix to indicate internationalized domain names [RFC5890].

2.2.1. Service Requirement TLV Encoding

Each TLV expressing a service requirement can be thought of as being binarily encoded as shown in Figure 1 although the specified encoding below in a DNS label is more readable.

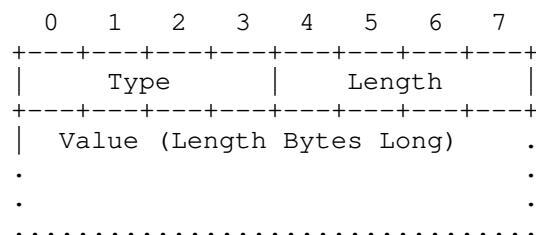


Figure 1: Service Requirement TLV Structure

Type: 4-bit unsigned integer indicating the type of service requirement.

Length: 4-bit unsigned integer indicating the length of the value associated with the service requirement in bytes. The presence of an explicit length makes it possible to skip unknown / unimplemented service requirements.

Value: The value, if any, associated with the service requirement.

Although the DNS does not constraint the octet values within a label, for ease of use and due to user interface restrictions, label octets are commonly limited to a subset of printing ASCII [RFC0020] character values. Furthermore, for name matching purposes, the DNS does not distinguish between octets having the upper case and lower case codes for an ASCII letter and in some cases the storage of a label in the DNS and/or its later retrieval may change the value of an octet in that label between the values for upper and lower case version of an ASCII letter [RFC4343].

To avoid possible problems with this DNS case insensitivity or possibly problematic byte values such as zero, the TLV or sequence of TLVs is included in the DNS name label in hexadecimal notation with one hex digit per byte using ASCII [RFC0020]. For "A" through "F", either upper or lower case may be used. Although there are more compact encoding that avoid most of these problems, for example as a customization of Bootstring similar to Punycode [RFC3492] or Base32 [RFC4648], hexadecimal is used for simplicity, to make the encoding into names more easily readable for debugging and other purposes, and to provide ample reserved code points for future extensions.

Such future extensions MUST use the same four prefix bytes and be structured as TLVs but may assign meaning to Type byte values reserved in this document and may extend the meaning of the Length to accomodate longer values by allowing letters "G" through "Z" (or "g" through "z") indicating Value length of 17 through 36 bytes. Length byte MUST NOT be any value than "0" through "9", "A" through "Z", or "a" through "z" and if such a prohibited value occurs, that TLV and the remainder of the label MUST be ignored.

2.2.2. Requirements Types and Value Encoding

The following types of QoS requirements are initially defined. If more than one requirements TLV of the same type occurs in a DNS name, all but the first (leftmost) occurrence MUST be ignored.

Coarse: A general indication of the most important service being sought encoded as a one byte integer patterned after the IPv4 ToS (Type of Service) value specified in [RFC1349]. (This is "coarse" in contrast with the more precise service requirements defined further below.) The following coarse values are defined:

0x00 - Normal service.

0x01 - Minimize cost.

0x02 - Maximize reliability.

0x04 - Maximize throughput.

0x08 - Minimize delay.

0x10 - Minimize jitter.

Bandwidth: The bandwidth requirement is encoded as a float32 (32-bit IEEE floating point format [ieee754] number). The unit is bits per second.

Delay: The delay requirement is encoded in 24-bit integer format. The unit is microseconds.

Jitter: The jitter (i.e., delay variation) is encoded in 24-bit integer format. The unit is microseconds.

Loss Rate: This lost rate (i.e., the percentage of packet loss) is encoded in 24-bit integer format. The basic unit is 0.000001% (i.e., one packet drop per 100 million packets), where $(2^{24} - 2) = 16.777214\%$ is the largest loss rate defined, $2^{24}-1$ means no loss rate requirement, and 0 means the drop rate should be smaller than 0.000001%.

Using IEEE 32-bit floating point for the values when appropriate provides a compact notation that can encode up to approximately 10^{-38} and down to approximately 10^{-38} with 6 to 9 significant digits of precision [ieee754].

2.2.3. Complete QoS DNS Names

The on-the-wire encoding of a domain name beginning with a service requirement label would be as shown in Figure 2 below. (In the DNS wire encoding, each label is preceded by a byte that indicates its length.)

```
+-----+-----+-----+ +-----+-----+-----+-----+
|length|prefix|TLV1|...|TLVn|Encoded Remainder of Domain Name|
+-----+-----+-----+ +-----+-----+-----+-----+
```

Figure 2: Name Wire Encoding Style 1

Alternatively, service requirements could split among a sequence of two or more labels in a DNS name to be queried, as shown in Figure 3.

```
+-----+-----+-----+ +-----+-----+-----+-----+
|length|prefix|TLV1|...|length|prefix|TLVn|Remainder of Name|
+-----+-----+-----+ +-----+-----+-----+-----+
```

Figure 3: Name Encoding Style 2

A display presentation of a DNS name requesting a coarse QoS requirement for minimum delay for communication with example.com would be as shown in Figure 4.

```

      qs-- Prefix
        1 TLV Type
        1 TLV Length
       08 TLV Value
example.com Remainder of domain name

qs--1108.example.com. Complete domain name

```

Figure 4: Example DNS Name

3. Security Considerations

TBD

4. IANA Considerations

This section conforms to [RFC8126].

IANA is requested to create the following registries.

4.1. Restricted LDH Label Prefixes

LDH labels are specified in [RFC5890] as consisting of letters, digits, and hyphen but not beginning or ending with a hyphen. That is, strings of length from 1 through 63 that match the ABNF (Augmented Backus-Naur Form [RFC5234]) expression for LDH below.

```
* LD = ( a-z / 0-9 ) ;letter or digit (case insensitive)
```

```
* HYPH = %x2D ;hyphen / minus
```

```
* LDH = LD / HYPH
```

```
* LDH-LABEL = LD / LD 0*61LDH LD
```

R-LDH (Restricted LDH) labels are specified in [RFC5890] as the subset of LDH-LABELs that begin with two letters/digits followed by two hyphens. That is, they are LDH-LABELs that match the ABNF regular expression [RFC5234] below.

```
* R-LDH-LABEL = 2LD HYPH HYPH 0*58LDH LD
```


4.1.1. R-LDH Registry

IANA is requested to create a registry on the Domain Name System (DNS) Parameters webpage as follows:

Name: DNS Restricted LDH (R-LDH) Label Prefixes

Registration Procedure: Expert Review

Reference: [this document]

Prefix	Description	Reference
qs--	QoS Requirements	[this document]
xn--	Internationalization	[RFC5890]

Table 1

4.1.2. R-LDH Expert Guidance

In reviewing applications for the assignment of an R-LDH prefix, the Expert should keep in mind the following guidance:

1. The use of labels with the requested prefix must be documented in an Internet Draft or RFC,
2. not significantly duplicate the use of any other R-LDH prefix,
3. not require any changes to DNS protocol messages or DNS mechanisms such as the handling of CNAME or DNAME RRs or wildcards, and
4. provide a substantial additional capability.
5. Prefixes where the first or second character is any of the digits "0", "1", and "5" or the letters "O", "I", and "L" should not be assigned, due to the possibilities of confusion, unless there are strong reasons to use these characters.
6. Assignment of more than one R-LDH for a purpose is prohibited. The remainder of an R-LDH label MUST include an appropriate extension mechanism such as a version number or multiple unassigned code points such that later versions or extensions can be accodated without the assignment of a new R-LDH label. If it is necessary to distinguish sub-uses under an R-LDH prefix, this

should be done by encoding within the R-LDH label after the prefix or by a further label or labels before and/or after the R-LDH label, such as a label beginning with underscore ("_").

4.2. Requirements Label Type Codes

IANA is requested to create a registry on the Domain Name System (DNS) Parameters webpage as follows:

Name: DNS QoS Requirements Label Type Codes

Registration Procedure: Expert review

Reference: [this document]

Code	Hex	Description	Reference
-	0x00-0x2F	reserved	
0	0x30	reserved	
1	0x31	Coarse QoS	[this document]
2	0x32	Bandwidth	[this document]
3	0x33	Delay	[this document]
4	0x34	Jitter	[this document]
5	0x35	Loss Rate	[this document]
6-9	0x33-0x39	unassigned	
-	0x3A-0x40	reserved	
10-15	0x41-0x46	unassigned	
-	0x47-0x60	reserved	
10-15	0x61-0x66	unassigned	
-	0x67-0xFF	reserved	

Table 2

4.2.1. Coarse Requirements Label Values

IANA is requested to create a sub-registry on the Domain Name System (DNS) Parameters webpage under the Requirements Label Type Codes registry as follows:

Name: DNS QoS Coarse Requirements Label Values

Registration Procedure: Expert review

Reference: [this document]

Value	Description	Reference
0x00	Normal service	[this document]
0x01	Mimimize cost	[this document]
0x02	Maximize reliability	[this document]
0x04	Maximize throughput	[this document]
0x08	Minimize delay	[this document]
0x10	Minimize jitter	[this document]
Other Values	unassigned	

Table 3

5. Acknowledgments

The suggestions of the following are gratefully acknowledged:

* TBD

6. Normative References

- [ieee754] IEEE 754 WG, IEEE., "IEEE 754-2019 - IEEE Standard for Floating-Point Arithmetic", 2019,
<<https://standards.ieee.org/standard/754-2019.html>>.
- [RFC0020] Cerf, V., "ASCII format for network interchange", STD 80, RFC 20, DOI 10.17487/RFC0020, October 1969,
<<https://www.rfc-editor.org/info/rfc20>>.

- [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>>.
- [RFC4343] Eastlake 3rd, D., "Domain Name System (DNS) Case Insensitivity Clarification", RFC 4343, DOI 10.17487/RFC4343, January 2006, <<https://www.rfc-editor.org/info/rfc4343>>.
- [RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <<https://www.rfc-editor.org/info/rfc5234>>.
- [RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, DOI 10.17487/RFC5890, August 2010, <<https://www.rfc-editor.org/info/rfc5890>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [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>>.

7. Informative References

- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [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>>.
- [RFC1349] Almquist, P., "Type of Service in the Internet Protocol Suite", RFC 1349, DOI 10.17487/RFC1349, July 1992, <<https://www.rfc-editor.org/info/rfc1349>>.
- [RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, DOI 10.17487/RFC2782, February 2000, <<https://www.rfc-editor.org/info/rfc2782>>.

- [RFC3492] Costello, A., "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", RFC 3492, DOI 10.17487/RFC3492, March 2003, <<https://www.rfc-editor.org/info/rfc3492>>.
- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <<https://www.rfc-editor.org/info/rfc4648>>.
- [RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <<https://www.rfc-editor.org/info/rfc6891>>.
- [RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", RFC 8499, DOI 10.17487/RFC8499, January 2019, <<https://www.rfc-editor.org/info/rfc8499>>.
- [RFC8552] Crocker, D., "Scoped Interpretation of DNS Resource Records through "Underscored" Naming of Attribute Leaves", BCP 222, RFC 8552, DOI 10.17487/RFC8552, March 2019, <<https://www.rfc-editor.org/info/rfc8552>>.

Authors' Addresses

Donald Eastlake
Futurewei Technologies, Inc.
2386 Panoramic Circle
Apopka, FL 32703
United States of America
Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Haoyu Song
Futurewei Technologies, Inc.
2220 Central Expressway
Santa Clara, CA 95050
United States of America
Email: haoyu.song@futurewei.com