

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: September 26, 2015

P. Faltstrom  
Netnod  
O. Kolkman  
ISOC  
March 25, 2015

**The Uniform Resource Identifier (URI) DNS Resource Record  
draft-faltstrom-uri-14**

Abstract

This document describes the already registered DNS resource record type called the Uniform Resource Identifier (URI) RR, for publishing mappings from hostnames to URIs.

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 <http://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 September 26, 2015.

Copyright Notice

Copyright (c) 2015 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 (<http://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 the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	<a href="#">Introduction</a>	<a href="#">2</a>
<a href="#">2.</a>	<a href="#">Applicability Statement</a>	<a href="#">3</a>
<a href="#">3.</a>	<a href="#">DNS considerations</a>	<a href="#">3</a>
<a href="#">4.</a>	<a href="#">The format of the URI RR</a>	<a href="#">4</a>
<a href="#">4.1.</a>	<a href="#">Ownername, class and type</a>	<a href="#">4</a>
<a href="#">4.2.</a>	<a href="#">Priority</a>	<a href="#">5</a>
<a href="#">4.3.</a>	<a href="#">Weight</a>	<a href="#">5</a>
<a href="#">4.4.</a>	<a href="#">Target</a>	<a href="#">5</a>
<a href="#">4.5.</a>	<a href="#">URI RDATA Wire Format</a>	<a href="#">5</a>
<a href="#">5.</a>	<a href="#">Usages</a>	<a href="#">6</a>
<a href="#">5.1.</a>	<a href="#">Example: FTP server in the example.com domain</a>	<a href="#">6</a>
<a href="#">5.2.</a>	<a href="#">Relation to S-NAPTR</a>	<a href="#">6</a>
<a href="#">5.3.</a>	<a href="#">Relation to U-NAPTR</a>	<a href="#">7</a>
<a href="#">5.4.</a>	<a href="#">Relation to SRV</a>	<a href="#">7</a>
<a href="#">6.</a>	<a href="#">IANA Considerations</a>	<a href="#">7</a>
<a href="#">6.1.</a>	<a href="#">Registration of the URI Resource Record Type</a>	<a href="#">7</a>
<a href="#">6.2.</a>	<a href="#">Registration of services</a>	<a href="#">7</a>
<a href="#">7.</a>	<a href="#">Security Considerations</a>	<a href="#">7</a>
<a href="#">8.</a>	<a href="#">Acknowledgements</a>	<a href="#">8</a>
<a href="#">9.</a>	<a href="#">References</a>	<a href="#">9</a>
<a href="#">9.1.</a>	<a href="#">Normative References</a>	<a href="#">9</a>
<a href="#">9.2.</a>	<a href="#">Non-normative references</a>	<a href="#">9</a>
<a href="#">Appendix A.</a>	<a href="#">The original RRTYPE Allocation Request</a>	<a href="#">10</a>
<a href="#">Authors' Addresses</a>		<a href="#">13</a>

## [1.](#) Introduction

This document explains the use of the Domain Name System (DNS) for the storage of URIs [[RFC3986](#)], and how to resolve hostnames to such URIs that can be used by various applications using the URI Resource Record Type. For resolution the application need to know both the hostname and the protocol that the URI is to be used for. The protocol is registered by IANA.

Historically, uses of the DNS to map a domain name to a URL have relied on the NAPTR RRTYPES and then on the DDDS [[RFC3401](#)] application framework with the DNS as a database as specified in [RFC 3404](#) [[RFC3404](#)]. This has a number of implications such as the fact the RRSet returned will contain all URIs "connected" with the owner, and not only the ones related to a specific service.

The URI resource record specified in this document enables the querying party to do the equivalence of selecting which ones of the NAPTR records one is interested in, and have only those returned. This because data in the service field of the NAPTR record is included in the owner part of the URI resource record type. It is



also the case that as the URI resource record type include the target URI directly as part of the RDATA, it is very easy to extract the correct target URI, instead of applying rewrite rules as in NAPTR.

Querying for URI resource records is not replacing querying for NAPTR resource records (or use of S-NAPTR [[RFC3958](#)]). Instead, the URI resource record type provides a complementary mechanism to use when one already knows what service field is interesting. With it, one can directly query for the specific subset of the otherwise possibly large RRSset given back when querying for NAPTR resource records.

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 [BCP 14](#), [RFC 2119](#) [[RFC2119](#)].

## **2. Applicability Statement**

In general, it is expected that URI records will be used by clients for applications where the relevant protocol to be used is known, but, for example, an extra abstraction is needed in order to separate a domain name from a point of service (as addressed by the URI). One example of such a situation is when an organisation has many domain names, but only one official web page.

Applications MUST know the specific service to prepend the hostname with. Using repetitive queries for URI records MUST NOT be a replacement for querying for NAPTR records according to the NAPTR (DDDS) or S-NAPTR algorithms. NAPTR records serve the purpose to discover the various services and URIs for looking up access points for a given service. Those are two very different kinds of needs.

## **3. DNS considerations**

Using prefix labels, such as underscored service tags, for a specific owner name may cause a counter-intuitive effect when the owner name is a wildcard name. For example, `_s2._s1.*.example.net.` is not a wildcard name and cannot be used to return a synthesized answer for a query name of `_s2._s1.a.example.net.` See [Section 4.5 of RFC4592](#) [[RFC4592](#)] for more details. Besides, underscored service tags used for the URI RR (based on the Service Name and Transport Protocol Port Number Registry) may have slightly different semantics than service tags used for underscored prefix labels that are used in combination with other (yet unspecified) RR types. This may cause subtle management problems when delegation structure that has developed within the context of URI RRs is also to be used for other RR types. Since the service labels might be overloaded, applications should



carefully check that the application level protocol is indeed the protocol they expect.

Subtle management issues may also arise when the delegations from service to sub service label involves several parties and different stake holders.

#### **4. The format of the URI RR**

This is the presentation format of the URI RR:

```
Ownername TTL Class URI Priority Weight Target
```

The URI RR does not cause any kind of Additional Section processing.

##### **4.1. Ownername, class and type**

The URI ownername is subject to special conventions.

Just like the SRV RR [[RFC2782](#)] the URI RR has service information encoded in its ownername. In order to encode the service for a specific owner name one uses service parameters. Valid service parameters used are those registered by IANA in the Service Name and Transport Protocol Port Number Registry [[RFC6335](#)], or as Enumservice Registrations [[RFC6117](#)]. The Enumservice Registration parameters are reversed (subtype(s) before type), prepended with an underscore (\_) and prepended to the owner name in separate labels. The underscore is prepended to the service parameters to avoid collisions with DNS labels that occur in nature, and the order is reversed to make it possible to do delegations, if needed, to different zones (and therefore providers of DNS).

For example, suppose we are looking for the URI for a service with ENUM Service Parameter "A:B:C" for host example.com. Then we would query for (QNAME,QTYPE)=("\_C.\_B.\_A.example.com","URI")

As another example, suppose we are looking for the URI for a service with Service Name "A" and Transport Protocol "B" for host example.com. Then we would query for (QNAME,QTYPE)=("\_A.\_B.example.com","URI")

The type number for the URI record is 256.

The URI resource record is class independent.

The URI RR has no special TTL requirements.



#### **4.2. Priority**

The priority of the target URI in this RR. Its range is 0-65535. A client **MUST** attempt to contact the URI with the lowest-numbered priority it can reach; URIs with the same priority **SHOULD** be selected according to propabilities defined by the weight field.

#### **4.3. Weight**

A server selection mechanism. The weight field specifies a relative weight for entries with the same priority. Larger weights **SHOULD** be given a proportionately higher probability of being selected. The range of this number is 0-65535.

#### **4.4. Target**

The URI of the target, enclosed in double-quote characters (""), where the URI is as specified in [RFC 3986](#) [[RFC3986](#)]. Resolution of the URI is according to the definitions for the Scheme of the URI.

Since the URI will not be encoded as a <character-string> (see [RFC1035 section 3.3](#) [[RFC1035](#)]) there is no 255 character size limitation.

The Target **MUST NOT** be an empty URI (").

#### **4.5. URI RDATA Wire Format**

The RDATA for a URI RR consists of a 2 octet Priority field, a two octet Weight field, and a variable length target field.

Priority and Weight are unsigned integers in network byte order.

The remaining data in the RDATA contains the Target field. The Target field contains the URI as a sequence of octets (without the enclosing double- quote characters used in the presentation format).

The length of the target field **MUST** be greater than zero.



```

          1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Priority          |          Weight          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
/
/
/
/
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

## 5. Usages

### 5.1. Example: FTP server in the example.com domain

An organisation has the domain names example.com and example.net, and their FTP archive is at ftp://ftp1.example.com/public. Given the Service Name "ftp" and Transport Protocol "tcp" (from the IANA registry of Service Name and Transport Protocol Port Numbers), the following URI Resource Records could be made available in the respective zones (example.com and example.net):

```

$ORIGIN example.com.
_ftp._tcp      IN URI 10 1 "ftp://ftp1.example.com/public"

$ORIGIN example.net.
_ftp._tcp      IN URI 10 1 "ftp://ftp1.example.com/public"

```

### 5.2. Relation to S-NAPTR

The URI resource record type is not a replacement for the S-NAPTR. It is instead an extension and the second step of the S-NAPTR resolution can resolve a URI resource record instead of using SRV records and yet another algorithm for how to use SRV records for the specific protocol.

```

$ORIGIN example.com.
;;      order pref flags
  IN NAPTR 100    10    "D"    "EM:ProtA"          ( ; service
                                     ""              ; regexp
                                     _http._tcp.example.com. ; replacement
_http._tcp IN URI 10 1 "http://www.example.com/path"

```



### **5.3. Relation to U-NAPTR**

The URI Resource Record Type, together with S-NAPTR, can be viewed as a replacement for U-NAPTR [[RFC4848](#)]. The URI Resource Record Type is though only interesting when one know a base domain name, a protocol and service so that one can compose the record to look up. NAPTR records of any kind are used to look up what services exists for a certain domain, which is one step before the URI resource record is used.

### **5.4. Relation to SRV**

The URI Resource Record Type can be viewed as a replacement for the SRV record. This because it like the SRV record can only be looked up if one know the base domain, the protocol and the service. It has a similar functionality, and uses the same registry for Service Names, but instead of returning a hostname and port number, the URI record return a full URI. As such, it can be viewed as a more powerful resource record than SRV.

## **6. IANA Considerations**

### **6.1. Registration of the URI Resource Record Type**

After an expert review in February 2011 (see [Appendix A](#)) IANA has allocated RRTYPE 256 for the URI Resource Record Type in the registry named Resource Record (RR) TYPEs and QTYPEs as defined in [BCP 42](#) (at the time [RFC 6195](#) [[RFC6195](#)]), located at <http://www.iana.org/assignments/dns-parameters>.

IANA is requested to update the reference with that registration to this RFC.

### **6.2. Registration of services**

No new registry is needed for the registration of services as the Service Name, Transport Protocol Port Numbers, Enumservices and the DNS SRV Service Type registries are used also for the URI resource record type.

## **7. Security Considerations**

Using the URI resource record together with security mechanisms that relies on verification of authentication of hostnames, like TLS, makes it important to choose the correct domain name when doing the comparison, and that the change in what hostname to use is secured by DNSSEC so that it can be trusted in a similar way as a redirect in HTTP using TLS.



If for example the URI resource record is not signed with the help of DNSSEC, and then validated successfully, trusting the non-signed URI will effectly lead to a downgrade attack.

The basic mechanism for successful use of URI works as follows:

1. Announce the fact example.com is hosted at example.org (with some URL) in DNS
2. Secure the URI resource record with DNSSEC. Best of course by doing validation in the application doing the lookup, but it could also be in the local recursive resolver or in the trusted recursive resolver also doing validation. All according to the local trust policy.
3. Verify the TLS (for example) certificate for the connection to example.org matches, i.e. use the hostname in the URI and not the hostname used originally when looking up the URI resource record.
4. If needed, do application layer authentication etc over the then encrypted connection.

What also can happen is that the URI in the resource record type has errors in it. Applications using the URI resource record type for resolution should behave similarly as if the user typed (or copy and pasted) the URI. At least it must be clear to the user that the error is not due to any error from his side.

One SHOULD NOT include userinfo (see User Information, [Section 3.2.1](#), in [RFC 3986](#) [[RFC3986](#)]) in a URI that is used in a URI resource record as DNS data must be viewed as publicly available information.

## **8. Acknowledgements**

Ideas on how to split the two different kind of queries "What services exists for this domain name" and "What is the URI for this service" came from Scott Bradner and Lawrence Conroy. Other people that have contributed to this document include Richard Barnes, Leslie Daigle, Victor Dukhovni, Olafur Gudmundsson, Philip Hallam-Baker, Ted Hardie, Sam Hartman, Evan Hunt, John klensin, Peter Koch, Eliot Lear, Andy Newton, Mark Nottingham, Penn Pfautz, Jinmei Tatuya, Willem Toorop, Nico Williams.

Cisco is acknowledged as mr Faltstrom's employer at the time this document was developed.



The NLnet Labs is acknowledged as mr Kolkman's employer at the time this document was developed.

## **9. References**

### **9.1. Normative References**

- [RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, [RFC 1035](#), November 1987.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, [RFC 3629](#), November 2003.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), January 2005.
- [RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)", [RFC 3987](#), January 2005.
- [RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", [RFC 5890](#), August 2010.
- [RFC6117] Hoeneisen, B., Mayrhofer, A., and J. Livingood, "IANA Registration of Enumservices: Guide, Template, and IANA Considerations", [RFC 6117](#), March 2011.
- [RFC6195] Eastlake, D., "Domain Name System (DNS) IANA Considerations", [RFC 6195](#), March 2011.
- [RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", [BCP 165](#), [RFC 6335](#), August 2011.

### **9.2. Non-normative references**

- [RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", [RFC 2782](#), February 2000.
- [RFC3401] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part One: The Comprehensive DDDS", [RFC 3401](#), October 2002.



- [RFC3403] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database", [RFC 3403](#), October 2002.
- [RFC3404] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Four: The Uniform Resource Identifiers (URI)", [RFC 3404](#), October 2002.
- [RFC3597] Gustafsson, A., "Handling of Unknown DNS Resource Record (RR) Types", [RFC 3597](#), September 2003.
- [RFC3833] Atkins, D. and R. Austein, "Threat Analysis of the Domain Name System (DNS)", [RFC 3833](#), August 2004.
- [RFC3958] Daigle, L. and A. Newton, "Domain-Based Application Service Location Using SRV RRs and the Dynamic Delegation Discovery Service (DDDS)", [RFC 3958](#), January 2005.
- [RFC4592] Lewis, E., "The Role of Wildcards in the Domain Name System", [RFC 4592](#), July 2006.
- [RFC4848] Daigle, L., "Domain-Based Application Service Location Using URIs and the Dynamic Delegation Discovery Service (DDDS)", [RFC 4848](#), April 2007.
- [RFC5507] IAB, Faltstrom, P., Austein, R., and P. Koch, "Design Choices When Expanding the DNS", [RFC 5507](#), April 2009.

#### [Appendix A](#). The original RRTYPE Allocation Request

On February 22, 2011 IANA assigned RRTYPE 256 for the URI resource record based on a request that followed the procedure documented in [RFC 6195](#) [[RFC6195](#)]. The DNS RRTYPE PARAMETER ALLOCATION form as submitted to IANA at that time is replicated below for reference.

A. Submission Date:

May 23, 2009

B. Submission Type:

- ☒ New RRTYPE
- ☐ Modification to existing RRTYPE



C. Contact Information for submitter:

Name: Patrik Faltstrom

Email Address: paf@cisco.com

International telephone number: +46-8-6859131

Other contact handles:

(Note: This information will be publicly posted.)

D. Motivation for the new RRTYPE application?

There is no easy way to get from a domain name to a URI (or IRI). Some mechanisms exist via use of the NAPTR [[RFC3403](#)] resource record. That implies quite complicated rules that are simplified via the S-NAPTR [[RFC3958](#)] specification. But, the ability to directly look up a URI still exists. This specification uses a prefix based naming mechanism originated in the definition of the SRV [[RFC2782](#)] resource record, and the RDATA is a URI, encoded as one text field.

See also above ([Section 1](#)).

E. Description of the proposed RR type.

The format of the URI resource record is as follows:

Ownername TTL Class URI Priority Weight Target

The URI RR has service information encoded in its ownername. In order to encode the service for a specific owner name one uses service parameters. Valid service parameters used are either Enumservice Registrations registered by IANA, or prefixes used for the SRV resource record.

The wire format of the RDATA is as follows:



```

          1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Priority          |          Weight          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
/
/
/
/
/
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

- F. What existing RRTYPE or RRTYPEs come closest to filling that need and why are they unsatisfactory?

The RRTYPE that come closest is the NAPTR resource record. It is for example used in the DDS and S-NAPTR algorithms. The main problem with the NAPTR is that selection of what record (or records) one is interested in is based on data stored in the RDATA portion of the NAPTR resource record. This, as explained in [RFC 5507](#) [RFC5507], is not optimal for DNS lookups. Further, most applications using NAPTR resource records uses regular expression based rewrite rules for creation of the URI, and that has shown be complicated to implement.

The second closest RRTYPE is the SRV record that given a prefixed based naming just like is suggested for the URI resource record, one get back a port number and domain name. This can also be used for creation of a URI, but, only URIs without path components.

- G. What mnemonic is requested for the new RRTYPE (optional)?

URI

- H. Does the requested RRTYPE make use of any existing IANA Registry or require the creation of a new IANA sub-registry in DNS Parameters?

Yes, partially.



One of the mechanisms to select a service is to use the Enumservice Registry managed by IANA. Another is to use services and protocols used for SRV records.

- I. Does the proposal require/expect any changes in DNS servers/resolvers that prevent the new type from being processed as an unknown RRTYPE (see [RFC 3597](#) [[RFC3597](#)])?

No

- J. Comments:

None

#### Authors' Addresses

Patrik Faltstrom  
Netnod

Email: [paf@netnod.se](mailto:paf@netnod.se)

Olaf Kolkman  
Internet Society

Email: [kolkman@isoc.org](mailto:kolkman@isoc.org)

