Network Working Group
Request for Comments: 4501
Category: Standards Track

S. Josefsson
SJD

May 2006

Domain Name System Uniform Resource Identifiers

Status of This Memo

Copyright Notice

Abstract

This document defines Uniform Resource Identifiers for Domain Name System resources.

Table of Contents

1. Introduction and Background 2
2. Usage Model 2
3. DNS URI Registration 3
4. Examples 6
5. Acknowledgements 7
6. Security Considerations 7
7. IANA Considerations 7
8. Copying Conditions 8
9. References 8
9.1. Normative References 8
9.2. Informative References 8

1. Introduction and Background

The Domain Name System (DNS) [1] [2] is a widely deployed system used, among other things, to translate host names into IP addresses. Several protocols use Uniform Resource Identifiers (URIs) to refer to data. By defining a URI scheme for DNS data, the gap between these two worlds is bridged. The DNS URI scheme defined here can be used to reference any data stored in the DNS.

Data browsers may support DNS URIs by forming DNS queries and rendering DNS responses using HTML [12], which is similar to what is commonly done for FTP [6] resources. Applications that are Multipurpose Internet Mail Extensions (MIME) [7] aware may tag DNS data retrieved using this scheme with the text/dns or application/dns types as specified in [15].

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 RFC 2119 [3].

2. Usage Model

Refer to section 1 of [5] for an in-depth discussion of URI classifications. In particular, the reader is assumed to be familiar with the distinction between "name" and "locator". This section describes how the DNS URI scheme is intended to be used and outlines future work that may be required to use URIs with the DNS for some applications.

The URI scheme described in this document focuses on the data stored in the DNS. As such, there is no provision to specify any of the fields in the actual DNS protocol. This is intended so that the URI may be used even in situations where the DNS protocol is not used directly. Two examples for this are zone file editors and DNS-related configuration files, which may use this URI scheme to identify data. The application would not use the DNS protocol to resolve the URIs.

A limitation of this design is that it does not accommodate all protocol parameters within the DNS protocol. It is expected that, for certain applications, a more detailed URI syntax that maps more closely to the DNS protocol may be required. However, such a URI definition is not included in this document. This document specifies a URI that is primarily intended to name DNS resources, but it can also be used to locate said resources for simple, yet common, applications.

3. DNS URI Registration

This section contains the registration template for the DNS URI scheme in accordance with [11].

URL scheme name: "dns".

URL scheme syntax: A DNS URI designates a DNS resource record set, referenced by domain name, class, type, and, optionally, the authority. The DNS URI follows the generic syntax from RFC 3986 [5] and is described using ABNF [4]. Strings are not case sensitive, and free insertion of linear-white-space is not permitted.

```
dnsurl          = "dns:" [ "//" dnsauthority "/" ]
                  dnsname ["?" dnsquery]

dnsauthority    = host [ ":" port ]
                            ; See RFC 3986 for the
                            ; definition of "host" and "port".

dnsname         = *pchar
                            ; See RFC 3986 for the
                            ; definition of "pchar".

                            ; The "dnsname" field may be a
                            ; "relative" or "absolute" name,
                            ; as per RFC 1034, section 3.1.

                            ; Note further that an empty
                            ; "dnsname" value is to be
                            ; interpreted as the root itself.
                            ; See below on relative dnsnames.

dnsquery        = dnsqueryelement [";" dnsquery]

dnsqueryelement = ( "CLASS=" dnsclassval ) / ( "TYPE=" dnstypeval )
                            ; Each clause MUST NOT be used more
                            ; than once.

dnsclassval     = 1*digit / "IN" / "CH" /
                  <Any IANA registered DNS class mnemonic>

dnstypeval      = 1*digit / "A" / "NS" / "MD" /
                  <Any IANA registered DNS type mnemonic>
```

Unless specified in the URI, the authority ("dnsauthority") is assumed to be locally known, the class ("dnsclassval") to be the Internet class ("IN"), and the type ("dnstypeval") to be the Address

type ("A"). These default values match the typical use of DNS: to look up addresses for host names.

A dnsquery element MUST NOT contain more than one occurrence of the "CLASS" and "TYPE" fields. For example, both "dns:example?TYPE=A;TYPE=TXT" and "dns:example?TYPE=A;TYPE=A" are invalid. However, the fields may occur in any order, so that both "dns:example?TYPE=A;CLASS=IN" and "dns:example?CLASS=IN;TYPE=A" are valid.

The digit representation of types and classes MAY be used when a mnemonic for the corresponding value is not well known (e.g., for newly introduced types or classes), but it SHOULD NOT be used for the types or classes defined in the DNS specification [2]. All implementations MUST recognize the mnemonics defined in [2].

To avoid ambiguity, relative "dnsname" values (i.e., those not ending with ".") are assumed to be relative to the root. For example, "dns:host.example" and "dns:host.example." both refer to the same owner name; namely, "host.example.". Further, an empty "dnsname" value is considered a degenerative form of a relative name, which refers to the root (".").

To resolve a DNS URI using the DNS protocol [2], a query is created, using as input the dnsname, dnsclassval, and dnstypeval from the URI string (or the appropriate default values). If an authority ("dnsauthority") is given in the URI string, this indicates the server that should receive the DNS query. Otherwise, the default DNS server should receive it.

Note that DNS URIs could be resolved by other protocols than the DNS protocol, or by using the DNS protocol in some other way than as described above (e.g., multicast DNS). DNS URIs do not require the use of the DNS protocol, although it is expected to be the typical usage. The previous paragraph only illustrates how DNS URIs are resolved using the DNS protocol.

A client MAY want to check that it understands the dnsclassval and dnstypeval before sending a query, so that it will be able to understand the response. However, a typical example of a client that would not need to check dnsclassval and dnstypeval would be a proxy that would just treat the received answer as opaque data.

Character encoding considerations: Characters are encoded as per RFC 3986 [5]. The DNS protocol does not consider character sets; it simply transports opaque data. In particular, the "dnsname" field of the DNS URI is to be considered an internationalized domain name (IDN) unaware domain name slot, in the terminology of RFC 3940 [14]. The considerations for "host" and "port" are discussed in [5].

Because "." is used as the DNS label separator, an escaping mechanism is required to encode a "." that is part of a DNS label. The escaping mechanism is described in section 5.1 of RFC 1035 [2]. For example, a DNS label of "exa.mple" can be escaped as "exa\.mple" or "exa\046mple". However, the URI specification disallows the "\" character from occurring directly in URIs, so it must be escaped as "%5c". The single DNS label "exa.mple" is thus encoded as "exa%5c.mple". The same mechanism can be used to encode other characters, for example, "?" and ";". Note that "." and "%2e" are equivalent within dnsname and are interchangeable.

This URI specification allows all possible domain names to be encoded, provided the encoding rules are observed per [5]). However, certain applications may restrict the set of valid characters. Care should be taken so that invalid characters in these contexts do not cause harm. In particular, host names in the DNS have certain restrictions. It is up to these applications to limit this subset; this URI scheme places no restrictions.

Intended usage: Whenever it is useful for DNS resources to be referenced by protocol-independent identifiers. Often, this occurs when the data is more important than the access method. Since software in general has coped without this so far, it is not anticipated to be implemented widely, nor migrated to by existing systems, but specific solutions (especially security-related) may find this appropriate.

Applications and/or protocols that use this scheme include Security-related software, DNS administration tools, and network programming packages.

Interoperability considerations: The data referenced by this URI scheme might be transferred by protocols that are not URI aware (such as the DNS protocol). This is not anticipated to have any serious interoperability impact.

Interoperability problems may occur if one entity understands a new DNS class/type mnemonic that another entity does not. This is an interoperability problem for DNS software in general, although it is not a major practical problem for current DNS deployments, as the DNS types and classes are fairly static. To guarantee interoperability, implementations can use integers for all mnemonics not defined in [2].

Interaction with Binary Labels [10] or other extended label types has not been analyzed. However, binary labels appear to be infrequently used in practice.

Contact: simon@josefsson.org

Author/Change Controller: simon@josefsson.org

4. Examples

A DNS URI is of the following general form. This is intended to illustrate, not define, the scheme:

dns:[//authority/]domain[?CLASS=class;TYPE=type]

The following illustrates a URI for a resource with the absolute name "www.example.org.", the Internet (IN) class, and the Address (A) type:

dns:www.example.org.?clAsS=IN;tYpE=A

Since the default class is IN and the default type is A, the same resource can be identified by a shorter URI using a relative name:

dns:www.example.org

The following illustrates a URI for a resource with the name "simon.example.org" for the CERT type in the Internet (IN) class:

dns:simon.example.org?type=CERT

The following illustrates a URI for a resource with the name "ftp.example.org", in the Internet (IN) class and the address (A) type, but from the DNS authority 192.168.1.1 instead of the default authority:

dns://192.168.1.1/ftp.example.org?type=A

The following illustrates various escaping techniques. The owner name would be "world wide web.example\.domain.org", where "\." denotes the character "." as part of a label, and "." denotes the label separator:

dns:world%20wide%20web.example%5c.domain.org?TYPE=TXT

The following illustrates a strange but valid DNS resource:

dns://fw.example.org/*.%20%00.example?type=TXT

5. Acknowledgements

Thanks to Stuart Cheshire, Donald Eastlake, Pasi Eronen, Bill Fenner, Ted Hardie, Russ Housley, Peter Koch, Andrew Main, Larry Masinter, Michael Mealling, Steve Mattson, Marcos Sanz, Jason Sloderbeck, Paul Vixie, Sam Weiler, and Bert Wijnen for comments and suggestions. The author acknowledges RSA Laboratories for supporting the work that led to this document.

6. Security Considerations

If a DNS URI references domains in the Internet DNS environment, both the URI itself and the information referenced by the URI is public information. If a DNS URI is used within an "internal" DNS environment, both the DNS URI and the data referenced should be handled using the same considerations that apply to DNS data in the "internal" environment.

If information referenced by DNS URIs are used to make security decisions (such data includes, but is not limited to, certificates stored in the DNS [9]), implementations may need to employ security techniques such as Secure DNS [16], CMS [13], or OpenPGP [8], to protect the data during transport. How to implement this will depend on the usage scenario, and it is not up to this URI scheme to define how the data referenced by DNS URIs should be protected.

If applications accept unknown dnsqueryelement values in a URI (e.g., URI "dns:www.example.org?secret=value") without knowing what the "secret=value" dnsqueryelement means, a covert channel used to "leak" information may be enabled. The implications of covert channels should be understood by applications that accept unknown dnsqueryelement values.

Slight variations, such as the difference between upper and lower case in the dnsname field, can be used as a covert channel to leak information.

7. IANA Considerations

The IANA has registered the DNS URI scheme, using the template in section 3, in accordance with RFC 2717 [11].

8. Copying Conditions

9. References

9.1. Normative References

[1] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

[2] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[3] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[4] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005.

[5] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.

9.2. Informative References

[6] Postel, J. and J. Reynolds, "File Transfer Protocol", STD 9, RFC 959, October 1985.

[7] Freed, N., Klensin, J., and J. Postel, "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", BCP 13, RFC 2048, November 1996.

[8] Callas, J., Donnerhacke, L., Finney, H., and R. Thayer, "OpenPGP Message Format", RFC 2440, November 1998.

[9] Eastlake 3rd, D. and O. Gudmundsson, "Storing Certificates in the Domain Name System (DNS)", RFC 2538, March 1999.

[10] Crawford, M., "Binary Labels in the Domain Name System", RFC 2673, August 1999.

[11] Petke, R. and I. King, "Registration Procedures for URL Scheme Names", BCP 35, RFC 2717, November 1999.

[12] Connolly, D. and L. Masinter, "The 'text/html' Media Type", RFC 2854, June 2000.

[13] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004.

[14] Faltstrom, P., Hoffman, P., and A. Costello, "Internationalizing Domain Names in Applications (IDNA)", RFC 3490, March 2003.

[15] Josefsson, S., "Domain Name System Media Types", RFC 4027, April 2005.

[16] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.

Author's Address

Simon Josefsson
SJD

EMail: simon@josefsson.org

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).