Attaching Meaning to Solicitation Class Keywords

Status of This Memo

Copyright Notice

Abstract

   This document proposes a mechanism for finding a URI associated with
   a solicitation class keyword, which is defined in RFC 3865, the No
   Soliciting SMTP Service Extension. Solicitation class keywords are
   simple labels consisting of a domain name that has been reversed,
   such as "org.example.adv". These solicitation class keywords are
   inserted in selected header fields or used in the ESMTP service
   extension, including a new "No-Solicit:" header, which can contain
   one or more solicitation class keywords inserted by the sender.

   This document specifies an application based on the Dynamic
   Delegation Discovery System (DDDS) described in RFC 3401 and related
   documents. An algorithm is specified to associate a solicitation
   class keyword with a URI which contains further information about the
   meaning and usage of that solicitation class keyword. For example,
   the registrant of the "example.org" domain could use this mechanism
   to create a URI which contains detailed information about the
   "org.example.adv" solicitation class keyword.

Table of Contents

1. Solicitation Class Keywords

[RFC3865] defines the concept of a "solicitation class keyword",
which is an arbitrary string or label which can be associated with an
electronic mail message and transported by the ESMTP mail service as
defined in [RFC2821] and related documents. Solicitation class
keywords are formatted like domain names, but reversed. For example,
the zone administrator of "example.com" might specify a particular
solicitation class keyword such as "com.example.adv" that could be
inserted in a "No-Solicit:" header by the message sender or in a
trace field by a message transfer agent (MTA). This solicitation
class keyword is inserted by the sender of the message, who may also
insert a variety of other solicitation class keywords as defined by
the sender or by other parties.

[RFC3865] explicitly places discovery of the meaning of a
solicitation class keyword as outside of the scope of the basic ESMTP
service extension. For the purposes of message transport, these
solicitation class keywords are opaque. However, if RFC 3865 becomes
widely used, a mail message might contain a large number of
solicitation class keywords. The "No-Solicit:" header has keywords
inserted by the sender of the message, which might include the
sender's own keywords, as well as those mandated by regulatory
authorities or recommended by voluntary industry associations.
Likewise, the "received:" trace fields might contain a large number
of keywords produced by message transfer agents, filtering software,
forwarding software in the message user agent (MUA), or any other
system in the chain of delivery.

As the number of keywords employed grows, it will be important to
find a method for discovering the meaning behind the various
solicitation class keywords. This document specifies such a
mechanism, associating a solicitation class keyword with a URI which
contains further information by using the DNS NAPTR Resource Record,

which is defined in [RFC3403]. An explicit design goal is to keep
the system as simple as possible. Approaches such as defining an
XML-based structure that would contain specific meta-data about the
solicitation class keyword or other approaches that define the format
of the explanation were ruled out. Instead, the goal is to simply to
associate a solicitation class keyword with a URI, which in turn
contains an explanation of the keyword.

1.1. Terminology

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, [RFC2119].

2. The No-Solicit NAPTR Application

The DDDS framework of [RFC3401] and related documents provides a
powerful set of mechanisms that can yield sophisticated applications
such as ENUM as specified in [RFC3761]. There is a simplification of
the DDDS framework called the Straightforward-NAPTR (S-NAPTR)
application as specified in [RFC3958]. Unfortunately, S-NAPTR does
not permit the use of the "U" flag for terminal lookups and does not
support the regular expression field of the NAPTR RR. Since a
replacement field in a NAPTR record must contain only a domain name,
and our goal is to find a URI, this document does not use the S-NAPTR
mechanism.

This document uses the NAPTR RR to do a single lookup from
solicitation class keyword to URI. The character "." is first
substituted for any instances of the character ":" and then the
solicitation class keyword is reversed, using the character "." as
the delimiter. This becomes the domain name lookup key. For
example, "org.example:ADV" becomes "ADV.example.org".

Note On Domain Names: RFC3865 states that a solicitation class
keyword consists of a valid domain name followed by the ":"
character and by additional valid characters. Several points are
important to remember for implementors. Since domain names are case
insensitive and the ":" character is translated to the "."
character, for purposes of this DDDS application, the following
solicitation class keywords are syntactically equivalent:
"com.example:ADV", "com.Example:adv", and "com:example:ADV".

In addition, it is important to remember that the resulting string
must meet other DNS validity checks. In particular, domain labels
are limited to 63 characters in length and the total length of the
resulting string must be less than 253 characters. Any non-ASCII

characters must be encoded using the Internationalized Domain Names
(IDN) specifications in [RFC3490] and related documents. Note that
non-ASCII characters may be encoded after the ":" character as well.

The fields of the NAPTR RR are used as follows:

o The "ORDER" and "PREFERENCE" fields are to be processed as
   specified in [RFC3403]: if multiple records are returned, the
   one(s) with the lowest "ORDER" value that have a matching
   "SERVICE" field MUST be used. Of those with the lowest ORDER
   value, those with the lowest "PREFERENCE" SHOULD be used.

o The "FLAGS" field MUST contain the character "U".

o The "SERVICES" field MUST contain only the string "no-solicit".

o The "REGEXP" field MUST contain a valid URI as further specified
   in this section.

o The "REPLACEMENT" field MUST be empty.

The "REGEXP" field is defined in [RFC3402] as consisting of a
"delim-character", a POSIX Extended Regular Expression, another
"delim-character", a replacement value, and a final
"delim-character". For this application the following rules apply:

o The "delim-character" MAY be any valid character as defined in
   section 3.2 of [RFC3402].

o The extended regular expression MUST be empty.

o The replacement value MUST contain a valid URI as specified in
   [RFC3986].

o The replacement value SHOULD contain a URI limited to the "ftp",
   "http", and "https" schemes as specified in [RFC3986] and
   [RFC2660].

o The document that is retrieved at the URI SHOULD conform to
   [HTML-4.01], including the Accessibility Guidelines contained
   therein.

3. Example

In this example, a set of NAPTR records are added to the
"example.com" zone and can be retrieved using "dig" or other DNS
utilities:

```
[carl@example.com]% dig 2795.example.com naptr

; <<>> DiG 9.2.3 <<>> 2795.example.com naptr
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY,
   status: NOERROR, id: 43494
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 5,
   AUTHORITY: 2, ADDITIONAL: 1

;; QUESTION SECTION:
;2795.example.com.              IN      NAPTR

;; ANSWER SECTION:
2795.example.com.       86400   IN
    NAPTR   1 1 "U" "iam+invalid"
    "!!http://invalid.example.com/contact.html!" .
2795.example.com.       86400   IN
    NAPTR   1 1 "U" "sip+invalid"
    "!!http://invalid.example.com/contact.html!" .
2795.example.com.       86400   IN
    NAPTR   1 2 "U" "no-solicit"
    "!!http://infinite.example.com/keywordinfo.html!" .
2795.example.com.       86400   IN
    NAPTR   2 1 "U" "no-solicit"
    "!!http://infinite.example.com/keywordinfo.html!" .
2795.example.com.       86400   IN
    NAPTR   1 1 "U" "no-solicit"
    "!!http://infinite.example.com/keywordinfo.html!" .
```

A simple utility written in PERL accepts a lookup key and returns a
URI using the specifications in this document. This example is
non-normative:

```perl
#!/usr/bin/perl

# THIS SAMPLE CODE IS NOT NORMATIVE

# This program accepts a solicitation class keyword and
# returns a URI on success.  It dies quietly on failure.
use strict;

# http://www.net-dns.org/
use Net::DNS;

# reverse the label to create a domain name
$ARGV[0] =~ tr/:/./ ;
my $target = join( ".", reverse( split( /\./, $ARGV[0] ) ) );

# create a resolver
my $res = Net::DNS::Resolver->new;

# find all naptr records
my $query = $res->query( "$target", "NAPTR" ) || exit ;

# Do your DNSSEC checks here, throw away all invalid RRs

# get the answers, strip out non-matching services,
# sort by order, preference
my @rr =
  sort {
    # sort records numerically by order, preference
    $a->order <=> $b->order
      || $a->preference <=> $b->preference
  }
  grep { $_->service =~ /no-solicit/ } $query->answer;

# print the first qualifying record, strip out the
# regexp markers
my $op = substr( my $answer = $rr[0]->regexp , 0, 1 )
    || exit ;
print split ( $op, $answer ) ; exit ;
```

Running the sample code gives the following results:

```
[carl@example.com]% lynx -source `./discover.pl com.example.2795`
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>About Our Solicitation Class Keyword</title>
  </head>
  <body>
    <center>
      <a href="monkey.mp3">
        <img alt="bouncy monkey logo"
             src="images/monkey_fpo.gif" border="0" />
        <br />
      </a>
      <br />
      About com.example.2795:<br />
      It has been determined that the content of this
      mail message<br />
      conforms to the spirit of RFC 2795.
      Congratulations?
    </center>
  </body>
</html>
```

4. DDDS Application Specification

The following definitions apply to this application:

o Application Unique String: The application unique string is a
   Solicitation Class Keyword as defined in [RFC3865].
o First Well Known Rule: The character "." is substituted for the
   character ":" and then the Solicitation Class Keyword is reversed
   in order to produce a valid domain name. For example,
   "com.example:adv" would become "adv.example.com".
o Valid Databases: The DNS _is_ the database.
o Expected Output: A URI.
o The "SERVICE" field MUST contain the string "no-solicit", the
   "FLAGS" field MUST contain the string "U", the "REPLACEMENT" field
   MUST be empty, and the "REGEXP" field MUST be formatted as
   specified in Section 2.

Wildcards are appropriate for this application, allowing multiple
solicitation class keywords that share a common prefix to all point
to the same URI. Note that the NAPTR Resource Record is known as a
"subtyping" RR, which means that additional selectors are available
within the RR to "winnow down" the choices. This means more records
are returned than are actually needed, resulting in more traffic.

But, this also means that wildcards may have unintended effects of
multiple types of NAPTR resource records are used. Implementors and
zone administrators should exercise care in the use of such wildcards
in this application.

5. Acknowledgements

The author would like to thank the following for their helpful
suggestions and reviews of this document: Leslie Daigle, Spencer
Dawkins, Arnt Gulbrandsen, Ted Hardie, Scott Hollenbeck, Russ
Housley, David Kessens, Peter Koch, Michael Mealling, Pekka Savola,
Mark Townsley, and Margaret Wasserman.

6. Security Considerations

This document specifies an application which depends on the Domain
Name System to associate a solicitation class keyword with a URI.
Four security considerations are raised by this application:

1. If the domain name lookup has been compromised, the application
    may return a URI with incorrect guidance on the use of a
    particular solicitation class keyword. In particular, if the
    application returns a URI with the "https:" scheme, and the DNS
    Security Extensions as defined in [RFC4033] and related documents
    are not used, the user would have an unwarranted illusion of
    authenticity making the possibility of active attacks a serious
    concern. Even if both DNS Security Extensions and the "https:"
    scheme are used, the client will need to take additional steps to
    ensure that the two different digital signature validation
    contexts are being administered by the same domain owner.

2. RFC 3865 bases solicitation class keywords on domain names.
    However, it does not define whom a user should trust. A sender
    or an intermediate MTA could insert a solicitation class keyword
    in a message and then use the application defined in this
    document to mislead the message recipient. For example, a
    malicious direct marketer might insert a keyword such as
    "org.example.certified.message" and use a URI to somehow indicate
    that the message (wrongly) has some official status. As with any
    URI, users must take further steps that are outside the scope of
    this specification to determine what and whom to believe.

3. Domain names are not persistent identifiers. As with any
    application that uses domain names, including the World Wide Web,
    if a domain name or a URI is embedded in an electronic mail
    message, there is a possibility that in the future the domain
    name will be controlled by a different zone administrator and
    that

       use of the application described in this document will yield
       different and possibly inconsistent results over time.

   4. A malicious sender could insert a large number of solicitation
       class keywords or improperly formatted solicitation keywords,
       thus performing a Denial of Service attack on the recipient's
       resources through the use of an excessive number of DNS lookups.
       If such a message is sent to many recipients, this can result in
       a Denial of Service attack on the provider at a particular URI
       (e.g., a large number of requests attempting to access a URI such
       as "http://example.net/index.html"). Improperly formatted
       solicitation class keywords, particularly those with a non-
       existent top level or second level domain, could result in a
       Denial of Service attack on DNS registry providers or the DNS
       root servers.

7. IANA Considerations

   There is no central registry maintained by the IANA of values that
   might appear in the "SERVICE" field of a NAPTR resource record.
   Thus, no direct IANA actions are required.

   However, the IANA does maintain an Application Service Tag Registry,
   which is used to support the S-NAPTR DDDS application defined in
   [RFC3958]. The IANA is advised that the "no-solicit" value for the
   SERVICE field is in use per this document and thus should not be used
   in the Application Service Tag Registry for other applications.

8. References

8.1. Normative References

   [HTML-4.01] Raggett, D., Hors, A., and I. Jacobs, "HTML 4.01
               Specification", W3C REC REC-html401-19991224,
               December 1999.

   [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC2660]   Rescorla, E. and A. Schiffman, "The Secure HyperText
               Transfer Protocol", RFC 2660, August 1999.

   [RFC3402]   Mealling, M., "Dynamic Delegation Discovery System (DDDS)
               Part Two: The Algorithm", RFC 3402, October 2002.

   [RFC3403]   Mealling, M., "Dynamic Delegation Discovery System (DDDS)
               Part Three: The Domain Name System (DNS) Database",
               RFC 3403, October 2002.

   [RFC3865]   Malamud, C., "A No Soliciting Simple Mail Transfer
                Protocol (SMTP) Service Extension", RFC 3865,
                September 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.

   [RFC3986]   Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
                Resource Identifier (URI): Generic Syntax", STD 66, RFC
                3986, January 2005.

8.2. Informative References

   [RFC2795]   Christey, S., "The Infinite Monkey Protocol Suite
                (IMPS)", RFC 2795, 1 April 2000.

   [RFC2821]   Klensin, J., "Simple Mail Transfer Protocol", RFC 2821,
                April 2001.

   [RFC3401]   Mealling, M., "Dynamic Delegation Discovery System (DDDS)
                Part One: The Comprehensive DDDS", RFC 3401, October 2002.

   [RFC3490]   Faltstrom, P., Hoffman, P., and A. Costello,
                "Internationalizing Domain Names in Applications (IDNA)",
                RFC 3490, March 2003.

   [RFC3761]   Faltstrom, P. and M. Mealling, "The E.164 to Uniform
                Resource Identifiers (URI) Dynamic Delegation Discovery
                System (DDDS) Application (ENUM)", RFC 3761, April 2004.

   [RFC4033]   Arends, R., Austein, R., Larson, M., Massey, D., and S.
                Rose, "DNS Security Introduction and Requirements",
                RFC 4033, March 2005.

Author's Address

   Carl Malamud
   Memory Palace Press
   PO Box 300
   Sixes, OR 97476
   US

   EMail: carl@media.org