## On Interoperation of Labels Between mDNS and DNS
### draft-ietf-dnssd-mdns-dns-interop-01

Abstract

   Despite its name, DNS-Based Service Discovery can use naming systems
   other than the Domain Name System when looking for services.
   Moreover, when it uses the DNS, DNS-Based Service Discovery uses the
   full capability of DNS, rather than using a subset of available
   octets.  In order for DNS-SD to be used effectively in environments
   where multiple different name systems and conventions for their
   operation are in use, it is important to attend to differences in the
   underlying technology and operational environment.  This memo
   presents an outline of the requirements for selection of labels for
   conventional DNS and other resolution systems when they are expected
   to interoperate in this manner.

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 January 5, 2016.

Table of Contents

## 1.  Introduction

DNS-Based Service Discovery (DNS-SD, [RFC6763]) specifies a mechanism
for discovering services using queries to the Domain Name System
(DNS, [RFC1034], [RFC1035]); and to any other system that uses domain
names, such as Multicast DNS (mDNS, [RFC6762]).  Conventional use of
the DNS generally follows the host name rules [RFC0952] for labels --
the so-called LDH rule.  That convention is the reason behind the
development of Internationalized Domain Names for Applications
(IDNA2008, [RFC5890], [RFC5891], [RFC5892], [RFC5893], [RFC5894],
[RFC5895]).  It is worth noting that the LDH rule is a convention,
and not a rule of the DNS; this is made entirely plain by [RFC2181],
section 11.  Nevertheless, there is a widespread belief that in many
circumstances domain names cannot be used in the DNS unless they
cleave to the LDH rule.

At the same time, mDNS requires that labels be encoded in UTF-8, and permits a range of characters in labels that are not permitted by IDNA2008 or the LDH rule.  For example, mDNS encourages the use of spaces and punctuation in mDNS names (see [RFC6763], section 4.1.3). It does not restrict which Unicode code points may be used in those labels, so long as the code points are UTF-8 in Net-Unicode [RFC5198] format.

Users of applications are, of course, frequently unconcerned with (not to say oblivious to) the name-resolution system(s) in service at any given moment, and are inclined simply to use the same domain names in different contexts.  As a result, the same domain name might be tried using different name resolution technologies.  If DNS-SD is to be used in an environment where multiple resolution systems (such as mDNS and DNS) are to be queried for services, then some parts of the domain names to be queried will need to be compatible with the rules and conventions for all the relevant technologies.

One approach to interoperability under these circumstances is to use a single operational convention (a "profile") for domain names under the different naming systems.  This memo assumes such a use profile, and attempts to outline what is necessary to make it work without specifying any particular technology.  It does assume, however, that the global DNS is eventually likely to be implicated.  Given the general tendency of all resolution eventually to fall through to the DNS, that assumption does not seem controversial.

It is worth noting that users of DNS-SD do not use the service discovery names in the same way that users of other domain names might.  Domain names often might as easily be entered as direct user input as by any other method.  But the service discovery context generally assumes users are picking a service from a list.  As a result, the sorts of application considerations that are appropriate to the general-purpose DNS name, and that resulted in the A-label/ U-label split (see below) in IDNA2008, are not entirely the right approach for DNS-SD.

## 1.1.  Conventions and terms used in this document

Wherever appropriate, this memo uses the terminology defined in Section 2 of [RFC5890].  In particular, the reader is assumed to be familiar with the terms "U-label", "LDH label", and "A-label" from that document.  Similarly, the reader is assumed to be familiar with the U+NNNN notation for Unicode code points used in [RFC5890] and other documents dealing with Unicode code points.  In the interests of brevity and consistency, the definitions are not repeated here.

Sometimes this memo refers to names in the DNS as though the LDH rule
and IDNA2008 are strict requirements.  They are not.  DNS labels are,
in principle, just collections of octets, and therefore in principle
the LDH rule is not a constraint.  In practice, applications
sometimes intercept labels that do not conform to the LDH rule and
apply IDNA and other transformations.

The DNS, perhaps unfortunately, has produced its own jargon.
Unfamiliar DNS-related terms in this memo should be found in
[I-D.ietf-dnsop-dns-terminology].

The term "owner name" (common to the DNS vernacular; see above) is
used here to apply not just to the domain names to be looked up in
the DNS, but to any name that might be looked up either in the DNS or
using other technologies.  It therefore includes names that might not
actually exist anywhere.  In addition, what follows depends on the
idea that not every domain name may be looked up in the DNS.  For
instance, names ending in "local." (in the presentation format) are
not ordinarily looked up in the DNS, but instead by querying mDNS.

DNS-SD specifies three portions of the owner name for a DNS-SD
resource record.  These are the <Instance> portion, the <Service>
portion, and the <Domain>.  The owner name made of these three parts
is called the Service Instance Name.  It is worth observing that a
portion may be more than one label long.  See [RFC6763], section 4.1.
Further discussion of the parts is found in Section 4.

Throughout this memo, mDNS is used liberally as the alternative
resolution mechanism to DNS.  This is for convenience rather than
rigour: any alternative name resolution to DNS could present the same
friction with the prevailing operational conventions of the global
DNS.  It so happens that mDNS is the overwhelmingly successful
alternative as of this writing, so it is used in order to make the
issues plainer to the reader.  Other alternative resolution
mechanisms may in general be read wherever mDNS appears in the text,
except where details of the mDNS specification appear.

## 2.  Why there could be a problem at all

One might reasonably wonder why there is a problem to be solved at
all.  After all, DNS labels permit any octet whatsoever, and anything
that can be useful with DNS-SD cannot use any names that are outside
the protocol strictures of the DNS.

The reason for the trouble is twofold.  First, and least troublesome,
is the possibility of resolvers that are attempting to offer IDNA
service system-wide.  Given the design of IDNA2008, it is reasonable
to suppose that on some systems high-level name resolution libraries

will perform the U-label/A-label transformation automatically, saving
applications from these details.  If this were the main problem,
however, it would presumably be self-correcting; for the right answer
would be, "Don't use those libraries for DNS-SD," and DNS-SD would
not work reliably in cases where such libraries were in use.  This
would be unfortunate; but given that DNS-SD in Internet contexts is
as of this writing not in ubiquitous use, it should not represent a
fatal issue.

The greater problem is that the "infrastructure" types of DNS service
-- the root zone, the top-level domains, and so on -- have embraced
IDNA and refuse registration of raw UTF-8 into their zones.  As of
this writing there is (perhaps unfortunately) no reliable way to
discover where these sorts of DNS services end.  Nevertheless, some
client programs (notably web browsers) have adopted a number of
different policies about how domain names will be looked up and
presented to users given the policies of the relevant DNS zone
operators.  None of these policies permit raw UTF-8.  Since it is
anticipated that DNS-SD when used with the DNS will be inside domain
names beneath those kinds of "infrastructure" domains, the
implications of IDNA2008 must be a consideration.

For further exploration of issues relating to encoding of domain
names generally, the reader should consult [RFC6055].

## 3.  Requirements for a profile for label interoperation

Any interoperability between DNS (including prevailing operational
conventions and other resolution technologies will require
interoperability across the portions of a DNS-SD Service Instance
Name that are implicated in regular DNS lookups.  Only some portions
are implicated.  In any case, if a given portion is implicated, the
profile will need to apply to all labels in that portion.

In addition, because DNS-SD Service Instance Names can be used in a
domain name slot, care must be taken by DNS-SD-aware resolvers to
handle the different portions as outlined here, so that DNS-SD
portions that do not use IDNA2008 will not be treated as U-labels and
will not accidentally undergo IDNA processing.

Because the profile will need to apply to names that might need to
interoperate with names in the public DNS, and because other
resolution mechanisms (such as mDNS) could permit labels that IDNA
does not, the profile might reduce the labels that could be used with
those other resolution mechanisms.  One consequence of this is that
some recommendations from [RFC6763] will not really be possible to
implement using names subject to the profile.  In particular,
[RFC6763], section 4.1.3 recommends that labels always be stored and

communicated as UTF-8, even in the DNS.  Because of the way the
public DNS is currently operated (see Section 2), the advice to store
and transmit labels as UTF-8 in the DNS is likely either to encounter
problems or result in unnecessary traffic to the public DNS (or
both).  In particular, the <Domain> part of a Service Instance Name
is unlikely to be found in its UTF-8 form in the public DNS tree for
zones that are using IDNA2008.  By contrast, for example, mDNS
normally uses UTF-8.

U-labels cannot contain upper case letters.  That restriction extends
to ASCII-range upper case letters that work fine in LDH-labels.  It
may be confusing that the character "A" works in the DNS when none of
the characters in the label has a diacritic, but does not work when
there is such a diacritic in the label.  Labels in mDNS names (or
other resolution technologies) may contain upper case characters, so
the profile will need either to restrict the use of upper case or
come up with a reliable and predictable (to users) convention for
case folding even in the presence of diacritics.

## 4.  DNS-SD portions

Service Instance Names are made up of three portions.

### 4.1.  The <Instance> Portion of the Service Instance Name

[RFC6763] is clear that the <Instance> portion of the Service
Instance Name is intended for presentation to users, and therefore
virtually any character is permitted in it.  There are two ways that
a profile might address this portion.

The first way would be to treat this portion as likely to be
intercepted by system-wide IDNA-aware resolvers, or likely subject to
strict IDNA conformance requirements for publication in the relevant
zone.  In this case, the portion would need to be made subject to the
profile, thereby curtailing what characters may appear in this
portion.  This approach permits DNS-SD to use any standard system
resolver but presents inconsistencies with the DNS-SD specification
and with DNS-SD that is exclusively mDNS-based.  Therefore, this
strategy is rejected.

Instead, DNS-SD implementations can intercept the <Instance> portion
of a Service Instance Name and ensure that those labels are never
handed to IDNA-aware resolvers that might attempt to convert these
labels into A-labels.  Under this approach, the DNS-SD <Instance>
portion works as it always does, but at the cost of using special
resolution code built into the DNS-SD system.  A practical
consequence of this is that zone operators need to be prepared not to
apply the LDH rule to all labels, and may need to make special

concessions to ensure that the <Instance> portion can contain spaces,
upper and lower case, and any UTF-8 code point; or else to prepare a
user interface to handle the exceptions that would otherwise be
generated.  Automatic conversion to A-labels is not acceptable.

## 4.2.  The <Service> Portion of the Service Instance Name

DNS-SD includes a <Service> component in the Service Instance Name.
This component is not really user-facing data, but is instead control
data embedded in the Service Instance Name.  This component includes
so-called "underscore labels", which are labels prepended with U+005F
(_).  The underscore label convention was established by DNS SRV
([RFC2782]) for identifying metadata inside DNS names.  A system-wide
resolver (or DNS middlebox) that cannot handle underscore labels will
not work with DNS-SD at all, so it is safe to suppose that such
resolvers will not attempt to do special processing on these labels.
Therefore, the <Service> portion of the Service Instance Name will
not be subject to the profile.  By the same token, it should be noted
that underscore labels are never subject to IDNA processing (they're
formally incompatible), and therefore concerns about IDNA are
irrelevant for these labels.

## 4.3.  The <Domain> Portion of the Service Instance Name

The <Domain> portion of the Service Instance Name forms an integral
part of the owner name submitted for DNS resolution.  A system-wide
resolver that is IDNA2008-aware is likely to interpret labels with
UTF-8 in the owner name as candidates for IDNA2008 processing.  More
important, operators of internationalized domain names will
frequently publish such names in the DNS as A-labels; certainly, the
top-most labels will always be A-labels.  Therefore, these labels
will need to be subject to the profile.  DNS-SD implementations ought
to identify the <Domain> portion of the Service Instance Name and
treat it subject to IDNA2008 in case the domain is to be queried from
the global DNS.  In the event that the <Domain> portion of the
Service Instance Name fails to resolve, it is acceptable to
substitute labels with plain UTF-8, starting at the lowest label in
the DNS tree and working toward the root.  This approach differs from
the rule for resolution published in [RFC6763], because it privileges
IDNA2008-compatible labels over UTF-8 labels.

One might argue against this restriction on either of two grounds:

1.  It is possible the names may be in the DNS in UTF-8, and RFC 6763
    already specifies a fallback strategy of progressively attempting
    first the UTF-8 label lookup (it might not be a U-label) and then
    if possible the A-label lookup.

2.  Zone administrators that wish to support DNS-SD can publish a
    UTF-8 version of the zone along side the A-label version of the
    zone.

The first of these is rejected because it represents a potentially
significant increase in DNS lookup traffic for no value.  It is
possible for a DNS-SD application to identify the <Domain> portion of
the Service Instance Name.  The standard way to publish IDNs on the
Internet uses IDNA.  Therefore, additional lookups should not be
encouraged.  When [RFC6763] was published, the bulk of IDNs were
lower in the tree.  Now that there are internationalized labels in
the root zone, it is desirable to minimize queries to the Internet
infrastructure if they are sure to be answered in the negative.

The second reason depends on the idea that it is possible to maintain
two names in sync with one another.  This is not strictly speaking
true, although in this case the domain operator could simply create a
DNAME record [RFC6672] from the UTF-8 name to the IDNA2008 zone.
This still, however, relies on being able to reach the (UTF-8) name
in question, and it is unlikely that the UTF-8 version of the zone
will be delegated from anywhere.  Moreover, in many organizations the
support for DNS-SD and the support for domain name delegations are
not performed by the same department, and depending on a co-
ordination between the two will make the system more fragile, or
slower, or both.

## 5.  Acknowledgements

The author gratefully acknowledges the insights of Stuart Cheshire,
Kerry Lynn, and Dave Thaler.  Kerry Lynn deserves special gratitude
for his energy and persistence in pressing unanswered questions.

## 6.  IANA Considerations

This memo makes no requests of IANA.

## 7.  Security Considerations

This memo presents some requirements for future development, but does
not specify anything.  Therefore, it has no implications for
security.

## 8.  Informative References

[I-D.ietf-dnsop-dns-terminology]
          Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS
          Terminology", draft-ietf-dnsop-dns-terminology-03 (work in
          progress), June 2015.

   [RFC0952]  Harrenstien, K., Stahl, M., and E. Feinler, "DoD Internet
              host table specification", RFC 952, October 1985.

   [RFC1034]  Mockapetris, P., "Domain names - concepts and facilities",
              STD 13, RFC 1034, November 1987.

   [RFC1035]  Mockapetris, P., "Domain names - implementation and
              specification", STD 13, RFC 1035, November 1987.

   [RFC2181]  Elz, R. and R. Bush, "Clarifications to the DNS
              Specification", RFC 2181, July 1997.

   [RFC2782]  Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for
              specifying the location of services (DNS SRV)", RFC 2782,
              February 2000.

   [RFC5198]  Klensin, J. and M. Padlipsky, "Unicode Format for Network
              Interchange", RFC 5198, March 2008.

   [RFC5890]  Klensin, J., "Internationalized Domain Names for
              Applications (IDNA): Definitions and Document Framework",
              RFC 5890, August 2010.

   [RFC5891]  Klensin, J., "Internationalized Domain Names in
              Applications (IDNA): Protocol", RFC 5891, August 2010.

   [RFC5892]  Faltstrom, P., "The Unicode Code Points and
              Internationalized Domain Names for Applications (IDNA)",
              RFC 5892, August 2010.

   [RFC5893]  Alvestrand, H. and C. Karp, "Right-to-Left Scripts for
              Internationalized Domain Names for Applications (IDNA)",
              RFC 5893, August 2010.

   [RFC5894]  Klensin, J., "Internationalized Domain Names for
              Applications (IDNA): Background, Explanation, and
              Rationale", RFC 5894, August 2010.

   [RFC5895]  Resnick, P. and P. Hoffman, "Mapping Characters for
              Internationalized Domain Names in Applications (IDNA)
              2008", RFC 5895, September 2010.

   [RFC6055]  Thaler, D., Klensin, J., and S. Cheshire, "IAB Thoughts on
              Encodings for Internationalized Domain Names", RFC 6055,
              February 2011.

   [RFC6672]  Rose, S. and W. Wijngaards, "DNAME Redirection in the
              DNS", RFC 6672, June 2012.

   [RFC6762]  Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762,
             February 2013.

   [RFC6763]  Cheshire, S. and M. Krochmal, "DNS-Based Service
             Discovery", RFC 6763, February 2013.

Appendix A.  Change History

A.1.  draft-ietf-dnssd-mdns-dns-interop-01

   Alter text to make clear that the main issue is the way the public
   DNS is currently administered, not system resolvers.  I suppose this
   should have been clear before, but I didn't do that.  Many thanks to
   Kerry Lynn for penetrating questions that illuminated what I'd left
   out.

A.2.  draft-ietf-dnssd-mdns-dns-interop-00

   1st WG version

   Add text to make clear that fallback from A-label lookup to UTF-8
   label lookup ok, per WG comments at IETF 91

A.3.  draft-sullivan-dnssd-mdns-dns-interop-01

   o  Decided which portions would be affected

   o  Explained the difference in user interfaces between DNS-SD and
      usual DNS operation

   o  Provided background on why the Domain portion should be treated
      differently

A.4.  draft-sullivan-dnssd-mdns-dns-interop-00

   Initial version.

Author's Address

   Andrew Sullivan
   Dyn
   150 Dow St.
   Manchester, NH  03101
   U.S.A.

   Email: asullivan@dyn.com