

Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: January 2, 2015

A. Aggarwal  
Qualcomm (QCE)  
July 01, 2014

Optimizing DNS-SD query using TXT records  
draft-aggarwal-dnssd-optimize-query-00

Abstract

DNS-SD allows a client to find a list of named instances of a service name over a particular transport within a domain of interest using standard DNS queries. As the number of potential responders increases, DNS-SD based discovery doesn't scale well. To mitigate the scaling issues, schemes to narrow down the search context would be needed. The document proposes to include key/value pairs in the form of a DNS TXT record along with the service name in the DNS query to assist with the discovery process. The DNS TXT record can be placed in the additional section of the query without requiring any changes to the structure of DNS messages.

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 2, 2015.

Copyright Notice

Copyright (c) 2014 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

1. Introduction . . . . .	2
2. Background . . . . .	3
3. Proposed Changes . . . . .	4
4. Realization of the proposal . . . . .	4
5. Deployment Considerations . . . . .	5
6. API Considerations . . . . .	5
7. Security Considerations . . . . .	5
8. IANA Considerations . . . . .	6
9. Acknowledgements . . . . .	6
10. Normative References . . . . .	6
Author's Address . . . . .	6

## 1. Introduction

DNS-SD [RFC6763] in combination with mDNS [RFC6762] provide a discovery framework for service names registered with IANA over a local link. The objective of DNS-SD was to discover service instances that implement a given service. The use of mDNS scales well when the number of service instances that implement a given service are limited in number on the local link. However, when the number of wireless devices (e.g., Wi-Fi) approach hundreds of devices in a typical link, several service instances may respond when a DNS-SD query is issued for a given service name. The number of wireless devices is slated to grow further as more devices (things) are deployed as part of the Internet of Things (IoT) era.

At the same time, the DNS-SD protocol also enables discovery in various operating environments that rely on unicast DNS. Being able to narrow down the search context beyond the service name scope will be even more critical for such DNS-SD based discovery schemes to scale.

This contribution proposes one such solution.

This document proposes no change to the structure of DNS messages, and no new operation codes, response codes, resource record types, or any other new DNS protocol values.

### 1.1. Sample Use Cases

Some sample use cases that might experience scaling problems are mentioned below:

- o A client application is looking to find color printers on the local network
- o A lighting application needs to discover lighting fixtures or bulbs from a given manufacturer before establishing a session with each device to control the fixtures

## 2. Background

There are two potential mechanisms that can help a DNS-SD querier narrow down the answers of interest within the scope of DNS-SD [RFC6763]:

- o Placing TXT records in the response: DNS has an efficiency feature whereby a DNS server may place additional records in the additional section of the DNS message. These additional records are records that the client did not explicitly request, but the server has reasonable grounds to expect that the client might request them shortly, so including them can save the client from having to issue additional queries. DNS-SD clarifies that the intention of DNS-SD TXT records is to convey a small amount of useful additional information about a service. Ideally, it should not be necessary for a client to retrieve this additional information before it can usefully establish a connection to the service. For a well-designed application protocol, even if there is no information at all in the TXT record, it should be possible, knowing only the host name, port number, and protocol being used, to communicate with that listening process and then perform version- or feature-negotiation to determine any further options or capabilities of the service instance.
- o Using subtype as part of the question: DNS-SD allows a querier to send a subtype along with the service name. It does require that the subtype be 63 octets or fewer. DNS-SD RFC further clarifies that these should be documented in the protocol specification in question and/or in the "notes" field of the registration request sent to IANA.

It can be argued that mechanisms in place to narrow down the search beyond the service name are not very flexible. While nothing prevents an application implementing DNS-SD to eventually find the service instance of interest, it results in unnecessary traffic and delay. The proposal is to enable a richer search query mechanism by

explicitly adding key/value pairs in the query to avoid having to establish sessions with all services that match the service name in the question. Since DNS-SD allows a responder to include TXT records in the additional section with key-value pairs that it thinks the client may request, session establishment with the responder can be avoided if the desired key/value pairs (from the client's perspective) were included in the response.

### 3. Proposed Changes

DNS-SD as defined in [RFC6763] uses DNS TXT records to store arbitrary key/value pairs conveying additional information about the named service. Each key/value pair is encoded as its own constituent string within the DNS TXT record, in the form "key=value" (without the quotation marks). The proposal is for the client to be able to query for key/value pairs along with the service name. The DNS TXT record in the additional section of the query serves to send this additional information. Since DNS messages are allowed to have an additional section, this proposal doesn't require any changes to the structure of DNS messages.

DNS TXT record is allowed to have multiple key/value pairs. If multiple keys are present in a given TXT record, they are AND'ed and the responder must match all the keys in the TXT record. At the same time, DNS query could include more than one TXT record analogous to multiple TXT records in the response. If multiple TXT records are present in the query, they are logically OR'ed while the keys of each TXT record are AND'ed as stated above.

### 4. Realization of the proposal

Actual key/value pairs that can be sent are specified within the application protocol specification. Some examples to aid in the understanding of the proposal are mentioned below. They correspond to the use cases introduced earlier e.g.

- o A client application looking for color printer can add color=true in the DNS TXT record as part of the additional section of the query
- o A lighting application looking to discover bulbs by a certain manufacturer (such as Philips), can add the DNS TXT record in the additional section of the query with manuf=Philips
- o The discovery scope can be further constrained by defining additional keys within the service protocol specification. By augmenting the query with additional context, the spurious traffic

and additional delay in finding the service instance of interest is reduced.

## 5. Deployment Considerations

An important deployment consideration is to analyze the behavior of an existing mDNS responder and unicast DNS to the receipt of DNS-SD query with a service name in the question section and TXT records in the additional section. If mDNS responder doesn't recognize TXT records, no filtering would occur and a response will be sent only if there is a match for the service name.

Regarding the behavior of unicast DNS when the standard query carries TXT records in the additional section, the DNS will respond strictly based on the service name in the question without any filtering based on the TXT records. DNS will issue a negative response unless there is a record matching the question. In summary, unicast DNS will continue to serve DNS queries that include TXT records in the additional section.

## 6. API Considerations

Several high level operating systems (Android, iOS) provide service discovery APIs. For the proposed enhancement to be realized, service registration should allow for service specific key/value pairs to be registered. This capability should already exist since it is allowed as per the current DNS-SD specification. The additional impact would be for the client application to be able to query for specific key/value pairs along with the service name over a specific transport.

## 7. Security Considerations

The additional data (key/value pairs signifying the search context) beyond the service name in the DNS-SD query inherently reveals more information about what the client is searching for. DNS and mDNS today do not provide confidentiality, so observers already have access to potentially sensitive information such as what names one is requesting; addressing this issue is outside the scope of this extension. Even if confidentiality were to be solved, this extension still provides more information to the actual DNS/mDNS responders themselves. A client concerned about such information disclosure can simply choose not to use this extension for such queries, and thus trade off efficiency for privacy.

## 8. IANA Considerations

This memo includes no request to IANA.

## 9. Acknowledgements

Thanks to Dave Thaler for helping develop this idea and formalizing as a contribution for DNS-SD enhancements.

## 10. Normative References

- [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.
- [RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, December 2012.
- [RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, December 2012.

## Author's Address

Ashutosh Aggarwal  
Qualcomm (QCE)  
5775 Morehouse Dr  
San Diego , California 92121  
USA  
  
Phone: +1 858 658 2229  
Email: [aggarwal@qce.qualcomm.com](mailto:aggarwal@qce.qualcomm.com)

Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: July 26, 2014

S. Cheshire  
Apple Inc.  
January 22, 2014

Hybrid Unicast/Multicast DNS-Based Service Discovery  
draft-cheshire-dnssd-hybrid-01

Abstract

Performing DNS-Based Service Discovery using purely link-local Multicast DNS enables discovery of services that are on the local link, but not (without some kind of proxy or similar special support) of services that are outside the local link. Using a very large local link with thousands of hosts improves service discovery, but at the cost of large amounts of multicast traffic.

Performing DNS-Based Service Discovery using purely Unicast DNS is more efficient, but requires configuration of DNS Update keys on the devices offering the services, which can be onerous for simple devices like printers and network cameras.

Hence a compromise is needed, that provides easy service discovery without requiring either large amounts of multicast traffic or onerous configuration.

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 July 26, 2014.

Copyright Notice

Copyright (c) 2014 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

1. Introduction . . . . .	3
2. Conventions and Terminology Used in this Document . . . . .	3
3. Hybrid Proxy Operation . . . . .	4
4. Implementation Status . . . . .	9
5. IPv6 Considerations . . . . .	11
6. Security Considerations . . . . .	11
7. Intellectual Property Rights . . . . .	11
8. IANA Considerations . . . . .	11
9. Acknowledgments . . . . .	11
10. References . . . . .	12
10.1. Normative References . . . . .	12
10.2. Informative References . . . . .	12
Author's Address . . . . .	13



## 1. Introduction

Multicast DNS [RFC6762] and its companion technology DNS-based Service Discovery [RFC6763] were created to provide IP networking with the ease-of-use and autoconfiguration for which AppleTalk was well known [RFC6760] [ZC].

Section 10 ("Populating the DNS with Information") of the DNS-SD specification [RFC6763] discusses possible ways that a service's PTR, SRV, TXT and address records can make their way into the DNS namespace, including manual zone file configuration [RFC1034] [RFC1035], DNS Update [RFC2136] [RFC3007] and proxies.

This document specifies a type of proxy called a Hybrid Proxy that uses Multicast DNS [RFC6762] to discover Multicast DNS records on its local link, and makes corresponding DNS records visible in the Unicast DNS namespace.

## 2. Conventions and Terminology Used in this Document

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 "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

Multicast DNS works between a hosts on the same link. A set of hosts is considered to be "on the same link", if:

- o when any host A from that set sends a packet to any other host B in that set, using unicast, multicast, or broadcast, the entire link-layer packet payload arrives unmodified, and
- o a broadcast sent over that link by any host from that set of hosts can be received by every other host in that set

The link-layer *\*header\** may be modified, such as in Token Ring Source Routing [802.5], but not the link-layer *\*payload\**. In particular, if any device forwarding a packet modifies any part of the IP header or IP payload then the packet is no longer considered to be on the same link. This means that the packet may pass through devices such as repeaters, bridges, hubs or switches and still be considered to be on the same link for the purpose of this document, but not through a device such as an IP router that decrements the TTL or otherwise modifies the IP header.

### 3. Hybrid Proxy Operation

In its simplest form, each local link in an organization is assigned a unique Unicast DNS domain name, such as "Building 1.example.com." or "4th Floor.Building 1.example.com." (Grouping multiple local links under the same Unicast DNS domain name is to be specified in a future companion document, but for the purposes of this document, assume that each link has its own unique Unicast DNS domain name.)

Each link in an organization has a Hybrid Proxy which serves it. This function could be performed by a router on that link, or, with appropriate VLAN configuration, a single Hybrid Proxy could have a logical presence on, and serve as the Hybrid Proxy for, multiple links. In the organization's DNS server, NS records are used to delegate ownership of each defined link name (e.g., "Building 1.example.com.") to the Hybrid Proxy which serves that link.

Domain Enumeration PTR records [RFC6763] are also created to inform clients of available Device Discovery domains, e.g.,:

b._dns-sd._udp.example.com.	PTR	Building 1.example.com.
	PTR	Building 2.example.com.
	PTR	Building 3.example.com.
	PTR	Building 4.example.com.
lb._dns-sd._udp.example.com.	PTR	Building 1.example.com.

When a DNS-SD client issues a Unicast DNS query to discover services in a particular Unicast DNS (e.g., "\_printer.\_tcp.Building 1.example.com. PTR ?") the normal DNS delegation mechanism results in that query being served from the delegated authoritative name server for that subdomain, namely the Hybrid Proxy on the link in question. Like a conventional Unicast DNS server, a Hybrid Proxy implements the usual Unicast DNS protocol [RFC1034] [RFC1035] over UDP and TCP. However, unlike a conventional Unicast DNS server that generates answers from the data in its manually-configured zone file, a Hybrid Proxy generates answers by performing a Multicast DNS query (e.g., "\_printer.\_tcp.local. PTR ?") on its local link, and then, from the data in the Multicast DNS replies it receives, generating the corresponding Unicast DNS reply.

### 3.1. Data Translation

Generating the corresponding Unicast DNS reply involves, at the very least, rewriting the "local" suffix to the appropriate Unicast DNS domain (e.g., "Building 1.example.com").

In addition it would be desirable to suppress Unicast DNS replies for records that are not useful outside the local link. For example, DNS A and AAAA records for IPv4 link-local addresses [RFC3927] and IPv6 link-local addresses [RFC4862] should be suppressed. Similarly, for sites that have multiple private address realms [RFC1918], private addresses from one private address realm should not be communicated to clients in a different private address realm.

By the same logic, DNS SRV records that reference target host names that have no addresses usable by the requester should be suppressed, and likewise, DNS PTR records that point to DNS names with DNS SRV records that reference target host names that have no addresses usable by the requester should be also be suppressed.

The same reachability requirement for advertised services also applies to the Hybrid Proxy itself. The mechanism specified in this document only works if the Hybrid Proxy is reachable from the client making the request.

### 3.1.1.1. Application-Specific Data Translation

There may be cases where Application-Specific Data Translation is appropriate.

For example, AirPrint printers tend to advertise fairly verbose information about their capabilities in their DNS-SD TXT record. This information is a legacy from LPR printing, because LPR does not have in-band capability negotiation, so all of this information is put in the DNS-SD TXT record instead. IPP printing does have in-band capability negotiation, but for convenience printers tend to include the same capability information in their IPP DNS-SD TXT records as well. For local mDNS use this extra TXT record information is inefficient, but not fatal. However, when a Hybrid Proxy aggregates data from multiple printers on a link, and sends it via unicast (via UDP or TCP) this amount of unnecessary TXT record information can result in large replies. Therefore, a Hybrid Proxy that is aware of the specifics of an application-layer protocol such as Apple's AirPrint (which uses IPP) can elide unnecessary key/value pairs from the DNS-SD TXT record for better network efficiency.

Note that this kind of Application-Specific Data Translation is expected to be very rare. It is the exception, rather than the rule. This is an example of a common theme in computing. It is frequently the case that it is wise to start with a clean, layered design, with clear boundaries. Then, in certain special cases, those layer boundaries may be violated, where the performance and efficiency benefits outweigh the inelegance of the layer violation.

As in other similar situations, these layer violations optional. They are done only for efficiency reasons, and are not required for correct operation. A Hybrid Proxy can operate solely at the mDNS layer, without any knowledge of DNS-SD semantics, or of any DNS-SD client semantics.

### 3.2. Answer Aggregation

In a simple analysis, simply gathering multicast answers and forwarding them in a unicast reply seems adequate, but it raises the question of how long the Hybrid Proxy should wait to be sure that it has received all the Multicast DNS replies it needs to form a complete Unicast DNS reply. If it waits too little time, then it risks its Unicast DNS reply being incomplete. If it waits too long, then it creates a poor user experience at the client end.

This dilemma is solved by use of DNS Long-Lived Queries (DNS LLQ) [I-D.sekar-dns-llq]. The Hybrid Proxy replies immediately to the Unicast DNS query using the Multicast DNS records it already has in its cache (if any). This provides a good client user experience by providing a near-instantaneous response. Simultaneously, the Hybrid Proxy issues a Multicast DNS query on the local link to discover if there are any additional Multicast DNS records it did not already know about. Should additional Multicast DNS replies be received, these are then delivered to the client using DNS LLQ update messages. The timeliness of such LLQ updates is limited only by the timeliness of the device responding to the Multicast DNS query. If the Multicast DNS device responds quickly, then the LLQ update is delivered quickly. If the Multicast DNS device responds slowly, then the LLQ update is delivered slowly. The benefit of using LLQ is that the Hybrid Proxy can respond promptly because it doesn't have to delay its unicast reply to allow for the expected worst-case delay for receiving all the Multicast DNS replies. Even if a proxy were to try to provide reliability by assuming an excessively pessimistic worst-case time (thereby giving a very poor user experience) there would still be the risk of a slow Multicast DNS device taking even longer than that (e.g, a device that is not even powered on until ten seconds after the initial query is received) resulting in incomplete replies. Using LLQs solves this dilemma: even very late replies are not lost; they are delivered in subsequent LLQ update messages.

There are two factors that determine specifically how replies are generated. The first factor is whether the Hybrid Proxy already has at least one record in its cache that positively answers the question. The second factor is whether the query from the client includes the LLQ option (typical with long-lived service browsing PTR queries) or not (typical with one-shot operations like SRV or address record queries).

- o No answer in cache; no LLQ option: Do local mDNS query three times, and then return NXDOMAIN if no answer after three tries.
- o No answer in cache; with LLQ option: As above, do local mDNS query three times, and then return NXDOMAIN if no answer after three tries. However, the query remains active for as long as the client maintains the LLQ state, and if mDNS answers are received later, LLQ update messages are sent. (Reasoning: We don't need to rush to send an empty answer.)
- o At least one answer in cache; no LLQ option: Send reply right away to minimise delay. No local mDNS queries are performed. (Reasoning: Given RRSets TTL harmonisation, if the proxy has one answer in its cache, it should have all of them.)
- o At least one answer in cache; with LLQ option: As above, send reply right away to minimise delay. However, the query remains active for as long as the client maintains the LLQ state, and if additional mDNS answers are received later, LLQ update messages are sent. (Reasoning: We want UI that is displayed very rapidly, yet continues to remain accurate even as the network environment changes.)

#### 4. Implementation Status

Some aspects of the mechanism specified in this document already exist in deployed software. Some aspects are new. This section outlines which aspects already exist and which are new.

##### 4.1. Already Implemented and Deployed

Domain enumeration discovery by the client (the "b.\_dns-sd.\_udp" queries) is already implemented and deployed.

Unicast queries to the indicated discovery domain is already implemented and deployed.

These are implemented and deployed in Mac OS X 10.4 and later (including all versions of Apple iOS, on all iPhone and iPads), in Bonjour for Windows, and in Android 4.1 "Jelly Bean" (API Level 16) and later.

Domain enumeration discovery and unicast querying have been used for several years at IETF meetings to make Terminal Room printers discoverable from outside the Terminal room. When you Press Cmd-P on your Mac, or select AirPrint on your iPad or iPhone, and the Terminal room printers appear, that is because your client is doing unicast DNS queries to the IETF DNS servers.

##### 4.2. Partially Implemented

The current APIs make multiple domains visible to client software, but most client UI today lumps all discovered services into a single flat list. This is largely a chicken-and-egg problem. Application writers were naturally reluctant to spend time writing domain-aware UI code when few customers today would benefit from it. If Hybrid Proxy deployment becomes common, then application writers will have a reason to provide better UI. Existing applications will work with the Hybrid Proxy, but will show all services in a single flat list. Applications with improved UI will group services by domain.

The Long-Lived Query mechanism [I-D.sekar-dns-llq] referred to in this specification exists and is deployed, but has not been standardized by the IETF. It is possible that the IETF may choose to standardize a different or better Long-Lived Query mechanism. In that case, the pragmatic deployment approach would be for vendors to produce Hybrid Proxies that implement both the deployed Long-Lived Query mechanism [I-D.sekar-dns-llq] (for today's clients) and a new IETF Standard Long-Lived Query mechanism (as the future long-term direction).

#### 4.3. Not Yet Implemented

The translating/filtering Hybrid Proxy specified in this document. Once implemented, such a Hybrid Proxy will immediately make wide-area discovery available with today's existing clients and devices.

A mechanism to 'stitch' together multiple ".local." zones so that they appear as one. Such a mechanism will be specified in a future companion document.



## 5. IPv6 Considerations

An IPv4-only host and an IPv6-only host behave as "ships that pass in the night". Even if they are on the same Ethernet, neither is aware of the other's traffic. For this reason, each physical link may have *\*two\** unrelated ".local." zones, one for IPv4 and one for IPv6. Since for practical purposes, a group of IPv4-only hosts and a group of IPv6-only hosts on the same Ethernet act as if they were on two entirely separate Ethernet segments, it is unsurprising that their use of the ".local." zone should occur exactly as it would if they really were on two entirely separate Ethernet segments.

It will be desirable to have a mechanism to 'stitch' together these two unrelated ".local." zones so that they appear as one. Such mechanism will need to be able to differentiate between a dual-stack (v4/v6) host participating in both ".local." zones, and two different hosts, one IPv4-only and the other IPv6-only, which are both trying to use the same name(s). Such a mechanism will be specified in a future companion document.

## 6. Security Considerations

A service proves its presence on a local link by its ability to answer link-local multicast queries on that link. If greater security is desired, then the Hybrid Proxy mechanism should not be used, and something with stronger security should be used instead, such as authenticated secure DNS Update [RFC2136] [RFC3007].

## 7. Intellectual Property Rights

Apple has submitted an IPR disclosure concerning the technique proposed in this document. Details are available on the IETF IPR disclosure page [IPR2119].

## 8. IANA Considerations

This document has no IANA Considerations.

## 9. Acknowledgments

Thanks to Markus Stenberg for helping develop the policy regarding the four styles of unicast reply.

## 10. References

### 10.1. Normative References

- [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.
- [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3927] Cheshire, S., Aboba, B., and E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses", RFC 3927, May 2005.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.
- [RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, December 2012.
- [RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, December 2012.
- [I-D.sekar-dns-llq] Sekar, K., "DNS Long-Lived Queries", draft-sekar-dns-llq-01 (work in progress), August 2006.

### 10.2. Informative References

- [IPR2119] "Apple Inc.'s Statement about IPR related to Hybrid Unicast/Multicast DNS-Based Service Discovery", <<https://datatracker.ietf.org/ipr/2119/>>.
- [RFC2136] Vixie, P., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 1997.
- [RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, November 2000.
- [RFC6760] Cheshire, S. and M. Krochmal, "Requirements for a Protocol

to Replace the AppleTalk Name Binding Protocol (NBP)",  
RFC 6760, December 2012.

[ZC] Cheshire, S. and D. Steinberg, "Zero Configuration  
Networking: The Definitive Guide", O'Reilly Media, Inc. ,  
ISBN 0-596-10100-7, December 2005.

#### Author's Address

Stuart Cheshire  
Apple Inc.  
1 Infinite Loop  
Cupertino, California 95014  
USA

Phone: +1 408 974 3207  
Email: cheshire@apple.com



DNS-SD/mDNS Extensions  
Internet-Draft  
Intended status: Informational  
Expires: January 5, 2015

K. Lynn, Ed.  
Consultant  
S. Cheshire  
Apple, Inc.  
M. Blanchet  
Viagenie  
D. Migault  
Orange  
July 4, 2014

Requirements for Scalable DNS-SD/mDNS Extensions  
draft-ietf-dnssd-requirements-03

Abstract

DNS-SD/mDNS is widely used today for discovery and resolution of services and names on a local link, but there are use cases to extend DNS-SD/mDNS to enable service discovery beyond the local link. This document provides a problem statement and a list of requirements.

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, 2015.

Copyright Notice

Copyright (c) 2014 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

1. Introduction . . . . .	2
2. Problem Statement . . . . .	3
3. Basic Use Cases . . . . .	5
4. Requirements . . . . .	6
5. Namespace Considerations . . . . .	8
6. Security Considerations . . . . .	8
7. IANA Considerations . . . . .	10
8. Acknowledgments . . . . .	10
9. References . . . . .	10
Authors' Addresses . . . . .	11

## 1. Introduction

DNS-Based Service Discovery [DNS-SD] in combination with its companion technology Multicast DNS [mDNS] is widely used today for discovery and resolution of services and names on a local link. However, as users move to multi-link home or campus networks they find that mDNS does not work across routers. DNS-SD can also be used in conjunction with conventional unicast DNS to enable wide-area service discovery, but this capability is not yet widely deployed. This disconnect between customer needs and current practice has led to calls for improvement, such as the Educause petition [EP].

In response to this and similar evidence of market demand, several products now enable service discovery beyond the local link using different ad-hoc techniques. As yet, no consensus has emerged regarding which approach represents the best long-term direction for DNS-based service discovery protocol development.

mDNS in its present form is also not optimized for network technologies where multicast transmissions are relatively expensive. Wireless networks such as [IEEE.802.11] may be adversely affected by excessive mDNS traffic due to the higher network overhead of multicast transmissions. Wireless mesh networks such as 6LoWPAN [RFC4944] are effectively multi-link subnets [RFC4903] where multicasts must be forwarded by intermediate nodes.

It is in the best interests of end users, network administrators, and vendors for all interested parties to cooperate within the context of the IETF to develop an efficient, scalable, and interoperable standards-based solution.

This document defines the problem statement and gathers requirements for Scalable DNS-SD/mDNS Extensions.

### 1.1. Requirements Language

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 "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

### 1.2. Terminology and Acronyms

**Service:** An endpoint (host and port) for a given application protocol. Services are identified by Service Instance Names.

**DNS-SD:** DNS-Based Service Discovery, as specified in [DNS-SD], is a conventional application of DNS Resource Records and messages to facilitate the discovery and location of services.

**mDNS:** Multicast DNS, as specified in [mDNS], is a transport protocol that facilitates DNS-SD on a local link in the absence of DNS infrastructure.

**SSD:** Scalable DNS-SD is a future extension of DNS-SD (and perhaps mDNS) that meets the requirements set forth in this document.

**Scope of Discovery:** A subset of a local or global namespace, e.g., a DNS zone, that is the target of a given SSD query.

**Zero Configuration:** A deployment of SSD that requires no administration (although some administration may be optional).

**Incremental Deployment:** An orderly transition, as a network installation evolves, from DNS-SD/mDNS to SSD.

## 2. Problem Statement

Service discovery beyond the local link is perhaps the most important feature currently missing from the DNS-SD/mDNS framework. Other issues and requirements are summarized below.

### 2.1. Multi-link Naming and Discovery

A list of desired DNS-SD/mDNS improvements from network administrators in the research and education community was issued in the form of the Educause petition [EP]. The following is a summary of the technical issues:

- o Products that advertise services such as printing and multimedia streaming via DNS-SD/mDNS are not currently discoverable by devices on other links. It is common practice for enterprises and institutions to use wireless links for client access and wired networks for server infrastructure, typically on different subnets. DNS-SD used with conventional unicast DNS does work when devices are on different links, but the resource records that describe the service must somehow be entered into the unicast DNS namespace.
- o DNS-SD resource records may be entered manually into a unicast DNS zone file, but this task must be performed by a DNS administrator. It is labor-intensive and brittle when IP addresses of devices change dynamically, as is common when DHCP is used.
- o Automatically adding DNS-SD records using DNS Update works, but requires that the DNS server be configured to allow DNS Updates, and requires that devices be configured with the DNS Update credentials to permit such updates, which has proven to be onerous.
- o Therefore, a mechanism is desired that populates the DNS namespace with the appropriate DNS-SD records with less manual administration than typically needed for a unicast DNS server.

The following is a summary of the technical requirements:

- o It must scale to a range of hundreds to thousands of DNS-SD/mDNS enabled devices in a given environment.
- o It must simultaneously operate over a variety of network link technologies, such as wired and wireless networks.
- o It must not significantly increase network traffic (wired or wireless).
- o It must be cost-effective to manage at up to enterprise scale.

## 2.2. IEEE 802.11 Wireless LANs

Multicast DNS was originally designed to run on Ethernet - the dominant link-layer at the time. In shared Ethernet networks, multicast frames place little additional demand on the shared network medium compared to unicast frames. In IEEE 802.11 networks however, multicast frames are transmitted at a low data rate supported by all receivers. In practice, this data rate leads to a larger fraction of airtime being devoted to multicast transmission. Some network



administrators block multicast traffic or convert it to a series of link-layer unicast frames.

Wireless links may be orders of magnitude less reliable than their wired counterparts. To improve transmission reliability, the IEEE 802.11 MAC requires positive acknowledgement of unicast frames. It does not, however, support positive acknowledgement of multicast frames. As a result, it is common to observe much higher loss of multicast frames on wireless as compared to wired network technologies.

Enabling service discovery on IEEE 802.11 networks requires that the number of multicast frames be restricted to a suitably low value, or replaced with unicast frames to use the MAC's reliability features.

### 2.3. Low Power and Lossy Networks (LLNs)

Emerging wireless mesh networking technologies such as RPL [RFC6550] and 6LoWPAN present several challenges for the current DNS-SD/mDNS design. First, Link-Local multicast scope [RFC4291] is defined as a single-hop neighborhood. A single subnet prefix in a wireless mesh network may often span multiple links, therefore a larger multicast scope is required to span it [I-D.ietf-6man-multicast-scopes]. mDNS is not currently specified for greater than Link-Local scope.

Additionally, low-power nodes may be offline for significant periods either because they are "sleeping" or due to connectivity problems. In such cases LLN nodes might fail to respond to queries or defend their names using the current design.

### 3. Basic Use Cases

The following use cases are defined with different characteristics to help motivate, distinguish, and classify the target requirements. They cover a spectrum of increasing deployment and administrative complexity.

(A) Personal Area networks (PANs): the simplest example of a network may consist of a single client and server, e.g., one laptop and one printer, on a common link. PANs that do not contain a router may use Zero Configuration to assign network addresses and mDNS to provide naming and service discovery.

(B) Classic home or 'hotspot' networks, consisting of:

- \* Single exit router: the network may have multiple upstream providers or networks, but all outgoing and incoming traffic goes through a single router.

- \* One-level depth: multiple links on the network are bridged to form a single subnet, which is connected to the default router.
- \* Single administrative domain: all nodes under the same admin entity. (However, this does not necessarily imply a network administrator.)

(C) Advanced home and small business networks  
[I-D.ietf-homenet-arch]:

Like B but consist of multiple wired and/or wireless links, connected by routers, behind the single exit router. However, the forwarding nodes are largely self-configuring and do not require routing protocol administration. Such networks should also not require DNS administration.

(D) Enterprise networks:

Like C but consist of arbitrary network diameter under a single administrative domain. A large majority of the forwarding and security devices are configured. Large-scale conference-style networks, which are predominantly wireless access, e.g., as available at IETF meetings, also fall within this category.

(E) Higher Education networks:

Like D but the core network may be under a central administrative domain while leaf networks are under local administrative domains.

(F) Mesh networks such as RPL/6LoWPAN:

Multi-link subnets with prefixes defined by one or more border routers. May comprise any part of networks C, D, or E.

#### 4. Requirements

Any successful SSD solution(s) will have to strike the proper balance between competing goals such as scalability, deployability, and usability. With that in mind, none of the requirements listed below should be considered in isolation.

REQ1: For use cases A, B, and C, there should be a Zero Configuration mode of operation. This implies that servers and clients should be able to automatically determine a default Scope of Discovery in which to advertise and discover services, respectively.

- REQ2: For use cases C, D, and E, there should be a way to configure Scopes of Discovery that support a range of topologically-independent zones (e.g., from department to campus-wide). If multiple scopes are available, there must be a way to enumerate the choices from which a selection can be made.
- REQ3: As stated in REQ2 above, the discovery scope need not be aligned to network topology. For example, it may instead be aligned to physical proximity or organizational structure.
- REQ4: For use cases C, D, and E, there should be an incremental way to deploy the solution.
- REQ5: SSD should integrate with current link scope DNS-SD/mDNS protocols and deployments.
- REQ6: SSD must not adversely affect or break any other current protocols or deployments.
- REQ7: SSD must be capable of operating across networks that are not limited to a single link or network technology, including clients and services on non-adjacent links.
- REQ8: It is desirable that a user or device, when away from such a site, is still able to discover services within that site, e.g., a user discovering services in their home network while remote from it.
- REQ9: SSD should operate efficiently in all networks, with particular consideration for potentially lossy or multicast-challenged wireless networks.
- REQ10: SSD should be considerate of networks where power consumption is a critical factor and, for example, nodes may be in a low power or sleeping state.
- REQ11: SSD must be scalable to thousands of nodes with minimal configuration and without degrading network performance. A possible figure of merit is that, as the number of services increases, the amount of traffic due to SSD on a given link remains relatively constant.
- REQ12: SSD should enable a way to provide a consistent user experience whether local or global services are being discovered.

- REQ13: The information presented by SSD should reflect reality. That is, new information should be available in a timely fashion and stale information should not persist.
- REQ14: SSD should operate over existing networks (as described by use cases A-F above) without requiring changes to the network technology or deployment.

## 5. Namespace Considerations

The unicast DNS namespace contains globally unique names. The mDNS namespace contains locally unique names. Clients discovering services may need to differentiate between local and global names or to determine that names in different namespaces identify the same service.

Multiple devices in different subnets may share the same label (perhaps due to vendor defaults) or have similarly appearing labels. This may lead to a local label disambiguation problem between presented results.

SSD should support rich internationalized labels within Service Instance Names, as DNS-SD/mDNS does today. SSD must not negatively impact the global DNS namespace or infrastructure.

The problem of publishing local services in the global DNS namespace may be generally viewed as exporting local resource records and their associated labels into some DNS zone. The issues related to defining labels that are interoperable between local and global namespaces are discussed in [I-D.sullivan-dnssd-mdns-dns-interop].

## 6. Security Considerations

Insofar as SSD may automatically gather DNS-SD resource records and publish them over a wide area, the security issues are likely to be the union of those discussed in [mDNS] and [DNS-SD]. The following sections highlight potential threats that are posed by deploying DNS-SD over multiple links or by automating DNS-SD administration.

### 6.1. Scope of Discovery

As mDNS is currently restricted to a single link, the scope of the advertisement is limited, by design, to the shared link between client and server. In a multi-link scenario, the owner of the advertised service may not have a clear indication of the scope of its advertisement.

If the advertisement propagates to a larger set of links than expected, this may result in unauthorized clients (from the perspective of the owner) discovering and then potentially attempting to connect to the advertised service. It also discloses information (about the host and service) to a larger set of potential attackers.

Note that discovery of a service does not necessarily imply that the service is reachable or can be connected to. Specific access control mechanisms are out of scope of this document.

If the scope of the discovery is not properly setup or constrained, then information leaks will happen outside the appropriate network.

## 6.2. Multiple Namespaces

There is a possibility of conflicts between the local and global DNS namespaces. Without adequate feedback, a discovering client may not know if the advertised service is the correct one, therefore enabling potential attacks.

## 6.3. Authorization

DNSSEC can assert the validity but not the veracity of records in a zone file. The trust model of the global DNS relies on the fact that human administrators either a) manually enter resource records into a zone file, or b) configure the DNS server to authenticate a trusted device (e.g., a DHCP server) that can automatically maintain such records.

An imposter may register on the local link and appear as a legitimate service. Such "rogue" services may then be automatically registered in unicast DNS-SD.

## 6.4. Authentication

Up to now, the "plug-and-play" nature of mDNS devices has relied only on physical connectivity. If a device is visible via mDNS then it is assumed to be trusted. This is not likely to be the case in foreign networks.

If there is a risk that clients may be fooled by the deployment of rogue services, then application layer authentication should be considered as part of any security solution. Authentication of any particular service is outside the scope of this document.

## 6.5. Privacy Considerations

Mobile devices such as smart phones that can expose the location of their owners by registering services in arbitrary zones pose a risk to privacy. Such devices must not register their services in arbitrary zones without the approval ("opt-in") of their users. However, it should be possible to configure one or more "safe" zones in which mobile devices may automatically register their services.

## 7. IANA Considerations

This document currently makes no request of IANA.

Note to RFC Editor: this section may be removed upon publication as an RFC.

## 8. Acknowledgments

We gratefully acknowledge contributions and review comments made by RJ Atkinson, Tim Chown, Guangqing Deng, Ralph Droms, Educause, David Farmer, Matthew Gast, Thomas Narten, David Thaler, and Peter Van Der Stok.

## 9. References

### 9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.
- [RFC4903] Thaler, D., "Multi-Link Subnet Issues", RFC 4903, June 2007.
- [RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.
- [RFC6550] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, March 2012.
- [mDNS] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, February 2013.

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

## 9.2. Informative References

- [I-D.ietf-6man-multicast-scopes]  
Droms, R., "IPv6 Multicast Address Scopes", draft-ietf-6man-multicast-scopes-07 (work in progress), June 2014.
- [I-D.ietf-homenet-arch]  
Chown, T., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", draft-ietf-homenet-arch-16 (work in progress), June 2014.
- [I-D.sullivan-dnssd-mdns-dns-interop]  
Sullivan, A., "Requirements for Labels to Interoperate Between mDNS and DNS", draft-sullivan-dnssd-mdns-dns-interop-00 (work in progress), January 2014.
- [EP] "Educause Petition", <https://www.change.org/petitions/from-educause-higher-ed-wireless-networking-admin-group>, July 2012.
- [IEEE.802.11]  
"Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", IEEE Std 802.11-2012, 2012, <<http://standards.ieee.org/getieee802/download/802.11-2012.pdf>>.
- [static] "Manually Adding DNS-SD Service Discovery Records to an Existing Name Server", July 2013, <<http://www.dns-sd.org/ServerStaticSetup.html>>.

## Authors' Addresses

Kerry Lynn (editor)  
Consultant

Phone: +1 978 460 4253  
Email: [kerlyn@ieee.org](mailto:kerlyn@ieee.org)

Stuart Cheshire  
Apple, Inc.  
1 Infinite Loop  
Cupertino , California 95014  
USA

Phone: +1 408 974 3207  
Email: cheshire@apple.com

Marc Blanchet  
Viagenie  
246 Aberdeen  
Quebec , Quebec G1R 2E1  
Canada

Email: Marc.Blanchet@viagenie.ca  
URI: <http://www.viagenie.ca>

Daniel Migault  
Orange  
38-40 rue du General Leclerc  
Issy-les-Moulineaux 92130  
France

Phone: +33 1 45 29 60 52  
Email: mglt.biz@gmail.com



DNSSD  
INTERNET-DRAFT  
Intended Status: Informational  
Expires: December 10, 2014

H. Rafiee  
Huawei Technologies  
June 10, 2014

Multicast DNS (mDNS) Threat Model and Security Consideration  
<draft-rafiee-dnssd-mdns-threatmodel-00.txt>

Abstract

This document describes threats associated with extending multicast DNS (mDNS) across layer 3.

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 December 10, 2014.

Copyright Notice

Copyright (c) 2014 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

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Threat Analysis . . . . .	4
3.1. mDNS Gateway is a single point of failure . . . . .	4
3.2. Large Traffic Production from mDNS Gateway . . . . .	4
3.3. DoS attack on any node in the mDNS enabled network . . . . .	5
3.4. Good mDNS gateway goes bad . . . . .	5
3.5. Fake mDNS gateway . . . . .	5
3.6. MAC address spoofing . . . . .	5
3.6.1. possible solution . . . . .	5
3.7. Cache Poisoning . . . . .	5
3.7.1. Possible solution . . . . .	6
3.8. Malicious update on unicast DNS . . . . .	6
3.9. Harming Privacy . . . . .	6
3.10. IP spoofing . . . . .	6
3.11. Resource spoofing . . . . .	7
3.12. Internet Group Management Protocol (IGMP) Attacks . . . . .	7
3.13. Multicast Listener Discovery (MLD) attacks . . . . .	7
3.14. Fake Resource Advertisement . . . . .	7
3.15. Dual stack attacks . . . . .	7
4. Possible solutions . . . . .	7
4.1. SAVI-DHCP . . . . .	7
4.2. DNS over TLS . . . . .	8
4.3. CGA-TSIG . . . . .	8
4.4. DNS Security Extension . . . . .	8
4.5. SSAS . . . . .	8
4.6. IPsec . . . . .	8
5. Security Considerations . . . . .	8
6. IANA Considerations . . . . .	8
7. Acknowledgements . . . . .	9
8. References . . . . .	9
8.1. Normative . . . . .	9
8.2. Informative . . . . .	9
Authors' Addresses . . . . .	11

## 1. Introduction

Multicast DNS (mDNS) was proposed in [RFC6762] to allow nodes in local links to use DNS-like names for their communication without the need for global DNS servers, infrastructure and administration processes for configuration. mDNS along with service discovery (DNS-SD) [RFC6763] provides nodes with the possibility to discover other services and the names of other nodes with zero configuration, i.e., connect a node into a local link and use resources such as a printer that are available in that network.

mDNS and service discovery use DNS-like query messages. The main assumption is that these services also use DNS security protocols such as DNSSEC. However, due to the limitation of DNSSEC in local link, i.e., the key authorization and configuration needed for DNSSEC, it is not easy to use this protocol for zero configuration services. This is why the current implementations use no security in local links and are vulnerable to several attacks.

The purpose of this document is to introduce threat models for mDNS and service discovery and allow implementers to be aware of the possible attacks in order to mitigate them with possible solutions. Since there are already old lists of known DNS threats available in [RFC3833], here we only analyze the ones that are which is applicable to mDNS. We also introduce new possible threats that could result from extending mDNS scope.

## 2. Terminology

Node: any host and routers in the network

Attack: an action to exploit a node and allow the attacker to gain access to that node. It can be also an action to prevent a node from providing a service or using a service on the network

Attacker: a person who uses any node in the network to attack other nodes using known or unknown threats

Threat: Anything that has a potential to harm a node in the network

Local link vulnerability: Any flaws that are the result of the assumption that a malicious node could gain access to legitimate nodes inside a local link network

Wide Area Network (WAN) vulnerability: Any flaws that are the result of the assumption that a malicious node could gain access to legitimate nodes inside any local links in an enterprise network with multiple Local Area Networks (LANs) or Virtual LANs (VLANs).

Host name: Fully qualified DNS Name (FQDN) of a node in the network

Constrained device: a small device with limited resources (battery, memory, etc.)

### 3. Threat Analysis

mDNS/DNS-SD cannot use DNSSEC approaches for security purposes. This is because, as mentioned earlier, DNSSEC is not a zero config protocol and it is not compatible with the plug and play nature of mDNS/DNS-SD. This is why mDNS is vulnerable to several attacks. Most threats in this section are a result of spoofing, Denial of Service (DoS), or a combination of them. Here we explain them in different example scenarios.

#### 3.1. mDNS Gateway is a single point of failure

An mDNS gateway needs to process all queries sent to/from different networks that this gateway is connected to and filters the traffic based on the policy explained in section 3.4 [mdns-extend]. A malicious node in any of these subnets can send several queries and carry out the DoS attack on these gateways.

#### 3.2. Large Traffic Production from mDNS Gateway

There are several scenarios associated with the Large Traffic Production case.

First scenario: a malicious node in any of the subnets that the gateway connects can advertise different fake services or spoof the information of the real services and replay the messages. This causes large traffic either in the local link or in other links since the gateway was also supposed to replicate the traffic to other links.

Second scenario : a malicious node spoofs the legitimate service advertisements of different nodes in the network and changes the Time To Leave (TTL) value to zero. This will result in producing large traffic since the mDNS gateway needs to ask all of the service advertisers to re-advertise their service. This is an especially effective attack in a network of constrained devices because it causes more energy consumption.

Third scenario: Since a hybrid proxy [hybrid-proxy] node aggregates all data and sends it back to the requester, a malicious node can generate several queries that produce large responses, spoof the source or MAC address of a victim node in this network, and forward all traffic to this victim node.

Fourth scenario: A malicious node can replay hybrid proxy aggregation messages [hybrid-proxy] and cause a DoS on a victim node.

### 3.3. DoS attack on any node in the mDNS enabled network

A malicious node spoofs the MAC address and source IP address of a legitimate victim node in this network and questions several services in the link. This will result in a large traffic return to the victim node from both mDNS gateway and also the service owner.

A malicious node can send a spoofed service probe message and direct all traffic to any victim node to this network (section 3.5 [mdns-extend]).

Second scenario: a malicious node claims the ownership of any name that the resource requester or a node uses and does not let the nodes choose a unique desired name for their service or for the devices.

### 3.4. Good mDNS gateway goes bad

mDNS gateway is compromised and submits wrong information to the links to which it is connected.

### 3.5. Fake mDNS gateway

A malicious node can play a role of gateway in any of those subnets and play a Man in the Middle (MITM) attack. Since the messages sent from gateway are usually unicast, no other nodes will detect these malicious activities of this fake gateway. (section 3.8.1 [mdns-extend]). This malicious node can then respond to any DNS-SD messages and play a role of passive gateway.

### 3.6. MAC address spoofing

In a wireless environment where [mdns-extend] is suggested to use MAC address filtering to avoid any malicious node joining to the network, a malicious node can easily spoof the MAC address of a legitimate node and join the network and perform malicious activities.

#### 3.6.1. possible solution

Filtering can be based on the signature of the public key and MAC address of the devices. This process might be through manual adding of this signature to the whitelist filter. The verification is the process of verifying the signature signed by the private key and the public key signature. This solution might require some manual step and changes on the current implementation to filter based on this signature.

### 3.7. Cache Poisoning

mDNS gateway stores all of the information related to the available wireless nodes in its cache. In section 3.8.1 [mdns-extend], it is not clear how mDNS gateway knows when a node leaves a wireless link. If the node sends a "leave message" to mDNS gateway, a malicious node can send this message on behalf of a legitimate node and presume that that the legitimate node does not exist in that link, thereby causing delay or possible problems in offering service to that node.

Second scenario: a malicious node can send a location update message to mDNS home gateway and cause delay in offering services to a legitimate node.

Third scenario: similar to Mobile IPv6 [RFC6275] possible attacks, a malicious node can start large traffic from a streaming server and then send a fake ?location update message? to the home mDNS gateway and send a update message with a different, spoofed source IP address. This will forward all of the large streaming traffic to a victim node.

Forth scenario: To decrease traffic in the network [hybrid-proxy], a hybrid proxy aggregates all answers received from different resources and sends a unicast DNS message on behalf of all of the resources to the resource requester. A malicious node can play the role of hybrid proxy and poison the cache of resource requester.

#### 3.7.1. Possible solution

IPsec can prevent this attack but it is not a zero configuration protocol and it needs a way to provide the initial trust between both end points of communication.

#### 3.8. Malicious update on unicast DNS

A malicious node can spoof the content of DNS update message and add malicious records to unicast DNS.

#### 3.9. Harming Privacy

If a malicious node is in any subnet (WLAN and WAN) of a network, it can learn about all services available in this network. The combination of mDNS and DNS-SD discloses some critical information about resources in this network which might be harmful to privacy.

#### 3.10. IP spoofing

A malicious node spoofs the content of Dynamic Host Configuration Protocol (DHCP) server messages and offers its own malicious information to the nodes in the network.

### 3.11. Resource spoofing

Resource owners in the network have permission to have the same name for load balancing. A malicious node can claim to be one of the load balanced resource devices and maliciously respond to requests.

### 3.12. Internet Group Management Protocol (IGMP) Attacks

IGMP that is suggested to be used in network bridging scenario [mdns-x] can be maliciously used by an attacker. Spoofing and DoS attacks are two sources of attack in IGMP protocol. A complete list of these attacks can be found in [IGMP-Attack].

### 3.13. Multicast Listener Discovery (MLD) attacks

The same as IGMP attacks, since these are signaling protocols, a simple DoS attack can use a lot of resources and produce large traffic. This is because a malicious node can send MLD to subscribe to a large number of high-bandwidth multicast groups. It can then cause bandwidth exhaustion, leading to a DoS. It might also lead to using more CPU resources on the nodes. This will be quite critical for constrained devices.

### 3.14. Fake Resource Advertisement

A malicious node in any subnet can advertise fake resources. The other nodes have no possibility to authenticate this node and authorize its resources. This can happen in both mDNS gateway scenario and hybrid proxy [hybrid-proxy].

### 3.15. Dual stack attacks

Having both IPv4 and IPv6 in the same network and trying to aggregate service discovery traffic on both IP stacks might cause new security flaws during the conversion or aggregation of this traffic. It can be similar to what explained here as an aggregated traffic or lead to a wide range of spoofing attacks.

## 4. Possible solutions

Since spoofing is the main source of attacks for many malicious activities, using approaches that can prevent IP spoofing or provide a means of secure authentication with minimum configuration is helpful.

### 4.1. SAVI-DHCP

SAVI-DHCP [DHCP-SAVI] approach uses a simple mechanism in switches or devices that knows information about the ports of switches to filter any malicious traffic. This mitigates attacks on DHCP server spoofing

#### 4.2. DNS over TLS

The approaches in this category are discussed in DANE WG. It might be a good solution to automate the authentication processes or avoid spoofed DNS update messages

#### 4.3. CGA-TSIG

CGA-TSIG [cga-tsig] is another possible solution that can provide the node with secure authentication, data integrity and data confidentiality. The new version supports both IPv4 and IPv6. It provides the node with zero or minimal configuration.

#### 4.4. DNS Security Extension

Due to the manual step requirement for DNSSEC configuration on each nodes and DNS servers, it is not an ideal solution mechanism for zero config services.

#### 4.5. SSAS

SSAS [ssas] can prevent the nodes from IP spoofing. This is dissimilar to other approach, CGA [RFC3972] that can only support IPv6 networks. The new version of this document supports both IPv4 and IPv6. It also offers a solution for MAC spoofing, however, due to operational barriers, MAC spoofing solution might not work well.

#### 4.6. IPsec

IPsec is another security protection mechanism. Similar to DNSSEC, it requires manual step for the configuration of the nodes. However, recently there are some new drafts to automate this process.

### 5. Security Considerations

There is no security consideration

### 6. IANA Considerations

There is no IANA consideration



## 7. Acknowledgements

The author would like to thank all those people who directly helped in improving this draft, especially John C. Klensin

## 8. References

### 8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6762] Cheshire, S., Krochmal, M., "Multicast DNS", RFC 6762, February 2013
- [RFC6763] Cheshire, S., Krochmal, M., "DNS-Based Service Discovery", RFC 6763, February 2013
- [RFC6275] Perkins, C., Johnson, D., Arkko, J., "Mobility Support in IPv6", RFC 6275, July 2011
- [RFC3833] Atkins, D., Austein, R., "Threat Analysis of the Domain Name System (DNS)", RFC 3833, August 2004

### 8.2. Informative References

- [mdns-extend] Bhandari, S., Fajalia, B., Schmieder, R., Orr, S., Dutta, A., "Extending multicast DNS across local links in Campus and Enterprise networks", <http://tools.ietf.org/html/draft-bhandari-dnssd-mdns-gateway-00>, October 2013
- [mdns-x] Otis, D., "mDNS X-link review", <http://tools.ietf.org/html/draft-otis-dnssd-mdns-xlink-03>, April 2014
- [IGMP-Attack] [http://www.securemulticast.org/GSEC/gsec3\\_ietf53\\_SecureIGMP1.pdf](http://www.securemulticast.org/GSEC/gsec3_ietf53_SecureIGMP1.pdf)
- [hybrid-proxy] Cheshire, S., "Hybrid Unicast/Multicast DNS-Based Service Discovery", <http://tools.ietf.org/html/draft-cheshire-dnssd-hybrid-01>, January 2014
- [DHCP-SAVI] Bi, J., Wu, J., Yao, G, Baker, F., "SAVI Solution for DHCP", <http://tools.ietf.org/html/draft-ietf-savi-dhcp-23>, April

2014

[cga-tsig] Rafiee, H., Loewis, M., Meinel, C., "Transaction  
SIGNature (TSIG) using CGA Algorithm in IPv6",  
<http://tools.ietf.org/html/draft-rafiiee-intarea-cga-tsig> ,  
February 2014

[ssas] Rafiee, H., Meinel, C., "SSAS: a Simple Secure  
Addressing Scheme for IPv6 AutoConfiguration".  
<http://tools.ietf.org/search/draft-rafiiee-6man-ssas>, 2013

Authors' Addresses

Hosnieh Rafiee  
<http://www.rozanak.com>  
Phone: +49 176 57 58 75 75  
Email: [ietf@rozanak.com](mailto:ietf@rozanak.com)

