Network Working Group S. Cheshire
Internet-Draft Apple Inc.
Intended status: Standards Track T. Lemon
Expires: May 3, 2018 Barefoot Consulting
October 30, 2017

Multicast DNS Discovery Relay
draft-sctl-dnssd-mdns-relay-01

Abstract

This document extends the Discovery Proxy for Multicast DNS-Based Service Discovery specification. It describes a lightweight relay mechanism, a Discovery Relay, which allows Discovery Proxies to provide service on links to which the hosts on which they are running are not directly attached.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Discovery Proxy for Multicast DNS-Based Service Discovery [I-D.ietf-dnssd-hybrid] specification defines a mechanism for discovering services on a subnetted network using Multicast DNS (mDNS) [RFC6762], through the use of Discovery Proxies, which issue mDNS requests on various links in the network on behalf of a host attempting service discovery.

In the original Discovery Proxy specification, it is assumed that for every link on which services will be discovered, a host will be present running a full Discovery Proxy. This document introduces a lightweight Discovery Relay which can be used to provide discovery services on a link without requiring a full Discovery Proxy on every link.

The Discovery Relay operates by listening for TCP connections from Discovery Proxies. When a Discovery Proxy connects, the connection is authenticated and secured using TLS. The Discovery Proxy can then send messages that will be relayed to specified links. The Discovery Proxy may also specify one or more links from which it wishes to receive mDNS traffic. DNS Session Signaling [I-D.ietf-dnsop-session-signal] is used as a framework for conveying interface and IP header information associated with each message.

The Discovery Relay functions essentially as a set of one or more virtual interfaces for the Discovery proxy, one on each link to which the Discovery Relay is connected. In a complex network, it is possible that more than one Discovery Relay will be connected to the same link; in this case, the Discovery Proxy ideally should only be using one such Relay Proxy per link, since using more than one will generate duplicate traffic.

How such duplication is detected and avoided is out of scope for this document: in principle it could be detected using HNCP [RFC7788] or configured using some sort of orchestration software in conjunction with NETCONF [RFC6241] or CPE WAN Management Protocol [TR-069].

2. Terminology

The following definitions may be of use:

mDNS Agent A host which sends and/or responds to mDNS queries.

Discovery Proxy A network service which receives well-formed questions using the DNS protocol, performs multicast DNS queries to answer those questions, and responds with those answers using the DNS protocol.

Discovery Relay A network service which sends mDNS messages on behalf of a Discovery Proxy and relays mDNS messages to a Discovery Relay.

link A maximal set of network connection points such that any host connected to any connection point may send a packet to a host connected to any other connection point without the help of a layer 3 router.

whitelist A list of one or more IP addresses from which a Discovery Relay may accept connections.

silently discard When a message that is not supported or not permitted is received, and the required response to that message is to "silently discard" it, that means that no response is sent by the service that is discarding the message to the service that sent it. The service receiving the message may log the event, and may also count such events: "silently" does not preclude such behavior.

3. Protocol Overview

This document describes a way for Discovery Proxies to communicate with mDNS agents on networks to which they are not directly connected using a Discovery Relay. As such, there are two parts to the protocol: connections between Discovery Proxies and Discovery Relays, and communications between Discovery Relays and mDNS agents.

3.1. Connections between Discovery and Discovery Relays

Discovery Relays listen for connections. Connections between Discovery Proxies and Discovery Relays are established by Discovery Proxies. Connections are authenticated and encrypted using TLS, with both client and server certificates. Connections are long-lived: a Discovery Proxy is expected to send many queries over the same connection, and Discovery Relays will forward all mDNS traffic from subscribed interfaces over the connection.

The stream encapsulated in TLS will carry DNS frames as in the DNS TCP protocol [RFC1035] Section 4.2.2. However, all messages will be DNS Session Signaling messages [I-D.ietf-dnsop-session-signal]. There will be three types of such messages:

o Subscribe messages from Discovery Proxy to Discovery Relay

o mDNS messages from Discovery Proxy to Discovery Relay

o mDNS messages from Discovery Relay to Discovery Proxy

Subscribe messages from the Discovery Proxy to the Discovery Relay indicate to the Discovery Relay that mDNS messages from one or more specified links are to be relayed to the Discovery Proxy.

mDNS messages from a Discovery Proxy to a Discovery Relay cause the Discovery Relay to re-transmit the mDNS message on one or more links to which the Discovery Relay host is directly attached.

mDNS messages from a Discovery Relay to a Discovery Proxy are sent whenever an mDNS message is received on a link to which the Discovery Relay has subscribed.

Discovery Relays are responsible for keeping connections alive when no traffic has been sent during a keepalive period (See [I-D.ietf-dnsop-session-signal] Section 4).

3.2. mDNS Messages On Links

Discovery Relays listen for mDNS traffic on all configured links. When a mDNS message is received on a link, it is forwarded on every open Discovery Proxy connection that is subscribed to mDNS traffic on that link. In the event of congestion, where a particular Discovery Proxy connection has no buffer space for an mDNS message that would otherwise be forwarded to it, the mDNS message is not forwarded to it. Normal mDNS retry behavior is used to recover from this sort of packet loss. Discovery Relays are not expected to buffer more than a few mDNS packets.

Discovery Relays accept mDNS traffic from Discovery Proxies. Such traffic is forwarded to zero or more more links to which the Discovery Relay host is directly connected.

4. Connections between Discovery Proxies and Discovery Relays

When a Discovery Relay starts, it opens a passive TCP listener to receive connections from Discovery Proxies. This listener may be bound to one or more source IP addresses, or to the wildcard address, depending on the TCP implementation. When a connection is received, the relay must first validate that it is a connection to an IP address to which connections are allowed. For example, it may be that only connections to ULAs are allowed, or to the IP addresses configured on certain interfaces. If the listener is bound to a specific IP address, this check is unnecessary.

The relay must then validate that the source IP address of the connection is on its whitelist. If the connection is not permitted either because of the source address or the destination address, the Discovery Relay responds to the TLS Client Hello message from the Discovery Proxy with a TLS user_canceled alert ([I-D.ietf-tls-tls13] Section 6.1).

Otherwise, the Discovery Relay will attempt to complete a TLS handshake with the Discovery Proxy. Discovery Proxies are required to send the post_handshake_auth extension ([I-D.ietf-tls-tls13] Section 4.2.5). If a relay proxy receives a ClientHello message with no post_handshake_auth extension, the Discovery Relay rejects the

connection with a certificate_required alert ([I-D.ietf-tls-tls13] Section 6.2).

Once the TLS handshake is complete, the Discovery Relay MUST request post-handshake authentication as described in ([I-D.ietf-tls-tls13] Section 4.6.2). If the Discovery Proxy refuses to send a certificate, or the key presented does not match the key associated with the IP address from which the connection originated, or the CertificateVerify does not validate, the connection is dropped with the TLS access_denied alert ([I-D.ietf-tls-tls13] Section 6.2).

Once the connection is established and authenticated, it is treated as a DNS TCP connection [RFC1035].

Aliveness of connections between Discovery Proxies and Relays is maintained as described in Section 4 of [I-D.ietf-dnsop-session-signal]. Discovery Proxies must also honor the 'Retry Delay' TLV (section 5 of [I-D.ietf-dnsop-session-signal]) if sent by the Discovery Relay.

Discovery Proxies may establish more than one connection to a specific Discovery Relay. This would happen in the case that a TCP connection stalls, and the Discovery Proxy is able to reconnect before the previous connection has timed out. It could also happen as a result of a server restart. It is not likely that two active connections from the same Discovery Proxy would be present at the same time, but it must be possible for additional connections to be established. The Discovery Relay may drop the old connection when the new one has been fully established, including a successful TLS handshake. What it means for two connections to be from the same Discovery Proxy is that the connections both have source addresses that belong to the same proxy, and that they were authenticated using the same client certificate.

5. Traffic from Relays to Proxies

The mere act of connecting to a Discovery Relay does not result in any mDNS traffic being forwarded. In order to request that mDNS traffic from a particular link be forwarded on a particular connection, the Discovery Proxy must send a session signaling message containing one or more MDNS Link Request TLVs (Section 8.1) indicating the link from which traffic is requested.

When such a message is received, the Discovery Relay validates that each specified link is available for forwarding, and that forwarding is enabled for that link. For each such message the Discovery Relay validates each link specified and includes in a single response a list of zero or more MDNS Link Invalid TLVs)Section 8.2) for links

that are not valid, and zero or more MDNS Link Subscribed TLVs (Section 8.3) for links that are valid. For each valid link, it begins forwarding all mDNS traffic from that link to the Discovery Proxy. Delivery is not guaranteed: if there is no buffer space, packets will be dropped. It is expected that regular mDNS retry processing will take care of retransmission of lost packets. The amount of buffer space is implementation dependent, but generally should not be more than the bandwidth delay product of the TCP connection [RFC1323].

mDNS messages from Relays to Proxies are framed within DNS Session Signaling messages. This allows multiple TLVs to be included. Each forwarded mDNS message is contained in an MDNS Message TLV Section 8.4. The layer 2 source address of the message, if known, MAY be encoded in a Layer 2 Source TLV (Section 8.5). The source IP address of the message MUST be encoded in a IP Source Address TLV (Section 8.6). The source port of the message MUST be encoded in an IP Source port TLV (Section 8.7). The link on which the message was received MUST be encoded in a Link Identifier TLV (Section 8.8). The Discovery Proxy MUST silently ignore unrecognized TLVs in mDNS messages, and MUST NOT discard mDNS messages that include unrecognized TLVs.

A Discovery Proxy may discontinue listening for mDNS messages on a particular link by sending a session signaling message containing an MDNS Link Discontinue TLV (Section 8.9). Subsequent messages from that link that had previously been queued may arrive after listening has been discontinued. The Discovery Proxy should silently ignore such messages. The Discovery Relay MUST discontinue generating such messages as soon as the request is received. The Discovery Relay does not respond to this message other than to discontinue forwarding mDNS messages from the specified links.

6. Traffic from Proxies to Relays

Like mDNS traffic from relays, each mDNS message sent by a Discovery Proxy to a Discovery Relay is encapsulated in an MDNS Message TLV (Section 8.4) within a session signaling message. Each message MUST contain one or more Link Identifier TLVs (Section 8.8). The Discovery Relay will transmit the message to the mDNS port and multicast address on each link. The message MUST include one or more IP address family TLVs (Section 8.10). For each such TLVs that is included, the message will be sent on each link using the specified IP address family. If no address family codes are recognized, no packets will be transmitted.

7. Discovery Proxy Behavior

Discovery Proxies treat links for which Discovery Relay service is being used as if they were virtual interfaces; in other words, a Discovery Proxy serving multiple links using multiple Discovery Relays behaves the same as a Discovery Proxy serving multiple links using multiple physical network interfaces. In this section we refer to links served directly by the Discovery Proxy as locally-connected links, and links served through the Discovery Relay as relay-connected links.

What this means is that when a Discovery Proxy receives a DNSSD query, it will generate mDNS messages for each link for which it is acting as a proxy. For locally-connected links, those messages will be sent directly. For relay-connected links, the messages will be sent through the Discovery Relay that is being used to serve that lihnk.

Responses from devices on locally-connected links are processed normally. Responses from devices on relay-connected links are received by the Discovery Relay, encapsulated, and forwarded to the Discovery Proxy; the discovery proxy then processes these messages using the link-identifying information included in encapsulation.

Discovery Proxies do not respond to mDNS queries on relay-connected links. If an mDNS query is received from a Discovery Relay, the Discovery Proxy silently discards it. It is assumed that any such query will be repeated using DNS service discovery.

In principle it could be the case that some device is capable of performing service discovery using mDNS, but not using the DNS protocol. Responding to mDNS queries received from the Discovery Relay could address this use case. However, it is believed that no such devices exist, and therefore the preferred behavior is that all queries be resolved with unicast rather than multicast.

8. Session Signaling TLVs

This document defines a modest number of new DNS Session Signaling TLVs.

8.1. MDNS Link Request

The MDNS Link Request TLV conveys a 32-bit link identifier from which a Discovery Proxy is requesting that a Discovery Relay forward mDNS traffic. The link identifier comes from the provisioning configuration (see Section 9). The SSOP-TYPE for this TLV is TBD1.

The SSOP-LENGTH is always 4. The SSOP-DATA is the 32-bit identifier in network byte order.

8.2. MDNS Link Invalid

The MDNS Link Invalid TLV is returned in response to a session signaling message containing an MDNS Link Request, and returns the 32-bit identifier that was contained in that request. The link identifier comes from an MDNS Link Request TLV in the message being responded to. The TLV indicates that the specified link identifier does not refer to a valid link, either because the link is not supported by the Discovery Relay, or because the identifier is not known. The SSOP-TYPE for this TLV is TBD2. The SSOP-LENGTH is always 4. The SSOP-DATA is the 32-bit identifier in network byte order.

8.3. MDNS Link Subscribed

The MDNS Link Subscribed TLV is returned in response to a session signaling message containing an MDNS Link Request, and returns the 32-bit identifier from a MDNS Link Request TLV that was contained in that request. It indicates that MDNS messages for the specified link have been successfully subscribed. The SSOP-TYPE for this TLV is TBD3. The SSOP-LENGTH is always 4. The SSOP-DATA is the 32-bit identifier in network byte order.

8.4. MDNS Message

The MDNS Message TLV is used to encapsulate an mDNS message that is being forwarded from a link to a Discovery Proxy, or is being forwarded from a Discovery Proxy to a link. The SSOP-TYPE for this TLV is TBD4. SSOP-LENGTH is the length of the application layer payload of the MDNS message. SSOP-DATA is the application layer payload of the message.

8.5. Layer 2 Source Address

The Layer 2 Source Address TLV is used to report the link-layer address from which an mDNS message was received. This TLV is optionally present in session signaling messages from Discovery Relays to Discovery Proxies that contain mDNS messages when the source link-layer address is known. The SSOP-TYPE is TBD5. SSOP-LENGTH is variable, depending on the length of link-layer addresses on the link from which the message was received. SSOP-data is the link-layer address as it was received on the link.

8.6. IP Source Address

The IP Source Address TLV is used to report the IP source address from which an mDNS message was received. This TLV is present in session signaling messages from Discovery Relays to Discovery Proxies that contain mDNS messages. SSOP-TYPE is TBD6. SSOP-LENGTH is either 4, for an IPv4 address, or 16, for an IPv6 address. SSOP-DATA is the IP Address.

8.7. IP Source Port

The IP Source Port TLV is used to report the IP source port from which an mDNS message was received. This TLV is present in session signaling messages from Discovery Relays to Discovery Proxies. SSOP-TYPE is TBD7. SSOP-LENGTH is 2. SSOP-DATA is the source port in network byte order.

8.8. Link Identifier

This option is used both in session signaling messages from Discovery Proxies to Discovery Relays that contain mDNS messages, and in message from Discovery Relays to Discovery Proxies that contain mDNS messages. In the former case, it indicates a link to which the message should be forwarded; in the latter case, it indicates the link on which the message was received. SSOP-TYPE is TBD8. SSOP-LENGTH is 4. SSOP-DATA is a 32-bit link identifier as described in Section 9.

8.9. MDNS Discontinue

This option is used by Discovery Proxies to unsubscribe to mDNS messages on the specified link. More than one may be present in a single session signaling message. SSOP-TYPE is TBD9. SSOP-LENGTH is 4. SSOP-DATA is a 32-bit link identifier as described in Section 9.

8.10. IP Address Family

This option is used in mDNS messages sent by Discovery Proxies to links to indicate to the Discovery Relay which IP address family or families should be used when transmitting the message on the link. More than one may be present in a single session signaling message. SSOP-TYPE is TBD10. SSOP-LENGTH is 1. SSOP-DATA is a 8-bit IP family identifier. A value of 1 indicates IPv4. A value of 2 indicates IPv6. Other values are reserved, and MUST be ignored if not recognized.

9. Provisioning

In order for a Discovery Proxy to use Discovery Relays, it must be configured with sufficient information to identify links on which service discovery is to be supported and connect to discovery relays supporting those links, if it is not running on a host that is directly connected to those links.

A Discovery Relay must be configured both with a set of links to which the host on which it is running is connected, on which mDNS relay service is to be provided, and also with a list of one or more Discovery Proxies authorized to use it.

On a network supporting DNS Service Discovery using Discovery Relays, more than one different Discovery Relay implementation is likely be present. While it may be that only a single Discovery Proxy is present, that implementation will need to be able to be configured to interoperate with all of the Discovery Relays that are present. Consequently, it is necessary that a standard set of configuration parameters be defined for both Discovery Proxies and Discovery Relays.

DNS Service Discovery generally operates within a constrained set of links, not across the entire internet. This section assumes that what will be configured will be a limited set of links operated by a single entity or small set of cooperating entities, among which services present on each link should be available to users on that link and every other link. This could be, for example, a home network, a small office network, or even a network covering an entire building or small set of buildings. The set of Discovery Proxies and Discovery Relays within such a network will be referred to in this section as a 'Discovery Domain'.

Depending on the context, several different candidates for configuration of Discovery Proxies and Discovery relays may be applicable. The simplest such mechanism is a configuration file.

9.1. Provisioned Objects

Three types of objects must be described in order for Discovery Proxies and Discovery Relays to be provisioned: Discovery Proxies, Links, and Discovery Relays.

9.1.1. Discovery Proxy

The description of a Discovery Proxy consists of:

name an optional human-readable name which can appear in provisioning, monitoring and debugging systems. Must be unique within a Discovery Domain.

public-key a public key that identifies the Discovery Proxy. This key can be shared across services on the Discovery Proxy Host. The public key is used both to uniquely identify the Discovery Proxy and to authenticate connections from it.

private-key the private key corresponding to the public key.

source-ip-addresses a list of IP addresses that may be used by the Discovery Proxy when connecting to Discovery Relays. These addresses should be addresses that are configured on the Discovery Proxy Host. They should not be temporary addresses. All such addresses must be reachable within the Discovery Domain.

public-ip-addresses a list of IP addresses that may be used to submit DNS queries to the Discovery Proxy. This is not used for interoperation with Discovery Relays, but is mentioned here for completeness: this list of addresses may differ from the 'source-ip-addresses' list. If any of these addresses are reachable from outside of the Discovery Domain, services in that domain will be discoverable outside of the domain.

The private key should never be distributed to other hosts; all of the other information describing a Discovery Proxy can be safely shared with Discovery Relays.

9.1.2. Link

The description of a link (See [RFC8200] Section 2) consists of:

name A human-readable name for the link. This name MUST be unique within the Discovery Domain. Each link MUST have exactly one such name.

link-identifier An identifier that uniquely identifies that link within the Discovery Domain. Each link MUST have exactly one such identifier. This identifier is not expected to be meaningful to a human.

ldh-name An identifier for the link that is used to form an LDH domain name as described in [I-D.ietf-dnssd-hybrid], section 5.3. This is a single DNS label, not the entire domain name.

The 'name' and 'label-name' names can be used to form the LDH and human readable domain names as described in [I-D.ietf-dnssd-hybrid],

section 5.3. A single Discovery Domain is likely to have a single domain in which all links will be named, so to form the LDH (letters, digits, hyphens) FQDN for each link, the 'ldh-name' is prepended to the Discovery Domain's domain name. To form the human-readable FQDN, prepend 'name' to the Discovery Domain's domain name.

For example, if the Discovery Domain's domain name is 'example.com', 'name' is 'Building 2 South' and 'ldh-name' is 'bldg2s', then the LDH domain name for the link would be 'bldg2s.example.com' and the human-readable name would be 'Building 2 South.example.com'.

9.1.3. Discovery Relay

The description of a Discovery Relay consists of:

name an optional human-readable name which can appear in provisioning, monitoring and debugging systems. Must be unique within a Discovery Domain.

public-key a public key that identifies the Discovery Relay. This key can be shared across services on the Discovery Relay Host. Indeed, if a Discovery Proxy and Discovery Relay are running on the same host, the same key may be used for both. The public key uniquely identifies the Discovery Relay and is used by the Discovery Proxy to verify that it is talking to the intended Discovery Relay after a TLS connection has been established.

private-key the private key corresponding to the public key.

connect-tuples a list of IP address/port tuples that may be used to connect to the Discovery Relay. The relay may be configured to listen on all addresses on a single port, but this is not required, so the port as well as the address must be specified.

The private key should never be distributed to other hosts; all of the other information describing a Discovery Relay can be safely shared with Discovery Proxies.

10. Security Considerations

11. IANA Considerations

The IANA is kindly requested to update the DNS Session Signaling Type Codes Registry [I-D.ietf-dnsop-session-signal] by allocating codes for each of the TBD type codes listed in the following table, and by updating this document, here and in Section 8. Each type code should list this document as its reference document.

Opcode	Status	Name
TBD1	Standard	MDNS Link Request
TBD2	Standard	MDNS Link Invalid
TBD3	Standard	MDNS Link Subscribed
TBD4	Standard	MDNS Messsage
TBD5	Standard	Layer Two Source Address
TBD6	Standard	IP Source Address
TBD7	Standard	IP Destination Address
TBD8	Standard	Link Identifier
TBD9	Standard	MDNS Discontinue
TBD10	Standard	IP Address Family

DNS Session Signaling Type Codes to be allocated

12. Acknowledgments

13. References

13.1. Normative References

[I-D.ietf-dnsop-session-signal]
Bellis, R., Cheshire, S., Dickinson, J., Dickinson, S., Mankin, A., and T. Pusateri, "DNS Stateful Operations", draft-ietf-dnsop-session-signal-04 (work in progress), September 2017.

[I-D.ietf-dnssd-hybrid]
Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", draft-ietf-dnssd-hybrid-07 (work in progress), September 2017.

[I-D.ietf-tls-tls13]
Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", draft-ietf-tls-tls13-21 (work in progress), July 2017.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC1323] Jacobson, V., Braden, R., and D. Borman, "TCP Extensions for High Performance", RFC 1323, DOI 10.17487/RFC1323, May 1992, <https://www.rfc-editor.org/info/rfc1323>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <https://www.rfc-editor.org/info/rfc6762>.

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, DOI 10.17487/RFC6763, February 2013, <https://www.rfc-editor.org/info/rfc6763>.

[RFC7788] Stenberg, M., Barth, S., and P. Pfister, "Home Networking Control Protocol", RFC 7788, DOI 10.17487/RFC7788, April 2016, <https://www.rfc-editor.org/info/rfc7788>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

13.2. Informative References

[TR-069] Broadband Forum, "CPE WAN Management Protocol", November 2013, <https://www.broadband-forum.org/technical/download/TR-069_Amendment-5.pdf>.

Authors' Addresses

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

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

Ted Lemon
Barefoot Consulting
Brattleboro, Vermont 05301
United States of America

Email: mellon@fugue.com