

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 26, 2021

T. Lemon
S. Cheshire
Apple Inc.
February 22, 2021

Multicast DNS Discovery Relay
draft-ietf-dnssd-mdns-relay-04

Abstract

This document complements the specification of the Discovery Proxy for Multicast DNS-Based Service Discovery. It describes a lightweight relay mechanism, a Discovery Relay, which, when present on a link, allows remote clients, not attached to that link, to perform mDNS discovery operations on that link.

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 <https://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 August 26, 2021.

Copyright Notice

Copyright (c) 2021 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 (<https://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	4
3. Protocol Overview	6
3.1. Connections between Clients and Relays (overview)	6
3.2. mDNS Messages On Multicast Links	7
4. Connections between Clients and Relays (details)	8
5. Traffic from Relays to Clients	10
6. Traffic from Clients to Relays	12
7. Discovery Proxy Behavior	13
8. DSO TLVs	14
8.1. mDNS Link Data Request	14
8.2. mDNS Link Data Discontinue	14
8.3. Link Identifier	15
8.4. Encapsulated mDNS Message	15
8.5. IP Source	15
8.6. Link State Request	16
8.7. Link State Discontinue	16
8.8. Link Available	16
8.9. Link Unavailable	16
8.10. Link Prefix	17
9. Provisioning	18
9.1. Provisioned Objects	19
9.1.1. Multicast Link	20
9.1.2. Discovery Proxy	21
9.1.3. Discovery Relay	22
9.2. Configuration Files	23
9.3. Discovery Proxy Private Configuration	25
9.4. Discovery Relay Private Configuration	25
10. Security Considerations	26
11. IANA Considerations	27
12. Acknowledgments	27
13. References	28
13.1. Normative References	28
13.2. Informative References	29
Authors' Addresses	30

1. Introduction

This document defines a Discovery Relay. A Discovery Relay is a companion technology that works in conjunction with Discovery Proxies, and other clients.

The Discovery Proxy for Multicast DNS-Based Service Discovery [RFC8766] is a mechanism for discovering services on a subnetted network through the use of Discovery Proxies. Discovery Proxies issue Multicast DNS (mDNS) requests [RFC6762] on various multicast links in the network on behalf of a remote host performing DNS-Based Service Discovery [RFC6763].

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

The primary purpose of a Discovery Relay is providing remote virtual interface functionality to Discovery Proxies, and this document is written with that usage in mind. However, in principle, a Discovery Relay could be used by any properly authorized client. In the context of this specification, a Discovery Proxy is a client to the Discovery Relay. This document uses the terms "Discovery Proxy" and "Client" somewhat interchangeably; the term "Client" is used when we are talking about the communication between the Client and the Relay, and the term "Discovery Proxy" when we are referring specifically to a Discovery Relay Client that also happens to be a Discovery Proxy. One example of another kind of device that can be a client of a Discovery Relay is an Advertising Proxy [AdProx].

The Discovery Relay operates by listening for TCP connections from Clients. When a Client connects, the connection is authenticated and secured using TLS. The Client can then specify one or more multicast links from which it wishes to receive mDNS traffic. The Client can also send messages to be transmitted on its behalf on one or more of those multicast links. DNS Stateful Operations (DSO) [RFC8490] is used as a framework for conveying interface and IP header information associated with each message. DSO formats its messages using type-length-value (TLV) data structures. This document defines additional DSO TLV types, used to implement the Discovery Relay functionality.

The Discovery Relay functions essentially as a set of one or more remote virtual interfaces for the Client, one on each multicast 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 multicast link; in this case, the Client ideally should only be using one such Relay Proxy per multicast 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].

Use of a Discovery Relay can be considered similar to using Virtual LAN (VLAN) trunk ports to give a Discovery Proxy device a virtual presence on multiple links or broadcast domains. The difference is that while a VLAN trunk port operates at the link layer and delivers all link-layer traffic to the Discovery Proxy device, a Discovery Relay operates further up the network stack and selectively delivers only relevant Multicast DNS traffic. Also, VLAN trunk ports are generally only available within a single administrative domain and require link-layer configuration and connectivity, whereas the Discovery Relay protocol, which runs over TCP, can be used between any two devices with IP connectivity to each other.

2. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here. These words may also appear in this document in lower case as plain English words, absent their normative meanings.

The following definitions may be of use:

Client A network service that uses a Discovery Relay to send and receive mDNS multicast traffic on a remote link, to enable it to communicate with mDNS Agents on that remote link.

mDNS Agent A host which sends and/or responds to mDNS queries directly on its local link(s). Examples include network cameras, networked printers, networked home electronics, etc.

Discovery Proxy A network service which receives well-formed questions using the DNS protocol, performs multicast DNS queries to find answers to those questions, and responds with those answers using the DNS protocol. A Discovery Proxy that can communicate with remote mDNS Agents, using the services of a Discovery Relay, is a Client of the Discovery Relay.

Discovery Relay A network service which relays mDNS messages received on a local link to a Client, and on behalf of that Client can transmit mDNS messages on a local link.

multicast link A maximal set of network connection points, such that any host connected to any connection point in the set may send a packet with a link-local multicast destination address (specifically the mDNS link-local multicast destination address [RFC6762]) that will be received by all hosts connected to all other connection points in the set. Note that it is becoming increasingly common for a multicast link to be smaller than its corresponding unicast link. For example it is becoming common to have multiple Wi-Fi access points on a shared Ethernet backbone, where the multiple Wi-Fi access points and their shared Ethernet backbone form a single unicast link (a single IPv4 subnet, or single IPv6 prefix) but not a single multicast link. Unicast packets sent directly between two hosts on that IPv4 subnet or IPv6 prefix, without passing through an intervening IP-layer router, are correctly delivered, but multicast packets are not forwarded between the various Wi-Fi access points. Given the slowness of Wi-Fi multicast [I-D.ietf-mboned-ieee802-mcast-problems], having a packet that may be of interest to only one or two end systems transmitted to hundreds of devices, across multiple Wi-Fi access points, is especially wasteful. Hence the common configuration decision to not forward multicast packets between Wi-Fi access points is very reasonable. This further motivates the need for technologies like Discovery Proxy and Discovery Relay to facilitate discovery on these networks.

allow-list 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.

Take care when reading this document not to confuse the terms "Discovery Proxy" and "Discovery Relay". A Discovery Proxy [RFC8766] provides Multicast DNS discovery service to remote clients. A Discovery Relay is a simple software entity that provides virtual link connectivity to one or more Discovery Proxies or other Discovery Relay clients.

3. Protocol Overview

This document describes a way for a Client to communicate with mDNS agents on remote multicast links to which the client is not directly connected, using a Discovery Relay. As such, there are two parts to the protocol: connections between Clients and Discovery Relays, and communications between Discovery Relays and mDNS agents.

3.1. Connections between Clients and Relays (overview)

Discovery Relays listen for incoming connection requests. Connections between Clients and Discovery Relays are established by Clients. Connections are authenticated and encrypted using TLS, with both client and server certificates. Connections are long-lived: a Client is expected to send many queries over a single 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 DSO messages [RFC8490]. There will be four types of such messages between Discovery Relays and Clients:

- o Control messages from Client to Relay
- o Link status messages from Relay to Client
- o Encapsulated mDNS messages from Client to Relay
- o Encapsulated mDNS messages from Relay to Client

Clients can send four different control messages to Relays: Link State Request, Link State Discontinue, Link Data Request and Link Data Discontinue. The first two are used by the Client to request that the Relay report on the set of links that can be requested, and to request that it discontinue such reporting. The second two are used by the Client to indicate to the Discovery Relay that mDNS messages from one or more specified multicast links are to be relayed to the Client, and to subsequently stop such relaying.

Link Status messages from a Discovery Relay to the Client inform the Client that a link has become available, or that a formerly-available link is no longer available.

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

Encapsulated mDNS messages from a Client to a Discovery Relay cause the Discovery Relay to transmit the mDNS message on the specified multicast link to which the Discovery Relay host is directly attached.

During periods with no traffic flowing, Clients are responsible for generating any necessary keepalive traffic, as stated in the DSO specification [RFC8490].

3.2. mDNS Messages On Multicast Links

Discovery Relays listen for mDNS traffic on all configured multicast links that have at least one active subscription from a Client. When an mDNS message is received on a multicast link, it is forwarded on every open Client connection that is subscribed to mDNS traffic on that multicast link. In the event of congestion, where a particular Client 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. Excess mDNS packets are silently discarded. In practice this is not expected to be a issue. Particularly on networks like Wi-Fi, multicast packets are transmitted at rates ten or even a hundred times slower than unicast packets. This means that even at peak multicast packets rates, it is likely that a unicast TCP connection will be able to carry those packets with ease.

Clients send encapsulated mDNS messages they wish to have sent on their behalf on remote multicast link(s) on which the Client has an active subscription. A Discovery Relay will not transmit mDNS packets on any multicast link on which the Client does not have an active subscription, since it makes no sense for a Client to ask to have a query sent on its behalf if it's not able to receive the responses to that query.

4. Connections between Clients and Relays (details)

When a Discovery Relay starts, it opens a passive TCP listener to receive incoming connection requests from Clients. This listener may be bound to one or more source IP addresses, or to the wildcard address, depending on the 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.

If the relay is using an IP address allow-list, the next step is for the relay to verify that the source IP address of the connection is on its allow-list. If the connection is not permitted either because of the source address or the destination address, the Discovery Relay closes the connection. If possible, before closing the connection, the Discovery Relay first sends a TLS `user_canceled` alert ([RFC8446] Section 6.1). Discovery Relays SHOULD refuse to accept TCP connections to invalid destination addresses, rather than accepting and then closing the connection, if this is possible.

Otherwise, the Discovery Relay will attempt to complete a TLS handshake with the Client. Clients are required to send the `post_handshake_auth` extension ([RFC8446] Section 4.2.5). If a Discovery Relay receives a `ClientHello` message with no `post_handshake_auth` extension, the Discovery Relay rejects the connection with a `certificate_required` alert ([RFC8446] Section 6.2).

Once the TLS handshake is complete, the Discovery Relay MUST request post-handshake authentication ([RFC8446] Section 4.6.2). If the Client 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 ([RFC8446] Section 6.2).

Clients MUST validate server certificates. If the client is configured with a server IP address and certificate, it can validate the server by comparing the certificate offered by the server to the certificate that was provided: they should be the same. If the certificate includes a Distinguished Name that is a fully-qualified domain name, the client SHOULD present that domain name to the server in an SNI request.

Rather than being configured with an IP address and a certificate, the client may be configured with the server's FQDN. In this case, the client uses the server's FQDN as a Authentication Domain Name

[RFC8310] Section 7.1, and uses the authentication method described in [RFC8310] section 8.1, if the certificate is signed by a root authority the client trusts, or the method described in section 8.2 of the same document if not. If neither method is available, then a locally-configured copy of the server certificate can be used, as in the previous paragraph.

Once the connection is established and authenticated, it is treated as a DNS TCP connection [RFC7766].

Aliveness of connections between Clients and Relays is maintained as described in Section 4 of the DSO specification [RFC8490]. Clients must also honor the 'Retry Delay' TLV (section 5 of [RFC8490]) if sent by the Discovery Relay.

Clients SHOULD avoid establishing more than one connection to a specific Discovery Relay. However, there may be situations where multiple connections to the same Discovery Relay are unavoidable, so Discovery Relays MUST be willing to accept multiple connections from the same Client.

In order to know what links to request, the Client can be configured with a list of links supported by the Relay. However, in some networking contexts, dynamic changes in the availability of links are likely; therefore Clients may also use the Report Link Changes TLV to request that the Relay report on the availability of its links. In some contexts, for example when debugging, a Client may operate with no information about the set of links supported by a relay, simply relying on the relay to provide one.

5. Traffic from Relays to Clients

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 multicast link be forwarded on a particular connection, the Client must send one or more DSO messages, each containing a single mDNS Link Data Request TLV (Section 8.1) indicating the multicast link from which traffic is requested.

When an mDNS Link Data Request message is received, the Discovery Relay validates that it recognizes the link identifier, and that forwarding is enabled for that link. If both checks are successful, it MUST send a response with RCODE=0 (NOERROR). If the link identifier is not recognized, it sends a response with RCODE=3 (NXDOMAIN/Name Error). If forwarding from that link to the Client is not enabled, it sends a response with RCODE=5 (REFUSED). If the relay cannot satisfy the request for some other reason, for example resource exhaustion, it sends a response with RCODE=2 (SERVFAIL).

If the requested link is valid, the Relay begins forwarding all mDNS messages from that link to the Client. 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 [RFC7323]. The Discovery Relay should use the TCP_NOTSENT_LOWAT mechanism [NOTSENT][PRIO] or equivalent, to avoid building up a backlog of data in excess of the amount necessary to have in flight to fill the bandwidth delay product of the TCP connection.

Encapsulated mDNS messages from Relays to Clients are framed within DSO messages. Each DSO message can contain multiple TLVs, but only a single encapsulated mDNS message is conveyed per DSO message. Each forwarded mDNS message is sent in an Encapsulated mDNS Message TLV (Section 8.4). The source IP address and port of the message MUST be encoded in an IP Source TLV (Section 8.5). The multicast link on which the message was received MUST be encoded in a Link Identifier TLV (Section 8.3). As described in the DSO specification [RFC8490], a Client MUST silently ignore unrecognized Additional TLVs in mDNS messages, and MUST NOT discard mDNS messages that include unrecognized Additional TLVs.

A Client may discontinue listening for mDNS messages on a particular multicast link by sending a DSO message containing an mDNS Link Data Discontinue TLV (Section 8.2). The Discovery Relay MUST discontinue forwarding mDNS messages when the Link Data Discontinue request is received. However, messages from that link that had previously been

queued may arrive after the Client has discontinued its listening. The Client should silently discard such messages. The Discovery Relay does not respond to the Link Data Discontinue message other than to discontinue forwarding mDNS messages from the specified links.

6. Traffic from Clients to Relays

Like mDNS traffic from relays, each mDNS message sent by a Client to a Discovery Relay is communicated in an Encapsulated mDNS Message TLV (Section 8.4) within a DSO message. Each message MUST contain exactly one Link Identifier TLV (Section 8.3). The Discovery Relay will transmit the mDNS message to the mDNS port and multicast address on the link specified in the message using the specified IP address family.

Although the communication between Clients and Relays uses the DNS stream protocol and DNS Stateless Operations, there is no case in which a Client would legitimately send a DNS query (or anything else other than a DSO message) to a Relay. Therefore, if a Relay receives any message other than a DSO message, it MUST immediately abort that DSO session with a TCP reset (RST).

When defining this behavior, the working group considered making it possible to specify more than one link identifier in an mDNSMessage TLV. A superficial evaluation of this suggested that this might be a useful optimization, since when a query is issued, it will often be issued to all links. However, on many link types, like Wi-Fi, multicast traffic is expensive [I-D.ietf-mboned-ieee802-mcast-problems] and should be generated frugally, so providing convenient ways to generate additional multicast traffic was determined to be an unwise optimization. In addition, because of the way mDNS handles retries, it will almost never be the case that the exact same message will be sent on more than one link. Therefore, the complexity that this optimization adds is not justified by the potential benefit, and this idea has been abandoned.

7. Discovery Proxy Behavior

Discovery Proxies treat multicast links for which Discovery Relay service is being used as if they were virtual interfaces; in other words, a Discovery Proxy serving multiple remote multicast links using multiple remote Discovery Relays behaves the same as a Discovery Proxy serving multiple local multicast links using multiple local physical network interfaces. In this section we refer to multicast links served directly by the Discovery Proxy as locally-connected links, and multicast links served through the Discovery Relay as relay-connected links. A relay-connected link can be thought of as similar to a link that a Discovery Proxy connects to using a USB Ethernet interface, just with a very long USB cable (that runs over TCP).

When a Discovery Proxy receives a DNS query from a DNS client via unicast, it will generate corresponding mDNS query messages on the relevant multicast link(s) for which it is acting as a proxy. For locally-connected link(s), those query messages will be sent directly. For relay-connected link(s), the query messages will be sent through the Discovery Relay that is being used to serve that multicast link.

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 Client; the Client then processes these messages using the link-identifying information included in the encapsulation.

In principle it could be the case that some device is capable of performing service discovery using Multicast DNS, but not using traditional unicast DNS. Responding to mDNS queries received from the Discovery Relay could address this use case. However, continued reliance on multicast is counter to the goals of the current work in service discovery, and to benefit from wide-area service discovery such client devices should be updated to support service discovery using unicast queries.

8. DSO TLVs

This document defines a modest number of new DSO TLVs.

8.1. mDNS Link Data Request

The mDNS Link Data Request TLV conveys a link identifier from which a Client is requesting that a Discovery Relay forward mDNS traffic. The link identifier comes from the provisioning configuration (see Section 9). The DSO-TYPE for this TLV is TBD-R. DSO-LENGTH is always 5. DSO-DATA is the 8-bit address family followed by the link identifier, a 32-bit unsigned integer in network (big endian) byte order, as described in Section 9. An address family value of 1 indicates IPv4 and 2 indicates IPv6, as recorded in the IANA Registry of Address Family Numbers [AdFam].

The mDNS Link Data Request TLV can only be used as a primary TLV, and requires an acknowledgement.

At most one mDNS Link Data Request TLV may appear in a DSO message. To request multiple link subscriptions, multiple separate DSO messages are sent, each containing a single mDNS Link Data Request TLV.

A Client MUST NOT request a link if it already has an active subscription to that link on the same DSO connection. If a Discovery Relay receives a duplicate link subscription request, it MUST immediately abort that DSO session with a TCP reset (RST).

8.2. mDNS Link Data Discontinue

The mDNS Link Data Discontinue TLV is used by Clients to unsubscribe to mDNS messages on the specified multicast link. DSO-TYPE is TBD-D. DSO-LENGTH is always 5. DSO-DATA is the 8-bit address family followed by the 32-bit link identifier, a 32-bit unsigned integer in network (big endian) byte order, as described in Section 9.

The mDNS Link Data Discontinue TLV can only be used as a DSO unidirectional message TLV, and is not acknowledged.

At most one mDNS Link Data Discontinue TLV may appear in a DSO message. To unsubscribe from multiple links, multiple separate DSO messages are sent, each containing a single mDNS Link Data Discontinue TLV.

8.3. Link Identifier

This option is used both in DSO messages from Discovery Relays to Clients that contain received mDNS messages, and from Clients to Discovery Relays that contain mDNS messages to be transmitted on the multicast link. In the former case, it indicates the multicast link on which the message was received; in the latter case, it indicates the multicast link on which the message should be transmitted. DSO-TYPE is TBD-L. DSO-LENGTH is always 5. DSO-DATA is the 8-bit address family followed by the link identifier, a 32-bit unsigned integer in network (big endian) byte order, as described in Section 9.

The Link Identifier TLV can only be used as an additional TLV. The Link Identifier TLV can only appear at most once in a Discovery Relay DSO message.

8.4. Encapsulated mDNS Message

The Encapsulated mDNS Message TLV is used to communicate an mDNS message that a Relay is forwarding from a multicast link to a Client, or that a Client is sending to a Relay for transmission on a multicast link. Only the application-layer payload of the mDNS message is carried in the DSO "Encapsulated mDNS Message" TLV, i.e., just the DNS message itself, beginning with the DNS Message ID, not the IP or UDP headers. The DSO-TYPE for this TLV is TBD-M. DSO-LENGTH is the length of the encapsulated mDNS message. DSO-DATA is the content of the encapsulated mDNS message.

The Encapsulated mDNS Message TLV can only be used as a DSO unidirectional message TLV, and is not acknowledged.

8.5. IP Source

The IP Source TLV is used to report the IP source address and port from which an mDNS message was received. This TLV is present in DSO messages from Discovery Relays to Clients that contain encapsulated mDNS messages. DSO-TYPE is TBD-S. DSO-LENGTH is either 6, for an IPv4 address, or 18, for an IPv6 address. DSO-DATA is the two-byte source port, followed by the 4- or 16-byte IP Address. Both port and address are in the canonical byte order (i.e., the same representation as used in the UDP and IP packet headers, with no byte swapping).

The IP Source TLV can only be used as an additional TLV. The IP Source TLV can only appear at most once in a Discovery Relay DSO message.

8.6. Link State Request

The Link State Request TLV requests that the Discovery Relay report link changes. When the relay is reporting link changes and a new link becomes available, it sends a Link Available message to the Client. When a link becomes unavailable, it sends a Link Unavailable message to the Client. If there are links available when the request is received, then for each such link the relay immediately sends a Link Available Message to the Client. DSO-TYPE is TBD-P. DSO-LENGTH is 0.

The mDNS Link State Request TLV can only be used as a primary TLV, and requires an acknowledgement. The acknowledgment does not contain a Link Available TLV: it is just a response to the Link State Request message.

8.7. Link State Discontinue

The Link State Discontinue TLV requests that the Discovery Relay stop reporting on the availability of links supported by the relay. This cancels the effect of a Link State Request TLV. DSO-TYPE is TBD-Q. DSO-LENGTH is 0.

The mDNS Link State Discontinue TLV can only be used as a DSO unidirectional message TLV, and is not acknowledged.

8.8. Link Available

The Link Available TLV is used by Discovery Relays to indicate to Clients that a new link has become available. The format is the same as the Link Identifier TLV. DSO-TYPE is TBD-V. The Link Available TLV may be accompanied by one or more Link Prefix TLVs which indicate IP prefixes the Relay knows to be present on the link.

The mDNS Link Available TLV can only be used as a DSO unidirectional message TLV, and is not acknowledged.

8.9. Link Unavailable

The Link Unavailable TLV is used by Discovery Relays to indicate to Clients that an existing link has become unavailable. The format is the same as the Link Identifier TLV. DSO-TYPE is TBD-U.

The mDNS Link Unavailable TLV can only be used as a DSO unidirectional message TLV, and is not acknowledged.

8.10. Link Prefix

The Link Prefix TLV represents an IP address or prefix configured on a link. The length is 17 for an IPv6 address or prefix, and 5 for an IPv4 address or prefix. The TLV consists of a prefix length, between 0 and 32 for IPv4 or between 0 and 128 for IPv6, represented as a single byte. This is followed by the IP address, either four or sixteen bytes. DSO-TYPE is TBD-K.

The Link Prefix TLV can only be used as a secondary TLV.

9. Provisioning

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

A Discovery Relay must be configured both with a set of multicast 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 Clients authorized to use it.

On a network supporting DNS Service Discovery using Discovery Relays, more than one different Discovery Relay implementation may 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 manual configuration file, but regardless of provisioning mechanism, certain configuration information needs to be communicated to the devices, as outlined below.

In the example we provide here, we only refer to configuring of IP addresses, private keys and certificates. It is also possible to use FQDNs to identify servers; this then allows for the use of DANE ([RFC8310] Section 8.2) or PKIX authentication [RFC6125]. Which method is used is to some extent up to the implementation, but at a minimum, it should be possible to associate an IP address with a self-signed certificate, and it should be possible to validate both

self-signed and PKIX-authenticated certificates, with PKIX, DANE or a pre-configured trust anchor.

9.1. Provisioned Objects

Three types of objects must be described in order for Discovery Proxies and Discovery Relays to be provisioned: Discovery Proxies, Multicast Links, and Discovery Relays. "Human-readable" below means actual words or proper names that will make sense to an untrained human being. "Machine-readable" means a name that will be used by machines to identify the entity to which the name refers. Each entity must have a machine-readable name and may have a human-readable name. No two entities can have the same human-readable name. Similarly, no two entities can have the same machine-readable name.

9.1.1. Multicast Link

The description of a multicast link consists of:

link-identifier A 32-bit identifier that uniquely identifies that link within the Discovery Domain. Each link **MUST** have exactly one such identifier. Link Identifiers do not have any special semantics, and are not intended to be human-readable.

ldh-name A fully-qualified domain name for the multicast link that is used to form an LDH domain name as described in section 5.3 of the Discovery Proxy specification [RFC8766]. This name is used to identify the link during provisioning, and must be present.

hr-name A human-readable user-friendly fully-qualified domain name for the multicast link. This name **MUST** be unique within the Discovery Domain. Each multicast link **MUST** have exactly one such name. The hr-name **MAY** be the same as the ldh-name. (The hr-name is allowed to contain spaces, punctuation and rich text, but it is not required to do so.)

The ldh-name and hr-name can be used to form the LDH and human-readable domain names as described in [RFC8766], section 5.3.

Note that the ldh-name and hr-name can be used in two different ways.

On a small home network with little or no human administrative configuration, link names may be directly visible to the user. For example, a search in 'home.arpa' on a small home network may discover services on both ethernet.home.arpa and wi-fi.home.arpa. In the case of a home user who has one Ethernet-connected printer and one Wi-Fi-connected printer, discovering that they have one printer on ethernet.home.arpa and another on wi-fi.home.arpa is understandable and meaningful.

On a large corporate network with hundreds of Wi-Fi access points, the individual link names of the hundreds of multicast links are less likely to be useful to end users. In these cases, Discovery Broker functionality [I-D.sctl-discovery-broker] may be used to translate the many link names to something more meaningful to users. For example, in a building with 50 Wi-Fi access points, each with their own link names, services on all the different physical links may be presented to the user as appearing in 'headquarters.example.com'. In this case, the individual link names can be thought of similar to MAC addresses or IPv6 addresses. They are used internally by the software as unique identifiers, but generally are not exposed to end users.

9.1.2. Discovery Proxy

The description of a Discovery Proxy consists of:

name a machine-readable name used to reference this Discovery Proxy in provisioning.

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

certificate a certificate that identifies the Discovery Proxy. This certificate can be shared across services on the Discovery Proxy Host. The public key in the certificate is used both to uniquely identify the Discovery Proxy and to authenticate connections from it. The certificate should be signed by its own private key.

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

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 a Discovery Proxy listens on to receive requests from clients. This is not used for interoperation with Discovery Relays, but is mentioned here for completeness: the list of addresses listened on for incoming client requests may differ from the 'source-ip-addresses' list of addresses used for issuing outbound connection requests to Discovery Relays. If any of these addresses are reachable from outside of the Discovery Domain, services in that domain will be discoverable outside of the domain.

multicast links a list of multicast links on which this Discovery Proxy is expected to provide service

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.

In some configurations it may make sense for the Discovery Relay not to have a list of links, but simply to support the set of all links available on relays to which the Discovery Proxy is configured to communicate.

9.1.3. Discovery Relay

The description of a Discovery Relay consists of:

`name` a required machine-readable identifier used to reference the relay

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

`certificate` a certificate that identifies the Discovery Relay. This certificate 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 certificate can be used for both. The public key in the certificate uniquely identifies the Discovery Relay and is used by a Discovery Relay Client (e.g., a Discovery Proxy) to verify that it is talking to the intended Discovery Relay after a TLS connection has been established. The certificate must either be signed by its own key, or have a signature chain that can be validated using PKIX authentication [RFC6125].

`private-key` the private key corresponding to the public key in the certificate.

`listen-tuple` 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.

`multicast links` a list of multicast links to which this relay is physically connected.

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.

In some cases a Relay may not be configured with a static list of links, but may simply discover links by monitoring the set of available interfaces on the host on which the Relay is running. In that case, the relay could be configured to identify links based on the names of network interfaces, or based on the set of available prefixes seen on those interfaces. The details of this sort of configuration are not specified in this document.

9.2. Configuration Files

For this discussion, we assume the simplest possible means of configuring Discovery Proxies and Discovery Relays: the configuration file. Any environment where changes will happen on a regular basis will either require some automatic means of generating these configuration files as the network topology changes, or will need to use a more automatic method for configuration, such as HNCP [RFC7788].

There are many different ways to organize configuration files. This discussion assumes that multicast links, relays and proxies will be specified as objects, as described above, perhaps in a master file, and then the specific configuration of each proxy or relay will reference the set of objects in the master file, referencing objects by name. This approach is not required, but is simply shown as an example. In addition, the private keys for each proxy or relay must appear only in that proxy or relay's configuration file.

The master file contains a list of Discovery Relays, Discovery Proxies and Multicast Links. Each object has a name and all the other data associated with it. We do not formally specify the format of the file, but it might look something like this:

```
Relay upstairs
  certificate xxx
  listen-tuple 192.0.2.1 1917
  listen-tuple fd00::1 1917
  link upstairs-wifi
  link upstairs-wired
  client-allow-list main

Relay downstairs
  certificate yyy
  listen-tuple 192.51.100.1 2088
  listen-tuple fd00::2 2088
  link downstairs-wifi
  link downstairs-wired
  client-allow-list main

Proxy main
  certificate zzz
  address 203.1.113.1

Link upstairs-wifi
  id 1
  hr-name Upstairs Wifi

Link upstairs-wired
  id 2
  hr-name Upstairs Wired

Link downstairs-wifi
  id 3
  hr-name Downstairs Wifi

Link downstairs-wired
  id 4
  hr-name Downstairs Wired
```

9.3. Discovery Proxy Private Configuration

The Discovery Proxy configuration contains enough information to identify which Discovery Proxy is being configured, enumerate the list of multicast links it is intended to serve, and provide keying information it can use to authenticate to Discovery Relays. It may also contain custom information about the port and/or IP address(es) on which it will respond to DNS queries.

An example configuration, following the convention used in this section, might look something like this:

```
Proxy main
  private-key zzz
  subscribe upstairs-wifi
  subscribe downstairs-wifi
  subscribe upstairs-wired
  subscribe downstairs-wired
```

When combined with the master file, this configuration is sufficient for the Discovery Proxy to identify and connect to the Discovery Relays that serve the links it is configured to support.

9.4. Discovery Relay Private Configuration

The Discovery Relay configuration just needs to tell the Discovery Relay what name to use to find its configuration in the master file, and what the private key is corresponding to its certificate (public key) in the master file. For example:

```
Relay Downstairs
  private-key yyy
```

10. Security Considerations

Part of the purpose of the Multicast DNS Discovery Relay protocol is to place a simple relay, analogous to a BOOTP relay, into routers and similar devices that may not be updated frequently. The BOOTP [RFC0951] protocol has been around since 1985, and continues to be useful today. The BOOTP protocol uses no encryption, and in many enterprise networks this is considered acceptable. In contrast, the Discovery Relay protocol requires TLS 1.3. A concern is that after 20 or 30 years, TLS 1.3, or some of the encryption algorithms it uses, may become obsolete, rendering devices that require it unusable. Our assessment is that TLS 1.3 probably will be around for many years to come. TLS 1.0 [RFC2246] was used for about a decade, and similarly TLS 1.2 [RFC5246] was also used for about a decade. We expect TLS 1.3 [RFC8446] to have at least that lifespan. In addition, recent IETF efforts are pushing for better software update practices for devices like routers, for other security reasons, making it likely that in ten years time it will be less common to be using routers that haven't had a software update for ten years. However, authors of encryption specifications and libraries should be aware of the potential backwards compatibility issues if an encryption algorithm becomes deprecated. This specification RECOMMENDS that if an encryption algorithm becomes deprecated, then rather than remove that encryption algorithm entirely, encryption libraries should disable that encryption algorithm by default, but leave the code present with an option for client software to enable it in special cases, such as a recent Client talking to an ancient Discovery Relay. Using no encryption, like BOOTP, would eliminate this backwards compatibility concern, but we feel that in such a future hypothetical scenario, using even a weak encryption algorithm still makes passive eavesdropping and tampering harder, and is preferable to using no encryption at all.

11. IANA Considerations

The IANA is kindly requested to update the DSO Type Codes Registry [RFC8490] 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.

DSO-TYPE	Status	Name
TBD-R	Standard	Link Data Request
TBD-D	Standard	Link Data Discontinue
TBD-L	Standard	Link Identifier
TBD-M	Standard	Encapsulated mDNS Message
TBD-S	Standard	IP Source
TBD-P	Standard	Link State Request
TBD-Q	Standard	Link State Discontinue
TBD-V	Standard	Link Available
TBD-U	Standard	Link Unavailable
TBD-K	Standard	Link Prefix

DSO Type Codes to be allocated

12. Acknowledgments

Thanks to Derek Atkins for the secdir early review.

13. References

13.1. Normative References

- [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>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <<https://www.rfc-editor.org/info/rfc6125>>.
- [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>>.
- [RFC7323] Borman, D., Braden, B., Jacobson, V., and R. Scheffenegger, Ed., "TCP Extensions for High Performance", RFC 7323, DOI 10.17487/RFC7323, September 2014, <<https://www.rfc-editor.org/info/rfc7323>>.
- [RFC7766] Dickinson, J., Dickinson, S., Bellis, R., Mankin, A., and D. Wessels, "DNS Transport over TCP - Implementation Requirements", RFC 7766, DOI 10.17487/RFC7766, March 2016, <<https://www.rfc-editor.org/info/rfc7766>>.
- [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>>.

- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <<https://www.rfc-editor.org/info/rfc8310>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [RFC8490] Bellis, R., Cheshire, S., Dickinson, J., Dickinson, S., Lemon, T., and T. Pusateri, "DNS Stateful Operations", RFC 8490, DOI 10.17487/RFC8490, March 2019, <<https://www.rfc-editor.org/info/rfc8490>>.
- [RFC8766] Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", RFC 8766, DOI 10.17487/RFC8766, June 2020, <<https://www.rfc-editor.org/info/rfc8766>>.

13.2. Informative References

- [AdFam] "IANA Address Family Numbers Registry", <<https://www.iana.org/assignments/address-family-numbers/>>.
- [AdProx] Cheshire, S. and T. Lemon, "Advertising Proxy for DNS-SD Service Registration Protocol", draft-sctl-advertising-proxy-00 (work in progress), July 2020.
- [I-D.ietf-mboned-ieee802-mcast-problems] Perkins, C., McBride, M., Stanley, D., Kumari, W., and J. Zuniga, "Multicast Considerations over IEEE 802 Wireless Media", draft-ietf-mboned-ieee802-mcast-problems-12 (work in progress), October 2020.
- [I-D.sctl-discovery-broker] Cheshire, S. and T. Lemon, "Service Discovery Broker", draft-sctl-discovery-broker-00 (work in progress), July 2017.
- [NOTSENT] Dumazet, E., "TCP_NOTSENT_LOWAT socket option", July 2013, <<https://lwn.net/Articles/560082/>>.

- [PRIO] Chan, W., "Prioritization Only Works When There's Pending Data to Prioritize", January 2014, <<https://insouciant.org/tech/prioritization-only-works-when-theres-pending-data-to-prioritize/>>.
- [RFC0951] Croft, W. and J. Gilmore, "Bootstrap Protocol", RFC 951, DOI 10.17487/RFC0951, September 1985, <<https://www.rfc-editor.org/info/rfc951>>.
- [RFC2246] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", RFC 2246, DOI 10.17487/RFC2246, January 1999, <<https://www.rfc-editor.org/info/rfc2246>>.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <<https://www.rfc-editor.org/info/rfc5246>>.
- [TR-069] Broadband Forum, "CPE WAN Management Protocol", November 2013, <https://www.broadband-forum.org/technical/download/TR-069_Amendment-5.pdf>.

Authors' Addresses

Ted Lemon
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America

Phone: +1 (408) 996-1010
Email: elemen@apple.com

Stuart Cheshire
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America

Phone: +1 (408) 996-1010
Email: cheshire@apple.com

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: 15 July 2021

T. Lemon
S. Cheshire
Apple Inc.
11 January 2021

Service Registration Protocol for DNS-Based Service Discovery
draft-ietf-dnssd-srp-09

Abstract

The Service Registration Protocol for DNS-Based Service Discovery uses the standard DNS Update mechanism to enable DNS-Based Service Discovery using only unicast packets. This makes it possible to deploy DNS Service Discovery without multicast, which greatly improves scalability and improves performance on networks where multicast service is not an optimal choice, particularly 802.11 (Wi-Fi) and 802.15.4 (IoT) networks. DNS-SD Service registration uses public keys and SIG(0) to allow services to defend their registrations against attack.

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 <https://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 15 July 2021.

Copyright Notice

Copyright (c) 2021 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 (<https://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. Service Registration Protocol	5
2.1. Protocol Variants	5
2.1.1. Full-featured Hosts	5
2.1.2. Constrained Hosts	6
2.1.3. Why two variants?	6
2.2. Protocol Details	6
2.2.1. What to publish	7
2.2.2. Where to publish it	7
2.2.3. How to publish it	8
2.2.4. How to secure it	9
2.2.5. Service Behavior	9
2.3. SRP Server Behavior	12
2.3.1. Validation of Adds and Deletes	12
2.3.2. Valid SRP Update Requirements	14
2.3.3. FCFS Name And Signature Validation	15
2.3.4. SRP Update response	16
2.3.5. Optional Behavior	16
3. TTL Consistency	17
4. Maintenance	17
4.1. Cleaning up stale data	17
5. Sleep Proxy	19
6. Security Considerations	20
6.1. Source Validation	20
6.2. SRP Server Authentication	21
6.3. Required Signature Algorithm	21
7. Privacy Considerations	21
8. Delegation of 'service.arpa.'	21
9. IANA Considerations	22
9.1. Registration and Delegation of 'service.arpa' as a Special-Use Domain Name	22
9.2. 'dnssd-srp' Service Name	22
9.3. 'dnssd-srp-tls' Service Name	22
9.4. Anycast Address	23
10. Implementation Status	23
11. Acknowledgments	24
12. Normative References	24
13. Informative References	26
Appendix A. Testing using standard RFC2136-compliant servers . .	27
Appendix B. How to allow services to update standard RFC2136-compliant servers	28

Appendix C. Sample BIND9 configuration for	
default.service.arpa.	28
Authors' Addresses	29

1. Introduction

DNS-Based Service Discovery [RFC6763] is a component of Zero Configuration Networking [RFC6760] [ZC] [I-D.cheshire-dnssd-roadmap].

This document describes an enhancement to DNS-Based Service Discovery [RFC6763] that allows services to register their services using the DNS protocol rather than using Multicast DNS [RFC6762] (mDNS). There is already a large installed base of DNS-SD clients that can discover services using the DNS protocol.

This document is intended for three audiences: implementors of software that provides services that should be advertised using DNS-SD, implementors of DNS servers that will be used in contexts where DNS-SD registration is needed, and administrators of networks where DNS-SD service is required. The document is intended to provide sufficient information to allow interoperable implementation of the registration protocol.

DNS-Based Service Discovery (DNS-SD) allows services to advertise the fact that they provide service, and to provide the information required to access that service. DNS-SD clients can then discover the set of services of a particular type that are available. They can then select a service from among those that are available and obtain the information required to use it. Although DNS-SD using the DNS protocol (as opposed to mDNS) can be more efficient and versatile, it is not common in practice, because of the difficulties associated with updating authoritative DNS services with service information.

Existing practice for updating DNS zones is to either manually enter new data, or else use DNS Update [RFC2136]. Unfortunately DNS Update requires either that the authoritative DNS server automatically trust updates, or else that the DNS Update client have some kind of shared secret or public key that is known to the DNS server and can be used to authenticate the update. Furthermore, DNS Update can be a fairly chatty process, requiring multiple round trips with different conditional predicates to complete the update process.

The SRP protocol adds a set of default heuristics for processing DNS updates that eliminates the need for DNS update conditional predicates: instead, the SRP server has a set of default predicates that are applied to the update, and the update either succeeds entirely, or fails in a way that allows the registering service to know what went wrong and construct a new update.

SRP also adds a feature called First-Come, First-Served Naming, which allows the registering service to claim a name that is not yet in use, and, using SIG(0) [RFC2931], to authenticate both the initial claim and subsequent updates. This prevents name conflicts, since a second SRP service attempting to claim the same name will not possess the SIG(0) key used by the first service to claim it, and so its claim will be rejected and the second service will have to choose a new name.

Finally, SRP adds the concept of a 'lease,' similar to leases in Dynamic Host Configuration Protocol [RFC8415]. The SRP registration itself has a lease which may be on the order of an hour; if the registering service does not renew the lease before it has elapsed, the registration is removed. The claim on the name can have a longer lease, so that another service cannot claim the name, even though the registration has expired.

The Service Registration Protocol for DNS-SD (SRP), described in this document, provides a reasonably secure mechanism for publishing this information. Once published, these services can be readily discovered by DNS-SD clients using standard DNS lookups.

The DNS-SD specification [RFC6763], Section 10 ("Populating the DNS with Information"), briefly discusses ways that services can publish their information in the DNS namespace. In the case of mDNS, it allows services to publish their information on the local link, using names in the ".local" namespace, which makes their services directly discoverable by peers attached to that same local link.

RFC6763 also allows clients to discover services using the DNS protocol [RFC1035]. This can be done by having a system administrator manually configure service information in the DNS, but manually populating DNS authoritative server databases is costly and potentially error-prone, and requires a knowledgeable network administrator. Consequently, although all DNS-SD client implementations of which we are aware support DNS-SD using DNS queries, in practice it is used much less frequently than mDNS.

The Discovery Proxy [RFC8766] provides one way to automatically populate the DNS namespace, but is only appropriate on networks where services are easily advertised using mDNS. This document describes a

solution more suitable for networks where multicast is inefficient, or where sleepy devices are common, by supporting both offering of services, and discovery of services, using unicast.

2. Service Registration Protocol

Services that implement SRP use DNS Update [RFC2136] [RFC3007] to publish service information in the DNS. Two variants exist, one for full-featured hosts, and one for devices designed for "Constrained-Node Networks" [RFC7228]. An SRP server is most likely an authoritative DNS server, or else is updating an authoritative DNS server. There is no requirement that the server that is receiving SRP requests be the same server that is answering queries that return records that have been registered.

2.1. Protocol Variants

2.1.1. Full-featured Hosts

Full-featured hosts are either configured manually with a registration domain, or use the "dr._dns-sd._udp.<domain>" query ([RFC6763], Section 11) to learn the default registration domain from the network. RFC6763 says to discover the registration domain using either ".local" or a network-supplied domain name for <domain>. Services using SRP MUST use the domain name received through the DHCPv4 Domain Name option ([RFC2132], Section 3.17), if available, or the Neighbor Discovery DNS Search List option [RFC8106]. If the DNS Search List option contains more than one domain name, it MUST NOT be used. If neither option is available, the Service Registration protocol is not available on the local network.

Manual configuration of the registration domain can be done either by querying the list of available registration zones ("r._dns-sd._udp") and allowing the user to select one from the UI, or by any other means appropriate to the particular use case being addressed. Full-featured devices construct the names of the SRV, TXT, and PTR records describing their service(s) as subdomains of the chosen service registration domain. For these names they then discover the zone apex of the closest enclosing DNS zone using SOA queries [RFC8765]. Having discovered the enclosing DNS zone, they query for the "_dnssd-srp._tcp.<zone>" SRV record to discover the server to which they should send DNS updates. Hosts that support SRP updates using TLS use the "_dnssd-srp-tls._tcp.<zone>" SRV record instead.

2.1.2. Constrained Hosts

For devices designed for Constrained-Node Networks [RFC7228] some simplifications are available. Instead of being configured with (or discovering) the service registration domain, the (proposed) special-use domain name (see [RFC6761]) "default.service.arpa" is used. The details of how SRP server(s) are discovered will be specific to the constrained network, and therefore we do not suggest a specific mechanism here.

SRP clients on constrained networks are expected to receive from the network a list of SRP servers with which to register. It is the responsibility of a Constrained-Node Network supporting SRP to provide one or more SRP server addresses. It is the responsibility of the SRP server supporting a Constrained-Node Network to handle the updates appropriately. In some network environments, updates may be accepted directly into a local "default.service.arpa" zone, which has only local visibility. In other network environments, updates for names ending in "default.service.arpa" may be rewritten internally to names with broader visibility.

2.1.3. Why two variants?

The reason for these different assumptions is that low-power devices that typically use Constrained-Node Networks may have very limited battery power. The series of DNS lookups required to discover an SRP server and then communicate with it will increase the power required to advertise a service; for low-power devices, the additional flexibility this provides does not justify the additional use of power. It is also fairly typical of such networks that some network service information is obtained as part of the process of joining the network, and so this can be relied upon to provide nodes with the information they need.

Networks that are not constrained networks can more complicated topologies at the Internet layer. Nodes connected to such networks can be assumed to be able to do DNSSD service registration domain discovery. Such networks are generally able to provide registration domain discovery and routing. By requiring the use of TCP, the possibility of off-network spoofing is eliminated.

2.2. Protocol Details

We will discuss several parts to this process: how to know what to publish, how to know where to publish it (under what name), how to publish it, how to secure its publication, and how to maintain the information once published.

2.2.1. What to publish

We refer to the DNS Update message sent by services using SRP as an SRP update. Three types of updates appear in an SRP update: Service Discovery records, Service Description records, and Host Description records.

- * Service Discovery records are one or more PTR RRs, mapping from the generic service type (or subtype) to the specific Service Instance Name.
- * Service Description records are exactly one SRV RR, exactly one KEY RR, and one or more TXT RRs, all with the same name, the Service Instance Name ([RFC6763], Section 4.1). In principle Service Description records can include other record types, with the same Service Instance Name, though in practice they rarely do. The Service Instance Name MUST be referenced by one or more Service Discovery PTR records, unless it is a placeholder service registration for an intentionally non-discoverable service name.
- * The Host Description records for a service are a KEY RR, used to claim exclusive ownership of the service registration, and one or more RRs of type A or AAAA, giving the IPv4 or IPv6 address(es) of the host where the service resides.

RFC 6763 describes the details of what each of these types of updates contains and is the definitive source for information about what to publish; the reason for summarizing this here is to provide the reader with enough information about what will be published that the service registration process can be understood at a high level without first learning the full details of DNS-SD. Also, the "Service Instance Name" is an important aspect of first-come, first-serve naming, which we describe later on in this document.

2.2.2. Where to publish it

Multicast DNS uses a single namespace, ".local", which is valid on the local link. This convenience is not available for DNS-SD using the DNS protocol: services must exist in some specific unicast namespace.

As described above, full-featured devices are responsible for knowing in what domain they should register their services. Devices made for Constrained-Node Networks register in the (proposed) special use domain name [RFC6761] "default.service.arpa", and let the SRP server handle rewriting that to a different domain if necessary.

2.2.3. How to publish it

It is possible to issue a DNS Update that does several things at once; this means that it's possible to do all the work of adding a PTR resource record to the PTR RRset on the Service Name, and creating or updating the Service Instance Name and Host Description, in a single transaction.

An SRP update takes advantage of this: it is implemented as a single DNS Update message that contains a service's Service Discovery records, Service Description records, and Host Description records.

Updates done according to this specification are somewhat different than regular DNS Updates as defined in RFC2136. The RFC2136 update process can involve many update attempts: you might first attempt to add a name if it doesn't exist; if that fails, then in a second message you might update the name if it does exist but matches certain preconditions. Because the registration protocol uses a single transaction, some of this adaptability is lost.

In order to allow updates to happen in a single transaction, SRP updates do not include update prerequisites. The requirements specified in Section 2.3 are implicit in the processing of SRP updates, and so there is no need for the service sending the SRP update to put in any explicit prerequisites.

2.2.3.1. How DNS-SD Service Registration differs from standard RFC2136 DNS Update

DNS-SD Service Registration is based on standard RFC2136 DNS Update, with some differences:

- * It implements first-come first-served name allocation, protected using SIG(0) [RFC2931].
- * It enforces policy about what updates are allowed.
- * It optionally performs rewriting of "default.service.arpa" to some other domain.
- * It optionally performs automatic population of the address-to-name reverse mapping domains.
- * An SRP server is not required to implement general DNS Update prerequisite processing.
- * SRP clients are allowed to send updates to the generic domain "default.service.arpa"

2.2.4. How to secure it

Traditional DNS update is secured using the TSIG protocol, which uses a secret key shared between the DNS Update client (which issues the update) and the server (which authenticates it). This model does not work for automatic service registration.

The goal of securing the DNS-SD Registration Protocol is to provide the best possible security given the constraint that service registration has to be automatic. It is possible to layer more operational security on top of what we describe here, but what we describe here is an improvement over the security of mDNS. The goal is not to provide the level of security of a network managed by a skilled operator.

2.2.4.1. First-Come First-Served Naming

First-Come First-Serve naming provides a limited degree of security: a service that registers its service using DNS-SD Registration protocol is given ownership of a name for an extended period of time based on the key used to authenticate the DNS Update. As long as the registration service remembers the name and the key used to register that name, no other service can add or update the information associated with that. FCFS naming is used to protect both the Service Description and the Host Description.

2.2.5. Service Behavior

2.2.5.1. Public/Private key pair generation and storage

The service generates a public/private key pair. This key pair **MUST** be stored in stable storage; if there is no writable stable storage on the SRP client, the SRP client **MUST** be pre-configured with a public/private key pair in read-only storage that can be used. This key pair **MUST** be unique to the device. This key pair **MUST** be unique to the device. A device with rewritable storage should retain this key indefinitely. When the device changes ownership, it may be appropriate to erase the old key and install a new one. Therefore, the SRP client on the device **SHOULD** provide a mechanism to overwrite the key, for example as the result of a "factory reset."

When sending DNS updates, the service includes a KEY record containing the public portion of the key in each Host Description update and each Service Description update. Each KEY record **MUST** contain the same public key. The update is signed using SIG(0), using the private key that corresponds to the public key in the KEY record. The lifetimes of the records in the update is set using the EDNS(0) Update Lease option [I-D.sekar-dns-ul].

The KEY record in Service Description updates MAY be omitted for brevity; if it is omitted, the SRP server MUST behave as if the same KEY record that is given for the Host Description is also given for each Service Description for which no KEY record is provided. Omitted KEY records are not used when computing the SIG(0) signature.

2.2.5.2. Name Conflict Handling

Both Host Description records and Service Description Records can have names that result in name conflicts. Service Discovery records cannot have name conflicts. If any Host Description or Service Description record is found by the server to have a conflict with an existing name, the server will respond to the SRP Update with a YXDOMAIN rcode. In this case, the Service MUST either abandon the service registration attempt, or else choose a new name.

There is no specific requirement for how this is done; typically, however, the service will append a number to the preferred name. This number could be sequentially increasing, or could be chosen randomly. One existing implementation attempts several sequential numbers before choosing randomly. So for instance, it might try host.service.arpa, then host-1.service.arpa, then host-2.service.arpa, then host-31773.service.arpa.

2.2.5.3. Record Lifetimes

The lifetime of the DNS-SD PTR, SRV, A, AAAA and TXT records [RFC6763] uses the LEASE field of the Update Lease option, and is typically set to two hours. This means that if a device is disconnected from the network, it does not appear in the user interfaces of devices looking for services of that type for too long.

The lifetime of the KEY records is set using the KEY-LEASE field of the Update Lease Option, and should be set to a much longer time, typically 14 days. The result of this is that even though a device may be temporarily unplugged, disappearing from the network for a few days, it makes a claim on its name that lasts much longer.

This means that even if a device is unplugged from the network for a few days, and its services are not available for that time, no other device can come along and claim its name the moment it disappears from the network. In the event that a device is unplugged from the network and permanently discarded, then its name is eventually cleaned up and made available for re-use.

2.2.5.4. Compression in SRV records

Although [RFC2782] requires that the target name in the SRV record not be compressed, an SRP client SHOULD compress the target in the SRV record. The motivation for not compressing in RFC2782 is not stated, but is assumed to be because a caching resolver that does not understand the format of the SRV record might store it as binary data and thus return an invalid pointer in response to a query. This does not apply in the case of SRP: an SRP server needs to understand SRV records in order to validate the SRP update. Compression of the target potentially saves substantial space in the SRP update.

2.2.5.5. Removing published services

2.2.5.5.1. Removing all published services

To remove all the services registered to a particular host, the SRP client retransmits its most recent update with an Update Lease option that has a LEASE value of zero. If the registration is to be permanently removed, KEY-LEASE should also be zero. Otherwise, it should have the same value it had previously; this holds the name in reserve for when the SRP client is once again able to provide the service.

SRP clients are normally expected to remove all service instances when removing a host. However, in some cases a SRP client may not have retained sufficient state to know that some service instance is pointing to a host that it is removing. This method of removing services is intended for the case where the client is going offline and does not want its services advertised. Therefore, it is sufficient for the client to send the Host Description Instruction (Section 2.3.1.3).

To support this, when removing services based on the lease time being zero, an SRP server MUST remove all service instances pointing to a host when a host is removed, even if the SRP client doesn't list them explicitly. If the key lease time is nonzero, the SRP server MUST NOT delete the KEY records for these SRP clients.

2.2.5.5.2. Removing some published services

In some use cases a client may need to remove some specific service, without removing its other services. This can be accomplished in one of two ways. To simply remove a specific service, the client sends a valid SRP update where the Service Discovery instruction (Section 2.3.1.1) contains a single Delete an RR from an RRset ([RFC2136], Section 2.5.4) update that deletes the PTR record whose target is the service instance name. The Service Description

instruction (Section 2.3.1.2) in this case contains a single Delete all RRsets from a Name ([RFC2136], Section 2.5.3) update to the service instance name.

The second alternative is used when some service is being replaced by a different service with a different service instance name. In this case, the old service is deleted as in the first alternative. The new service is added, just as it would be in an update that wasn't deleting the old service. Because both the removal of the old service and the add of the new service consist of a valid Service Discovery instruction and a valid Service Description instruction, the update as a whole is a valid SRP update, and will result in the old service being removed and the new one added, or, to put it differently, in the old service being replaced by the new service.

It is perhaps worth noting that if a service is being updated without the service instance name changing, that will look very much like the second alternative above. The difference is that because the target for the PTR record in the Service Discovery instruction is the same for both the Delete An RR From An RRset update and the Add To An RRset update, these will be seen as a single Service Description instruction, not as two instructions. The same would be true of the Service Description instruction.

Whichever of these two alternatives is used, the host lease will be updated with the lease time provided in the SRP update. In neither of these cases is it permissible to delete the host. All services must point to a host. If a host is to be deleted, this must be done using the method described in Section 2.2.5.5.1, which deletes the host and all services that have that host as their target.

2.3. SRP Server Behavior

2.3.1. Validation of Adds and Deletes

The SRP server first validates that the DNS Update is a syntactically and semantically valid DNS Update according to the rules specified in RFC2136.

SRP Updates consist of a set of `_instructions_` that together add or remove one or more services. Each instruction consists some combination of delete updates and add updates. When an instruction contains a delete and an add, the delete **MUST** precede the add.

The SRP server checks each instruction in the SRP update to see that it is either a Service Discovery update, a Service Description update, or a Host Description update. Order matters in DNS updates. Specifically, deletes must precede adds for records that the deletes

would affect; otherwise the add will have no effect. This is the only ordering constraint; aside from this constraint, updates may appear in whatever order is convenient when constructing the update.

Because the SRP update is a DNS update, it MUST contain a single question that indicates the zone to be updated. Every delete and update in an SRP update MUST be within the zone that is specified for the SRP Update.

2.3.1.1. Service Discovery Instruction

An instruction is a Service Discovery Instruction if it contains

- * exactly one "Add to an RRSet" or exactly one "Delete an RR from an RRSet" ([RFC2136], Section 2.5.1) RR update,
- * which updates a PTR RR,
- * which points to a Service Instance Name
- * for which a Service Description Instruction is present in the SRP Update
- * if the Service Discovery update is an "Add to an RRSet" instruction, the Service Description Instruction does not match if it does not contain an "Add to an RRset" update for the SRV RR describing that service.
- * if the Service Discovery Instruction is an "Delete an RR from an RRSet" update, the Service Description Instruction does not match if it contains an "Add to an RRset" update.
- * Service Discovery Instructions do not contain any other add or delete updates.

2.3.1.2. Service Description Instruction

An instruction is a Service Description Instruction if, for the appropriate Service Instance Name, it contains

- * exactly one "Delete all RRsets from a name" update for the service instance name ([RFC2136], Section 2.5.3),
- * zero or one "Add to an RRset" SRV RR,
- * zero or one "Add to an RRset" KEY RR that, if present, contains the public key corresponding to the private key that was used to sign the message (if present, the KEY MUST match the KEY RR given in the Host Description),
- * zero or more "Add to an RRset" TXT RRs,
- * If there is one "Add to an RRset" SRV update, there MUST be at least one "Add to an RRset" TXT update.
- * the target of the SRV RR Add, if present points to a hostname for which there is a Host Description Instruction in the SRP Update, or
- * if there is no "Add to an RRset" SRV RR, then either

- the name to which the "Delete all RRsets from a name" applies does not exist, or
 - there is an existing KEY RR on that name, which matches the key with which the SRP Update was signed.
- * Service Descriptions Instructions do not modify any other RRs.

An SRP server MUST correctly handle compressed names in the SRV target.

2.3.1.3. Host Description Instruction

An instruction is a Host Description Instruction if, for the appropriate hostname, it contains

- * exactly one "Delete all RRsets from a name" RR,
- * one or more "Add to an RRset" RRs of type A and/or AAAA,
- * A and/or AAAA records must be of sufficient scope to be reachable by all hosts that might query the DNS. If a link-scope address or IPv4 autoconfiguration address is provided by the SRP client, the SRP server MUST treat this as if no address records were received; that is, the Host Description is not valid.
- * exactly one "Add to an RRset" RR that adds a KEY RR that contains the public key corresponding to the private key that was used to sign the message,
- * there is a Service Instance Name Instruction in the SRP update for which the SRV RR that is added points to the hostname being updated by this update.
- * Host Description updates do not modify any other records.

2.3.2. Valid SRP Update Requirements

An SRP Update MUST include zero or more Service Discovery instructions. For each Service Discovery instruction, there MUST be at least one Service Description instruction. For each Service Description instruction there MUST be at least one Service Discovery instruction with its service instance name as the target of its PTR record. There MUST be exactly one Host Description Instruction. Every Service Description instruction must have that Host Description instruction as the target of its SRV record. A DNS Update that does not meet these constraints is not an SRP update.

A DNS Update that contains any additional adds or deletes that cannot be identified as Service Discovery, Service Description or Host Description instructions is not an SRP update. A DNS update that contains any prerequisites is not an SRP update. Such messages should either be processed as regular RFC2136 updates, including access control checks and constraint checks, if supported, or else rejected with RCODE=REFUSED.

In addition, in order for an update to be a valid SRP update, the target of every Service Discovery Instruction MUST be a Service Description Instruction that is present in the SRP Update. There MUST NOT be any Service Description Instruction to which no Service Discovery Instruction points. The target of the SRV record in every Service Description instruction MUST be the single Host Description Instruction.

If the definitions of each of these instructions are followed carefully and the update requirements are validated correctly, many DNS Updates that look very much like SRP updates nevertheless will fail to validate. For example, a DNS update that contains an RRset Add to a Service Name and an RRset Add to a Service Instance Name, where the Service Name does not reference the Service Instance Name, is not a valid SRP update message, but may be a valid RFC2136 update.

2.3.3. FCFS Name And Signature Validation

Assuming that a DNS Update message has been validated with these conditions and is a valid SRP Update, the server checks that the name in the Host Description Instruction exists. If so, then the server checks to see if the KEY record on that name is the same as the KEY record in the Host Description Instruction. The server performs the same check for the KEY records in any Service Description Instructions. For KEY records that were omitted from Service Description Instructions, the KEY from the Host Description Instruction is used. If any existing KEY record corresponding to a KEY record in the SRP Update does not match the KEY same record in the SRP Update (whether provided or taken from the Host Description Instruction), then the server MUST reject the SRP Update with the YXDOMAIN RCODE.

Otherwise, the server validates the SRP Update using SIG(0) on the public key in the KEY record of the Host Description update. If the validation fails, the server MUST reject the SRP Update with the REFUSED RCODE. Otherwise, the SRP update is considered valid and authentic, and is processed according to the method described in RFC2136.

KEY record updates omitted from Service Description update are processed as if they had been explicitly present: every Service Description that is updated MUST, after the update, have a KEY RR, and it must be the same KEY RR that is present in the Host Description to which the Service Description refers.

2.3.4. SRP Update response

The status that is returned depends on the result of processing the update, and can be either SUCCESS or SERVFAIL: all other possible outcomes should already have been accounted for when applying the constraints that qualify the update as an SRP Update.

2.3.5. Optional Behavior

The server MAY add a Reverse Mapping that corresponds to the Host Description. This is not required because the Reverse Mapping serves no protocol function, but it may be useful for debugging, e.g. in annotating network packet traces or logs. In order for the server to add a reverse mapping update, it must be authoritative for the zone or have credentials to do the update. The SRP client MAY also do a reverse mapping update if it has credentials to do so.

The server MAY apply additional criteria when accepting updates. In some networks, it may be possible to do out-of-band registration of keys, and only accept updates from pre-registered keys. In this case, an update for a key that has not been registered should be rejected with the REFUSED RCODE.

There are at least two benefits to doing this rather than simply using normal SIG(0) DNS updates. First, the same registration protocol can be used in both cases, so both use cases can be addressed by the same service implementation. Second, the registration protocol includes maintenance functionality not present with normal DNS updates.

Note that the semantics of using SRP in this way are different than for typical RFC2136 implementations: the KEY used to sign the SRP update only allows the SRP client to update records that refer to its Host Description. RFC2136 implementations do not normally provide a way to enforce a constraint of this type.

The server may also have a dictionary of names or name patterns that are not permitted. If such a list is used, updates for Service Instance Names that match entries in the dictionary are rejected with YXDOMAIN.

3. TTL Consistency

All RRs within an RRset are required to have the same TTL (Clarifications to the DNS Specification [RFC2181], Section 5.2). In order to avoid inconsistencies, SRP places restrictions on TTLs sent by services and requires that SRP servers enforce consistency.

Services sending SRP updates MUST use consistent TTLs in all RRs within the SRP update.

SRP update servers MUST check that the TTLs for all RRs within the SRP update are the same. If they are not, the SRP update MUST be rejected with a REFUSED RCODE.

Additionally, when adding RRs to an RRset, for example when processing Service Discovery records, the server MUST use the same TTL on all RRs in the RRset. How this consistency is enforced is up to the implementation.

TTLs sent in SRP updates are advisory: they indicate the SRP client's guess as to what a good TTL would be. SRP servers may override these TTLs. SRP servers SHOULD ensure that TTLs are reasonable: neither too long nor too short. The TTL should never be longer than the lease time (Section 4.1). Shorter TTLs will result in more frequent data refreshes; this increases latency on the DNS-SD client side, increases load on any caching resolvers and on the authoritative server, and also increases network load, which may be an issue for constrained networks. Longer TTLs will increase the likelihood that data in caches will be stale. TTL minimums and maximums SHOULD be configurable by the operator of the SRP server.

4. Maintenance

4.1. Cleaning up stale data

Because the DNS-SD registration protocol is automatic, and not managed by humans, some additional bookkeeping is required. When an update is constructed by the SRP client, it MUST include an EDNS(0) Update Lease Option [I-D.sekar-dns-ul]. The Update Lease Option contains two lease times: the Lease Time and the Key Lease Time.

These leases are promises, similar to DHCP leases [RFC2131], from the SRP client that it will send a new update for the service registration before the lease time expires. The Lease time is chosen to represent the time after the update during which the registered records other than the KEY record should be assumed to be valid. The Key Lease time represents the time after the update during which the KEY record should be assumed to be valid.

The reasoning behind the different lease times is discussed in the section on first-come, first-served naming (Section 2.2.4.1). SRP servers may be configured with limits for these values. A default limit of two hours for the Lease and 14 days for the SIG(0) KEY are currently thought to be good choices. Constrained devices with limited battery that wake infrequently are likely to negotiate longer leases. SRP clients that are going to continue to use names on which they hold leases should update well before the lease ends, in case the registration service is unavailable or under heavy load.

The lease time applies specifically to the host. All service instances, and all service entries for such service instances, depend on the host. When the lease on a host expires, the host and all services that reference it MUST be removed at the same time—it is never valid for a service instance to remain when the host it references has been removed. If the KEY record for the host is to remain, the KEY record for any services that reference it MUST also remain. However, the service PTR record MUST be removed, since it has no key associated with it, and since it is never valid to have a service PTR record for which there is no service instance on the target of the PTR record.

SRP Servers SHOULD also track a lease time per service instance. The reason for doing this is that a client may re-register a host with a different set of services, and not remember that some different service instance had previously been registered. In this case, when that service instance lease expires, the SRP server SHOULD remove the service instance (although the KEY record for the service instance SHOULD be retained until the key lease on that service expires). This is beneficial because if the SRP client continues to renew the host, but never mentions the stale service again, the stale service will continue to be advertised.

The SRP server MUST include an EDNS(0) Update Lease option in the response if the lease time proposed by the service has been shortened or lengthened. The service MUST check for the EDNS(0) Update Lease option in the response and MUST use the lease times from that option in place of the options that it sent to the server when deciding when to update its registration. The times may be shorter or longer than those specified in the SRP update; the SRP client must honor them in either case.

SRP clients should assume that each lease ends N seconds after the update was first transmitted, where N is the lease duration. Servers should assume that each lease ends N seconds after the update that was successfully processed was received. Because the server will always receive the update after the SRP client sent it, this avoids the possibility of misunderstandings.

SRP servers MUST reject updates that do not include an EDNS(0) Update Lease option. Dual-use servers MAY accept updates that don't include leases, but SHOULD differentiate between SRP updates and other updates, and MUST reject updates that would otherwise be SRP updates if they do not include leases.

Lease times have a completely different function than TTLs. On an authoritative DNS server, the TTL on a resource record is a constant: whenever that RR is served in a DNS response, the TTL value sent in the answer is the same. The lease time is never sent as a TTL; its sole purpose is to determine when the authoritative DNS server will delete stale records. It is not an error to send a DNS response with a TTL of 'n' when the remaining time on the lease is less than 'n'.

5. Sleep Proxy

Another use of SRP is for devices that sleep to reduce power consumption.

In this case, in addition to the DNS Update Lease option [I-D.sekar-dns-ul] described above, the device includes an EDNS(0) OWNER Option [I-D.cheshire-edns0-owner-option].

The EDNS(0) Update Lease option constitutes a promise by the device that it will wake up before this time elapses, to renew its registration and thereby demonstrate that it is still attached to the network. If it fails to renew the registration by this time, that indicates that it is no longer attached to the network, and its registration (except for the KEY in the Host Description) should be deleted.

The EDNS(0) OWNER Option indicates that the device will be asleep, and will not be receptive to normal network traffic. When a DNS server receives a DNS Update with an EDNS(0) OWNER Option, that signifies that the SRP server should set up a proxy for any IPv4 or IPv6 address records in the DNS Update message. This proxy should send ARP or ND messages claiming ownership of the IPv4 and/or IPv6 addresses in the records in question. In addition, the proxy should answer future ARP or ND requests for those IPv4 and/or IPv6 addresses, claiming ownership of them. When the DNS server receives a TCP SYN or UDP packet addressed to one of the IPv4 or IPv6 addresses for which it proxying, it should then wake up the sleeping device using the information in the EDNS(0) OWNER Option. At present version 0 of the OWNER Option specifies the "Wake-on-LAN Magic Packet" that needs to be sent; future versions could be extended to specify other wakeup mechanisms.

Note that although the authoritative DNS server that implements the SRP function need not be on the same link as the sleeping host, the Sleep Proxy must be on the same link.

It is not required that sleepy nodes on a Constrained-Node Network support sleep proxy. Such devices may have different mechanisms for dealing with sleep and wakeup. An SRP registration for such a device will be useful regardless of the mechanism whereby messages are delivered to the sleepy end device. For example, the message might be held in a buffer for an extended period of time by an intermediate device on a mesh network, and then delivered to the device when it wakes up. The exact details of such behaviors are out of scope for this document.

6. Security Considerations

6.1. Source Validation

SRP updates have no authorization semantics other than first-come, first-served. This means that if an attacker from outside of the administrative domain of the server knows the server's IP address, it can in principle send updates to the server that will be processed successfully. Servers should therefore be configured to reject updates from source addresses outside of the administrative domain of the server.

For updates sent to an anycast IP address of an SRP server, this validation must be enforced by every router on the path from the Constrained-Device Network to the unconstrained portion of the network. For TCP updates, the initial SYN-SYN+ACK handshake prevents updates being forged by an off-network attacker. In order to ensure that this handshake happens, Service Discovery Protocol servers relying on three-way-handshake validation MUST NOT accept TCP Fast Open payloads. If the network infrastructure allows it, an SRP server MAY accept TCP Fast Open payloads if all such packets are validated along the path, and the network is able to reject this type of spoofing at all ingress points.

Note that these rules only apply to the validation of SRP updates. A server that accepts updates from SRP clients may also accept other DNS updates, and those DNS updates may be validated using different rules. However, in the case of a DNS service that accepts SRP updates, the intersection of the SRP update rules and whatever other update rules are present must be considered very carefully.

For example, a normal, authenticated DNS update to any RR that was added using SRP, but that is authenticated using a different key, could be used to override a promise made by the registration

protocol, by replacing all or part of the service registration information with information provided by an SRP client. An implementation that allows both kinds of updates should not allow DNS Update clients that are using different authentication and authorization credentials to update records added by SRP clients.

6.2. SRP Server Authentication

This specification does not provide a mechanism for validating responses from DNS servers to SRP clients. In the case of Constrained Network/Constrained Node clients, such validation isn't practical because there's no way to establish trust. In principle, a KEY RR could be used by a non-constrained SRP client to validate responses from the server, but this is not required, nor do we specify a mechanism for determining which key to use.

6.3. Required Signature Algorithm

For validation, SRP servers MUST implement the ECDSA_{P256}SHA256 signature algorithm. SRP servers SHOULD implement the algorithms specified in [RFC8624], Section 3.1, in the validation column of the table, that are numbered 13 or higher and have a "MUST", "RECOMMENDED", or "MAY" designation in the validation column of the table. SRP clients MUST NOT assume that any algorithm numbered lower than 13 is available for use in validating SIG(0) signatures.

7. Privacy Considerations

Because DNSSD SRP updates can be sent off-link, the privacy implications of SRP are different than for multicast DNS responses. Host implementations that are using TCP SHOULD also use TLS if available. Server implementations MUST offer TLS support. The use of TLS with DNS is described in [RFC7858] and [RFC8310].

Hosts that implement TLS support SHOULD NOT fall back to TCP; since servers are required to support TLS, it is entirely up to the host implementation whether to use it.

Public keys can be used as identifiers to track hosts. SRP servers MAY elect not to return KEY records for queries for SRP registrations.

8. Delegation of 'service.arpa.'

In order to be fully functional, the owner of the 'arpa.' zone must add a delegation of 'service.arpa.' in the '.arpa.' zone [RFC3172]. This delegation should be set up as was done for 'home.arpa', as a result of the specification in Section 7 of [RFC8375].

9. IANA Considerations

9.1. Registration and Delegation of 'service.arpa' as a Special-Use Domain Name

IANA is requested to record the domain name 'service.arpa.' in the Special-Use Domain Names registry [SUDN]. IANA is requested, with the approval of IAB, to implement the delegation requested in Section 8.

IANA is further requested to add a new entry to the "Transport-Independent Locally-Served Zones" subregistry of the the "Locally-Served DNS Zones" registry [LSDZ]. The entry will be for the domain 'service.arpa.' with the description "DNS-SD Registration Protocol Special-Use Domain", listing this document as the reference.

9.2. 'dnsssd-srp' Service Name

IANA is also requested to add a new entry to the Service Names and Port Numbers registry for dnsssd-srp with a transport type of tcp. No port number is to be assigned. The reference should be to this document, and the Assignee and Contact information should reference the authors of this document. The Description should be as follows:

Availability of DNS Service Discovery Service Registration Protocol Service for a given domain is advertised using the "_dnsssd-srp._tcp.<domain>." SRV record gives the target host and port where DNSSD Service Registration Service is provided for the named domain.

9.3. 'dnsssd-srp-tls' Service Name

IANA is also requested to add a new entry to the Service Names and Port Numbers registry for dnsssd-srp with a transport type of tcp. No port number is to be assigned. The reference should be to this document, and the Assignee and Contact information should reference the authors of this document. The Description should be as follows:

Availability of DNS Service Discovery Service Registration Protocol Service for a given domain over TLS is advertised using the "_dnsssd-srp-tls._tcp.<domain>." SRV record gives the target host and port where DNSSD Service Registration Service is provided for the named domain.

9.4. Anycast Address

IANA is requested to allocate an IPv6 Anycast address from the IPv6 Special-Purpose Address Registry, similar to the Port Control Protocol anycast address, 2001:1::1. The value TBD should be replaced with the actual allocation in the table that follows. The values for the registry are:

Attribute	value
Address Block	2001:1::TBD/128
Name	DNS-SD Service Registration Protocol Anycast Address
RFC	[this document]
Allocation Date	[date of allocation]
Termination Date	N/A
Source	True
Destination	True
Forwardable	True
Global	True
Reserved-by-protocol	False

Table 1

10. Implementation Status

[Note to the RFC Editor: please remove this section prior to publication.]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 7942. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was

supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to RFC 7942, "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

There are two known independent implementations of SRP clients:

- * SRP Client for OpenThread:
<https://github.com/openthread/openthread/pull/6038>
- * mDNSResponder open source project: <https://github.com/Abhayakara/mdnsresponder>

There are two related implementations of an SRP server. One acts as a DNS Update proxy, taking an SRP update and applying it to the specified DNS zone using DNS update. The other acts as an Advertising Proxy [I-D.sctl-advertising-proxy]. Both are included in the mDNSResponder open source project mentioned above.

11. Acknowledgments

Thanks to Toke Høiland-Jørgensen, Jonathan Hui, Esko Dijk, Kangping Dong and Abtin Keshavarzian for their thorough technical reviews. Thanks to Kangping and Abtin as well for testing the document by doing an independent implementation. Thanks to Tamara Kemper for doing a nice developmental edit, Tim Wattenberg for doing a SRP client proof-of-concept implementation at the Montreal Hackathon at IETF 102, and Tom Pusateri for reviewing during the hackathon and afterwards.

12. Normative References

- [I-D.sekar-dns-ul]
Cheshire, S. and T. Lemon, "Dynamic DNS Update Leases", Work in Progress, Internet-Draft, draft-sekar-dns-ul-02, 2 August 2018,
<<https://tools.ietf.org/html/draft-sekar-dns-ul-02>>.
- [RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, DOI 10.17487/RFC2132, March 1997,
<<https://www.rfc-editor.org/info/rfc2132>>.

- [RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, DOI 10.17487/RFC2136, April 1997, <<https://www.rfc-editor.org/info/rfc2136>>.
- [RFC2931] Eastlake 3rd, D., "DNS Request and Transaction Signatures (SIG(0)s)", RFC 2931, DOI 10.17487/RFC2931, September 2000, <<https://www.rfc-editor.org/info/rfc2931>>.
- [RFC3172] Huston, G., Ed., "Management Guidelines & Operational Requirements for the Address and Routing Parameter Area Domain ("arpa")", BCP 52, RFC 3172, DOI 10.17487/RFC3172, September 2001, <<https://www.rfc-editor.org/info/rfc3172>>.
- [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>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC8106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 8106, DOI 10.17487/RFC8106, March 2017, <<https://www.rfc-editor.org/info/rfc8106>>.
- [RFC8375] Pfister, P. and T. Lemon, "Special-Use Domain 'home.arpa.'", RFC 8375, DOI 10.17487/RFC8375, May 2018, <<https://www.rfc-editor.org/info/rfc8375>>.
- [RFC8624] Wouters, P. and O. Sury, "Algorithm Implementation Requirements and Usage Guidance for DNSSEC", RFC 8624, DOI 10.17487/RFC8624, June 2019, <<https://www.rfc-editor.org/info/rfc8624>>.
- [RFC8765] Pusateri, T. and S. Cheshire, "DNS Push Notifications", RFC 8765, DOI 10.17487/RFC8765, June 2020, <<https://www.rfc-editor.org/info/rfc8765>>.
- [SUDN] "Special-Use Domain Names Registry", July 2012, <<https://www.iana.org/assignments/special-use-domain-names/special-use-domain-names.xhtml>>.
- [LSDZ] "Locally-Served DNS Zones Registry", July 2011, <<https://www.iana.org/assignments/locally-served-dns-zones/locally-served-dns-zones.xhtml>>.

13. Informative References

- [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>>.
- [RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <<https://www.rfc-editor.org/info/rfc2131>>.
- [RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, DOI 10.17487/RFC2181, July 1997, <<https://www.rfc-editor.org/info/rfc2181>>.
- [RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, DOI 10.17487/RFC2782, February 2000, <<https://www.rfc-editor.org/info/rfc2782>>.
- [RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, DOI 10.17487/RFC3007, November 2000, <<https://www.rfc-editor.org/info/rfc3007>>.
- [RFC6760] Cheshire, S. and M. Krochmal, "Requirements for a Protocol to Replace the AppleTalk Name Binding Protocol (NBP)", RFC 6760, DOI 10.17487/RFC6760, February 2013, <<https://www.rfc-editor.org/info/rfc6760>>.
- [RFC6761] Cheshire, S. and M. Krochmal, "Special-Use Domain Names", RFC 6761, DOI 10.17487/RFC6761, February 2013, <<https://www.rfc-editor.org/info/rfc6761>>.
- [RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <<https://www.rfc-editor.org/info/rfc6762>>.
- [RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <<https://www.rfc-editor.org/info/rfc7228>>.
- [RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <<https://www.rfc-editor.org/info/rfc8310>>.

- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.
- [RFC8766] Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", RFC 8766, DOI 10.17487/RFC8766, June 2020, <<https://www.rfc-editor.org/info/rfc8766>>.
- [I-D.cheshire-dnssd-roadmap]
Cheshire, S., "Service Discovery Road Map", Work in Progress, Internet-Draft, draft-cheshire-dnssd-roadmap-03, 23 October 2018, <<https://tools.ietf.org/html/draft-cheshire-dnssd-roadmap-03>>.
- [I-D.cheshire-edns0-owner-option]
Cheshire, S. and M. Krochmal, "EDNS0 OWNER Option", Work in Progress, Internet-Draft, draft-cheshire-edns0-owner-option-01, 3 July 2017, <<https://tools.ietf.org/html/draft-cheshire-edns0-owner-option-01>>.
- [I-D.sctl-advertising-proxy]
Cheshire, S. and T. Lemon, "Advertising Proxy for DNS-SD Service Registration Protocol", Work in Progress, Internet-Draft, draft-sctl-advertising-proxy-00, 13 July 2020, <<https://tools.ietf.org/html/draft-sctl-advertising-proxy-00>>.
- [ZC] Cheshire, S. and D.H. Steinberg, "Zero Configuration Networking: The Definitive Guide", O'Reilly Media, Inc. , ISBN 0-596-10100-7, December 2005.

Appendix A. Testing using standard RFC2136-compliant servers

It may be useful to set up a DNS server for testing that does not implement SRP. This can be done by configuring the server to listen on the anycast address, or advertising it in the `_dnssd-srp._tcp.<zone>` SRV and `_dnssd-srp-tls._tcp.<zone>` record. It must be configured to be authoritative for "default.service.arpa", and to accept updates from hosts on local networks for names under "default.service.arpa" without authentication, since such servers will not have support for FCFS authentication (Section 2.2.4.1).

A server configured in this way will be able to successfully accept and process SRP updates from services that send SRP updates. However, no prerequisites will be applied, and this means that the test server will accept internally inconsistent SRP updates, and will not stop two SRP updates, sent by different services, that claim the same name(s), from overwriting each other.

Since SRP updates are signed with keys, validation of the SIG(0) algorithm used by the client can be done by manually installing the client public key on the DNS server that will be receiving the updates. The key can then be used to authenticate the client, and can be used as a requirement for the update. An example configuration for testing SRP using BIND 9 is given in Appendix C.

Appendix B. How to allow services to update standard RFC2136-compliant servers

Ordinarily SRP updates will fail when sent to an RFC 2136-compliant server that does not implement SRP because the zone being updated is "default.service.arpa", and no DNS server that is not an SRP server should normally be configured to be authoritative for "default.service.arpa". Therefore, a service that sends an SRP update can tell that the receiving server does not support SRP, but does support RFC2136, because the RCODE will either be NOTZONE, NOTAUTH or REFUSED, or because there is no response to the update request (when using the anycast address)

In this case a service MAY attempt to register itself using regular RFC2136 DNS updates. To do so, it must discover the default registration zone and the DNS server designated to receive updates for that zone, as described earlier, using the _dns-update._udp SRV record. It can then make the update using the port and host pointed to by the SRV record, and should use appropriate prerequisites to avoid overwriting competing records. Such updates are out of scope for SRP, and a service that implements SRP MUST first attempt to use SRP to register itself, and should only attempt to use RFC2136 backwards compatibility if that fails. Although the owner name for the SRV record specifies the UDP protocol for updates, it is also possible to use TCP, and TCP should be required to prevent spoofing.

Appendix C. Sample BIND9 configuration for default.service.arpa.

```
zone "default.service.arpa." {  
    type master;  
    file "/etc/bind/master/service.db";  
    allow-update { key demo.default.service.arpa.; };  
};
```

Figure 1: Zone Configuration in named.conf

```

$ORIGIN .
$TTL 57600 ; 16 hours
default.service.arpa IN SOA      ns3.default.service.arpa.
                                postmaster.default.service.arpa. (
                                2951053287 ; serial
                                3600      ; refresh (1 hour)
                                1800      ; retry (30 minutes)
                                604800    ; expire (1 week)
                                3600      ; minimum (1 hour)
                                )
                                NS       ns3.default.service.arpa.
                                SRV 0 0 53 ns3.default.service.arpa.
$ORIGIN default.service.arpa.
$TTL 3600 ; 1 hour
_ipps._tcp PTR      demo._ipps._tcp
$ORIGIN _ipps._tcp.default.service.arpa.
demo      TXT      "0"
          SRV 0 0 9992 demo.default.service.arpa.
$ORIGIN _udp.default.service.arpa.
$TTL 3600 ; 1 hour
_dns-update PTR      ns3.default.service.arpa.
$ORIGIN _tcp.default.service.arpa.
_dns-sd-srp PTR      ns3.default.service.arpa.
$ORIGIN default.service.arpa.
$TTL 300 ; 5 minutes
ns3      AAAA      2001:db8:0:1::1
$TTL 3600 ; 1 hour
demo     AAAA      2001:db8:0:2::1
          KEY 513 3 13 (
                    qweEmaaq0FAWok5//ftuQtZgiZoiFSUsm0srWREdywQU
                    9dpvtOhrdKWUuPT3uEFF5TZU6B4q1z1I662GdaUwqg==
                    ); alg = ECDSAP256SHA256 ; key id = 15008
          AAAA      ::1

```

Figure 2: Example Zone file

Authors' Addresses

Ted Lemon
 Apple Inc.
 One Apple Park Way
 Cupertino, California 95014
 United States of America

Email: mellon@fugue.com

Stuart Cheshire
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America

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

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: 26 October 2022

T. Lemon
S. Cheshire
Apple Inc.
24 April 2022

Service Registration Protocol for DNS-Based Service Discovery
draft-ietf-dnssd-srp-13

Abstract

The Service Registration Protocol for DNS-Based Service Discovery uses the standard DNS Update mechanism to enable DNS-Based Service Discovery using only unicast packets. This makes it possible to deploy DNS Service Discovery without multicast, which greatly improves scalability and improves performance on networks where multicast service is not an optimal choice, particularly 802.11 (Wi-Fi) and 802.15.4 (IoT) networks. DNS-SD Service registration uses public keys and SIG(0) to allow services to defend their registrations against attack.

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 <https://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 26 October 2022.

Copyright Notice

Copyright (c) 2022 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	3
2. Service Registration Protocol	5
2.1. Protocol Variants	5
2.1.1. Full-featured Hosts	5
2.1.2. Constrained Hosts	6
2.1.3. Why two variants?	6
2.2. Protocol Details	7
2.2.1. What to publish	7
2.2.2. Where to publish it	7
2.2.3. How to publish it	8
2.2.3.1. How DNS-SD Service Registration differs from standard RFC2136 DNS Update	8
2.2.4. How to secure it	9
2.2.4.1. First-Come First-Served Naming	9
2.2.5. Service Behavior	9
2.2.5.1. Public/Private key pair generation and storage	9
2.2.5.2. Name Conflict Handling	10
2.2.5.3. Record Lifetimes	10
2.2.5.4. Compression in SRV records	11
2.2.5.5. Removing published services	11
2.3. Validation and Processing of SRP Updates	12
2.3.1. Validation of Adds and Deletes	12
2.3.1.1. Service Discovery Instruction	13
2.3.1.2. Service Description Instruction	13
2.3.1.3. Host Description Instruction	14
2.3.2. Valid SRP Update Requirements	14
2.3.3. FCFS Name And Signature Validation	15
2.3.4. Handling of Service Subtypes	16
2.3.5. SRP Update response	16
2.3.6. Optional Behavior	16
3. TTL Consistency	17
4. Maintenance	18
4.1. Cleaning up stale data	18
5. Security Considerations	19
5.1. Source Validation	19
5.2. SRP Server Authentication	20
5.3. Required Signature Algorithm	20
6. Privacy Considerations	21
7. Delegation of 'service.arpa.'	21
8. IANA Considerations	21
8.1. Registration and Delegation of 'service.arpa' as a Special-Use Domain Name	21

8.2. 'dnssd-srp' Service Name	21
8.3. 'dnssd-srp-tls' Service Name	22
8.4. Anycast Address	22
9. Implementation Status	23
10. Acknowledgments	24
11. Normative References	24
12. Informative References	26
Appendix A. Testing using standard RFC2136-compliant servers . .	27
Appendix B. How to allow services to update standard RFC2136-compliant servers	28
Appendix C. Sample BIND9 configuration for default.service.arpa.	28
Authors' Addresses	29

1. Introduction

DNS-Based Service Discovery [RFC6763] is a component of Zero Configuration Networking [RFC6760] [ZC] [I-D.cheshire-dnssd-roadmap].

This document describes an enhancement to DNS-Based Service Discovery [RFC6763] that allows services to register their services using the DNS protocol rather than using Multicast DNS [RFC6762] (mDNS). There is already a large installed base of DNS-SD clients that can discover services using the DNS protocol.

This document is intended for three audiences: implementors of software that provides services that should be advertised using DNS-SD, implementors of DNS servers that will be used in contexts where DNS-SD registration is needed, and administrators of networks where DNS-SD service is required. The document is intended to provide sufficient information to allow interoperable implementation of the registration protocol.

DNS-Based Service Discovery (DNS-SD) allows services to advertise the fact that they provide service, and to provide the information required to access that service. DNS-SD clients can then discover the set of services of a particular type that are available. They can then select a service from among those that are available and obtain the information required to use it. Although DNS-SD using the DNS protocol (as opposed to mDNS) can be more efficient and versatile, it is not common in practice, because of the difficulties associated with updating authoritative DNS services with service information.

Existing practice for updating DNS zones is to either manually enter new data, or else use DNS Update [RFC2136]. Unfortunately DNS Update requires either that the authoritative DNS server automatically trust updates, or else that the DNS Update client have some kind of shared

secret or public key that is known to the DNS server and can be used to authenticate the update. Furthermore, DNS Update can be a fairly chatty process, requiring multiple round trips with different conditional predicates to complete the update process.

The SRP protocol adds a set of default heuristics for processing DNS updates that eliminates the need for DNS update conditional predicates: instead, the SRP server has a set of default predicates that are applied to the update, and the update either succeeds entirely, or fails in a way that allows the registering service to know what went wrong and construct a new update.

SRP also adds a feature called First-Come, First-Served Naming, which allows the registering service to claim a name that is not yet in use, and, using SIG(0) [RFC2931], to authenticate both the initial claim and subsequent updates. This prevents name conflicts, since a second SRP service attempting to claim the same name will not possess the SIG(0) key used by the first service to claim it, and so its claim will be rejected and the second service will have to choose a new name.

Finally, SRP adds the concept of a 'lease,' similar to leases in Dynamic Host Configuration Protocol [RFC8415]. The SRP registration itself has a lease which may be on the order of an hour; if the registering service does not renew the lease before it has elapsed, the registration is removed. The claim on the name can have a longer lease, so that another service cannot claim the name, even though the registration has expired.

The Service Registration Protocol for DNS-SD (SRP), described in this document, provides a reasonably secure mechanism for publishing this information. Once published, these services can be readily discovered by DNS-SD clients using standard DNS lookups.

The DNS-SD specification [RFC6763], Section 10 ("Populating the DNS with Information"), briefly discusses ways that services can publish their information in the DNS namespace. In the case of mDNS, it allows services to publish their information on the local link, using names in the ".local" namespace, which makes their services directly discoverable by peers attached to that same local link.

RFC6763 also allows clients to discover services using the DNS protocol [RFC1035]. This can be done by having a system administrator manually configure service information in the DNS, but manually populating DNS authoritative server databases is costly and potentially error-prone, and requires a knowledgeable network administrator. Consequently, although all DNS-SD client implementations of which we are aware support DNS-SD using DNS queries, in practice it is used much less frequently than mDNS.

The Discovery Proxy [RFC8766] provides one way to automatically populate the DNS namespace, but is only appropriate on networks where services are easily advertised using mDNS. This document describes a solution more suitable for networks where multicast is inefficient, or where sleepy devices are common, by supporting both offering of services, and discovery of services, using unicast.

2. Service Registration Protocol

Services that implement SRP use DNS Update [RFC2136] [RFC3007] to publish service information in the DNS. Two variants exist, one for full-featured hosts, and one for devices designed for "Constrained-Node Networks" [RFC7228]. An SRP server is most likely an authoritative DNS server, or else is updating an authoritative DNS server. There is no requirement that the server that is receiving SRP requests be the same server that is answering queries that return records that have been registered.

2.1. Protocol Variants

2.1.1. Full-featured Hosts

Full-featured hosts are either configured manually with a registration domain, or use the "dr._dns-sd._udp.<domain>" query ([RFC6763], Section 11) to learn the default registration domain from the network. RFC6763 says to discover the registration domain using either ".local" or a network-supplied domain name for <domain>. Services using SRP MUST use the domain name received through the DHCPv4 Domain Name option ([RFC2132], Section 3.17), if available, or the Neighbor Discovery DNS Search List option [RFC8106]. If the DNS Search List option contains more than one domain name, it MUST NOT be used. If neither option is available, the Service Registration protocol is not available on the local network.

Manual configuration of the registration domain can be done either by querying the list of available registration zones ("r._dns-sd._udp") and allowing the user to select one from the UI, or by any other means appropriate to the particular use case being addressed. Full-featured devices construct the names of the SRV, TXT, and PTR records

describing their service(s) as subdomains of the chosen service registration domain. For these names they then discover the zone apex of the closest enclosing DNS zone using SOA queries [RFC8765]. Having discovered the enclosing DNS zone, they query for the "_dnssd-srp._tcp.<zone>" SRV record to discover the server to which they should send DNS updates. Hosts that support SRP Updates using TLS use the "_dnssd-srp-tls._tcp.<zone>" SRV record instead.

2.1.2. Constrained Hosts

For devices designed for Constrained-Node Networks [RFC7228] some simplifications are available. Instead of being configured with (or discovering) the service registration domain, the (proposed) special-use domain name (see [RFC6761]) "default.service.arpa" is used. The details of how SRP server(s) are discovered will be specific to the constrained network, and therefore we do not suggest a specific mechanism here.

SRP clients on constrained networks are expected to receive from the network a list of SRP servers with which to register. It is the responsibility of a Constrained-Node Network supporting SRP to provide one or more SRP server addresses. It is the responsibility of the SRP server supporting a Constrained-Node Network to handle the updates appropriately. In some network environments, updates may be accepted directly into a local "default.service.arpa" zone, which has only local visibility. In other network environments, updates for names ending in "default.service.arpa" may be rewritten internally to names with broader visibility.

2.1.3. Why two variants?

The reason for these different assumptions is that low-power devices that typically use Constrained-Node Networks may have very limited battery power. The series of DNS lookups required to discover an SRP server and then communicate with it will increase the power required to advertise a service; for low-power devices, the additional flexibility this provides does not justify the additional use of power. It is also fairly typical of such networks that some network service information is obtained as part of the process of joining the network, and so this can be relied upon to provide nodes with the information they need.

Networks that are not constrained networks can have more complicated topologies at the Internet layer. Nodes connected to such networks can be assumed to be able to do DNSSD service registration domain discovery. Such networks are generally able to provide registration domain discovery and routing. By requiring the use of TCP, the possibility of off-network spoofing is eliminated.

2.2. Protocol Details

We will discuss several parts to this process: how to know what to publish, how to know where to publish it (under what name), how to publish it, how to secure its publication, and how to maintain the information once published.

2.2.1. What to publish

We refer to the DNS Update message sent by services using SRP as an SRP Update. Three types of updates appear in an SRP update: Service Discovery records, Service Description records, and Host Description records.

- * Service Discovery records are one or more PTR RRs, mapping from the generic service type (or subtype) to the specific Service Instance Name.
- * Service Description records are exactly one SRV RR, exactly one KEY RR, and one or more TXT RRs, all with the same name, the Service Instance Name ([RFC6763], Section 4.1). In principle Service Description records can include other record types, with the same Service Instance Name, though in practice they rarely do. The Service Instance Name MUST be referenced by one or more Service Discovery PTR records, unless it is a placeholder service registration for an intentionally non-discoverable service name.
- * The Host Description records for a service are a KEY RR, used to claim exclusive ownership of the service registration, and one or more RRs of type A or AAAA, giving the IPv4 or IPv6 address(es) of the host where the service resides.

[RFC6763] describes the details of what each of these types of updates contains, with the exception of the KEY RR, which is defined in [RFC2539]. These RFCs should be considered the definitive source for information about what to publish; the reason for summarizing this here is to provide the reader with enough information about what will be published that the service registration process can be understood at a high level without first learning the full details of DNS-SD. Also, the "Service Instance Name" is an important aspect of first-come, first-serve naming, which we describe later on in this document.

2.2.2. Where to publish it

Multicast DNS uses a single namespace, ".local", which is valid on the local link. This convenience is not available for DNS-SD using the DNS protocol: services must exist in some specific unicast namespace.

As described above, full-featured devices are responsible for knowing in what domain they should register their services. Devices made for Constrained-Node Networks register in the (proposed) special use domain name [RFC6761] "default.service.arpa", and let the SRP server handle rewriting that to a different domain if necessary.

2.2.3. How to publish it

It is possible to issue a DNS Update that does several things at once; this means that it's possible to do all the work of adding a PTR resource record to the PTR RRset on the Service Name, and creating or updating the Service Instance Name and Host Description, in a single transaction.

An SRP Update takes advantage of this: it is implemented as a single DNS Update message that contains a service's Service Discovery records, Service Description records, and Host Description records.

Updates done according to this specification are somewhat different than regular DNS Updates as defined in RFC2136. The RFC2136 update process can involve many update attempts: you might first attempt to add a name if it doesn't exist; if that fails, then in a second message you might update the name if it does exist but matches certain preconditions. Because the registration protocol uses a single transaction, some of this adaptability is lost.

In order to allow updates to happen in a single transaction, SRP Updates do not include update prerequisites. The requirements specified in Section 2.3 are implicit in the processing of SRP Updates, and so there is no need for the service sending the SRP Update to put in any explicit prerequisites.

2.2.3.1. How DNS-SD Service Registration differs from standard RFC2136 DNS Update

DNS-SD Service Registration is based on standard RFC2136 DNS Update, with some differences:

- * It implements first-come first-served name allocation, protected using SIG(0) [RFC2931].
- * It enforces policy about what updates are allowed.
- * It optionally performs rewriting of "default.service.arpa" to some other domain.
- * It optionally performs automatic population of the address-to-name reverse mapping domains.
- * An SRP server is not required to implement general DNS Update prerequisite processing.

- * Constrained-Node SRP clients are allowed to send updates to the generic domain "default.service.arpa"

2.2.4. How to secure it

Traditional DNS update is secured using the TSIG protocol, which uses a secret key shared between the DNS Update client (which issues the update) and the server (which authenticates it). This model does not work for automatic service registration.

The goal of securing the DNS-SD Registration Protocol is to provide the best possible security given the constraint that service registration has to be automatic. It is possible to layer more operational security on top of what we describe here, but what we describe here is an improvement over the security of mDNS. The goal is not to provide the level of security of a network managed by a skilled operator.

2.2.4.1. First-Come First-Served Naming

First-Come First-Serve naming provides a limited degree of security: a service that registers its service using DNS-SD Registration protocol is given ownership of a name for an extended period of time based on the key used to authenticate the DNS Update. As long as the registration service remembers the name and the key used to register that name, no other service can add or update the information associated with that. FCFS naming is used to protect both the Service Description and the Host Description.

2.2.5. Service Behavior

2.2.5.1. Public/Private key pair generation and storage

The service generates a public/private key pair. This key pair **MUST** be stored in stable storage; if there is no writable stable storage on the SRP client, the SRP client **MUST** be pre-configured with a public/private key pair in read-only storage that can be used. This key pair **MUST** be unique to the device. A device with rewritable storage should retain this key indefinitely. When the device changes ownership, it may be appropriate to erase the old key and install a new one. Therefore, the SRP client on the device **SHOULD** provide a mechanism to overwrite the key, for example as the result of a "factory reset."

When sending DNS updates, the service includes a KEY record containing the public portion of the key in each Host Description Instruction and each Service Description Instruction. Each KEY record **MUST** contain the same public key. The update is signed using

SIG(0), using the private key that corresponds to the public key in the KEY record. The lifetimes of the records in the update is set using the EDNS(0) Update Lease option [I-D.sekar-dns-ul].

The KEY record in Service Description updates MAY be omitted for brevity; if it is omitted, the SRP server MUST behave as if the same KEY record that is given for the Host Description is also given for each Service Description for which no KEY record is provided. Omitted KEY records are not used when computing the SIG(0) signature.

2.2.5.2. Name Conflict Handling

Both Host Description records and Service Description Records can have names that result in name conflicts. Service Discovery records cannot have name conflicts. If any Host Description or Service Description record is found by the server to have a conflict with an existing name, the server will respond to the SRP Update with a YXDOMAIN rcode. In this case, the Service MUST either abandon the service registration attempt, or else choose a new name.

There is no specific requirement for how this is done; typically, however, the service will append a number to the preferred name. This number could be sequentially increasing, or could be chosen randomly. One existing implementation attempts several sequential numbers before choosing randomly. So for instance, it might try host.service.arpa, then host-1.service.arpa, then host-2.service.arpa, then host-31773.service.arpa.

2.2.5.3. Record Lifetimes

The lifetime of the DNS-SD PTR, SRV, A, AAAA and TXT records [RFC6763] uses the LEASE field of the Update Lease option, and is typically set to two hours. This means that if a device is disconnected from the network, it does not appear in the user interfaces of devices looking for services of that type for too long.

The lifetime of the KEY records is set using the KEY-LEASE field of the Update Lease Option, and should be set to a much longer time, typically 14 days. The result of this is that even though a device may be temporarily unplugged, disappearing from the network for a few days, it makes a claim on its name that lasts much longer.

This means that even if a device is unplugged from the network for a few days, and its services are not available for that time, no other device can come along and claim its name the moment it disappears from the network. In the event that a device is unplugged from the network and permanently discarded, then its name is eventually cleaned up and made available for re-use.

2.2.5.4. Compression in SRV records

Although [RFC2782] requires that the target name in the SRV record not be compressed, an SRP client SHOULD compress the target in the SRV record. The motivation for not compressing in RFC2782 is not stated, but is assumed to be because a caching resolver that does not understand the format of the SRV record might store it as binary data and thus return an invalid pointer in response to a query. This does not apply in the case of SRP: an SRP server needs to understand SRV records in order to validate the SRP Update. Compression of the target potentially saves substantial space in the SRP Update.

2.2.5.5. Removing published services

2.2.5.5.1. Removing all published services

To remove all the services registered to a particular host, the SRP client retransmits its most recent update with an Update Lease option that has a LEASE value of zero. If the registration is to be permanently removed, KEY-LEASE should also be zero. Otherwise, it should have the same value it had previously; this holds the name in reserve for when the SRP client is once again able to provide the service.

SRP clients are normally expected to remove all service instances when removing a host. However, in some cases a SRP client may not have retained sufficient state to know that some service instance is pointing to a host that it is removing. This method of removing services is intended for the case where the client is going offline and does not want its services advertised. Therefore, it is sufficient for the client to send the Host Description Instruction (Section 2.3.1.3).

To support this, when removing services based on the lease time being zero, an SRP server MUST remove all service instances pointing to a host when a host is removed, even if the SRP client doesn't list them explicitly. If the key lease time is nonzero, the SRP server MUST NOT delete the KEY records for these SRP clients.

2.2.5.5.2. Removing some published services

In some use cases a client may need to remove some specific service, without removing its other services. This can be accomplished in one of two ways. To simply remove a specific service, the client sends a valid SRP Update where the Service Discovery Instruction (Section 2.3.1.1) contains a single Delete an RR from an RRset ([RFC2136], Section 2.5.4) update that deletes the PTR record whose target is the service instance name. The Service Description

Instruction (Section 2.3.1.2) in this case contains a single Delete all RRsets from a Name ([RFC2136], Section 2.5.3) update to the service instance name.

The second alternative is used when some service is being replaced by a different service with a different service instance name. In this case, the old service is deleted as in the first alternative. The new service is added, just as it would be in an update that wasn't deleting the old service. Because both the removal of the old service and the add of the new service consist of a valid Service Discovery Instruction and a valid Service Description Instruction, the update as a whole is a valid SRP Update, and will result in the old service being removed and the new one added, or, to put it differently, in the old service being replaced by the new service.

It is perhaps worth noting that if a service is being updated without the service instance name changing, that will look very much like the second alternative above. The difference is that because the target for the PTR record in the Service Discovery Instruction is the same for both the Delete An RR From An RRset update and the Add To An RRset update, these will be seen as a single Service Description Instruction, not as two Instructions. The same would be true of the Service Description Instruction.

Whichever of these two alternatives is used, the host lease will be updated with the lease time provided in the SRP update. In neither of these cases is it permissible to delete the host. All services must point to a host. If a host is to be deleted, this must be done using the method described in Section 2.2.5.5.1, which deletes the host and all services that have that host as their target.

2.3. Validation and Processing of SRP Updates

2.3.1. Validation of Adds and Deletes

The SRP server first validates that the DNS Update is a syntactically and semantically valid DNS Update according to the rules specified in RFC2136.

SRP Updates consist of a set of `_instructions_` that together add or remove one or more services. Each instruction consists of some combination of delete updates and add updates. When an instruction contains a delete and an add, the delete **MUST** precede the add.

The SRP server checks each instruction in the SRP Update to see that it is either a Service Discovery Instruction, a Service Description Instruction, or a Host Description Instruction. Order matters in DNS updates. Specifically, deletes must precede adds for records that

the deletes would affect; otherwise the add will have no effect. This is the only ordering constraint; aside from this constraint, updates may appear in whatever order is convenient when constructing the update.

Because the SRP Update is a DNS update, it MUST contain a single question that indicates the zone to be updated. Every delete and update in an SRP Update MUST be within the zone that is specified for the SRP Update.

2.3.1.1. Service Discovery Instruction

An instruction is a Service Discovery Instruction if it contains

- * exactly one "Add to an RRSet" or exactly one "Delete an RR from an RRSet" ([RFC2136], Section 2.5.1) RR update,
- * which updates a PTR RR,
- * the target of which is a Service Instance Name
- * for which name a Service Description Instruction is present in the SRP Update
- * if the Service Discovery Instruction is an "Add to an RRSet" instruction, the Service Description Instruction does not match if it does not contain an "Add to an RRset" update for the SRV RR describing that service.
- * if the Service Discovery Instruction is a "Delete an RR from an RRSet" update, the Service Description Instruction does not match if it contains an "Add to an RRset" update.
- * Service Discovery Instructions do not contain any other add or delete updates.

2.3.1.2. Service Description Instruction

An instruction is a Service Description Instruction if, for the appropriate Service Instance Name, it contains

- * exactly one "Delete all RRsets from a name" update for the service instance name ([RFC2136], Section 2.5.3),
- * zero or one "Add to an RRset" SRV RR,
- * zero or one "Add to an RRset" KEY RR that, if present, contains the public key corresponding to the private key that was used to sign the message (if present, the KEY MUST match the KEY RR given in the Host Description),
- * zero or more "Add to an RRset" TXT RRs,
- * If there is one "Add to an RRset" SRV update, there MUST be at least one "Add to an RRset" TXT update.
- * the target of the SRV RR Add, if present points to a hostname for which there is a Host Description Instruction in the SRP Update, or

- * if there is no "Add to an RRset" SRV RR, then either
 - the name to which the "Delete all RRsets from a name" applies does not exist, or
 - there is an existing KEY RR on that name, which matches the key with which the SRP Update was signed.
- * Service Descriptions Instructions do not modify any other resource records.

An SRP server MUST correctly handle compressed names in the SRV target.

2.3.1.3. Host Description Instruction

An instruction is a Host Description Instruction if, for the appropriate hostname, it contains

- * exactly one "Delete all RRsets from a name" RR,
- * one or more "Add to an RRset" RRs of type A and/or AAAA,
- * A and/or AAAA records must be of sufficient scope to be reachable by all hosts that might query the DNS. If a link-scope address or IPv4 autoconfiguration address is provided by the SRP client, the SRP server MUST treat this as if no address records were received; that is, the Host Description is not valid.
- * exactly one "Add to an RRset" RR that adds a KEY RR that contains the public key corresponding to the private key that was used to sign the message,
- * there is a Service Instance Name Instruction in the SRP Update for which the SRV RR that is added points to the hostname being updated by this update.
- * Host Description Instructions do not modify any other resource records.

2.3.2. Valid SRP Update Requirements

An SRP Update MUST include zero or more Service Discovery Instructions. For each Service Discovery Instruction, there MUST be at least one Service Description Instruction. Note that in the case of SRP subtypes (Section 7.1 of [RFC6763]), it's quite possible that two Service Discovery Instructions might reference the same Service Description Instruction. For each Service Description Instruction there MUST be at least one Service Discovery Instruction with its service instance name as the target of its PTR record. There MUST be exactly one Host Description Instruction. Every Service Description Instruction must have that Host Description Instruction as the target of its SRV record. A DNS Update that does not meet these constraints is not an SRP Update.

A DNS Update that contains any additional adds or deletes that cannot be identified as Service Discovery, Service Description or Host Description Instructions is not an SRP Update. A DNS update that contains any prerequisites is not an SRP Update. Such messages should either be processed as regular RFC2136 updates, including access control checks and constraint checks, if supported, or else rejected with RCODE=REFUSED.

In addition, in order for an update to be a valid SRP Update, the target of every Service Discovery Instruction MUST be a Service Description Instruction that is present in the SRP Update. There MUST NOT be any Service Description Instruction to which no Service Discovery Instruction points. The target of the SRV record in every Service Description Instruction MUST be the single Host Description Instruction.

If the definitions of each of these instructions are followed carefully and the update requirements are validated correctly, many DNS Updates that look very much like SRP Updates nevertheless will fail to validate. For example, a DNS update that contains an Add to an RRset instruction for a Service Name and an Add to an RRset instruction for a Service Instance Name, where the PTR record added to the Service Name does not reference the Service Instance Name, is not a valid SRP Update message, but may be a valid RFC2136 update.

2.3.3. FCFS Name And Signature Validation

Assuming that a DNS Update message has been validated with these conditions and is a valid SRP Update, the server checks that the name in the Host Description Instruction exists. If so, then the server checks to see if the KEY record on that name is the same as the KEY record in the Host Description Instruction. The server performs the same check for the KEY records in any Service Description Instructions. For KEY records that were omitted from Service Description Instructions, the KEY from the Host Description Instruction is used. If any existing KEY record corresponding to a KEY record in the SRP Update does not match the KEY record in the SRP Update (whether provided or taken from the Host Description Instruction), then the server MUST reject the SRP Update with the YXDOMAIN RCODE.

Otherwise, the server validates the SRP Update using SIG(0) against the public key in the KEY record of the Host Description Instruction. If the validation fails, the server MUST reject the SRP Update with the REFUSED RCODE. Otherwise, the SRP Update is considered valid and authentic, and is processed according to the method described in RFC2136.

KEY record updates omitted from Service Description Instruction are processed as if they had been explicitly present: every Service Description that is updated MUST, after the SRP Update has been applied, have a KEY RR, and it must be the same KEY RR that is present in the Host Description to which the Service Description refers.

2.3.4. Handling of Service Subtypes

SRP servers MUST treat the update instructions for a service type and all its subtypes as atomic. That is, when a service and its subtypes are being updated, whatever information appears in the SRP Update is the entirety of information about that service and its subtypes. If any subtype appeared in a previous update but does not appear in the current update, then the DNS server MUST remove that subtype.

Similarly, there is no mechanism for deleting subtypes. A delete of a service deletes all of its subtypes. To delete an individual subtype, an SRP Update must be constructed that contains the service type and all subtypes for that service.

2.3.5. SRP Update response

The status that is returned depends on the result of processing the update, and can be either SUCCESS or SERVFAIL: all other possible outcomes should already have been accounted for when applying the constraints that qualify the update as an SRP Update.

2.3.6. Optional Behavior

The server MAY add a Reverse Mapping that corresponds to the Host Description. This is not required because the Reverse Mapping serves no protocol function, but it may be useful for debugging, e.g. in annotating network packet traces or logs. In order for the server to add a reverse mapping update, it must be authoritative for the zone or have credentials to do the update. The SRP client MAY also do a reverse mapping update if it has credentials to do so.

The server MAY apply additional criteria when accepting updates. In some networks, it may be possible to do out-of-band registration of keys, and only accept updates from pre-registered keys. In this case, an update for a key that has not been registered should be rejected with the REFUSED RCODE.

There are at least two benefits to doing this rather than simply using normal SIG(0) DNS updates. First, the same registration protocol can be used in both cases, so both use cases can be addressed by the same service implementation. Second, the registration protocol includes maintenance functionality not present with normal DNS updates.

Note that the semantics of using SRP in this way are different than for typical RFC2136 implementations: the KEY used to sign the SRP Update only allows the SRP client to update records that refer to its Host Description. RFC2136 implementations do not normally provide a way to enforce a constraint of this type.

The server may also have a dictionary of names or name patterns that are not permitted. If such a list is used, updates for Service Instance Names that match entries in the dictionary are rejected with YXDOMAIN.

3. TTL Consistency

All RRs within an RRset are required to have the same TTL (Clarifications to the DNS Specification [RFC2181], Section 5.2). In order to avoid inconsistencies, SRP places restrictions on TTLs sent by services and requires that SRP servers enforce consistency.

Services sending SRP Updates MUST use consistent TTLs in all RRs within the SRP Update.

SRP servers MUST check that the TTLs for all RRs within the SRP Update are the same. If they are not, the SRP update MUST be rejected with a REFUSED RCODE.

Additionally, when adding RRs to an RRset, for example when processing Service Discovery records, the server MUST use the same TTL on all RRs in the RRset. How this consistency is enforced is up to the implementation.

TTLs sent in SRP Updates are advisory: they indicate the SRP client's guess as to what a good TTL would be. SRP servers may override these TTLs. SRP servers SHOULD ensure that TTLs are reasonable: neither too long nor too short. The TTL should never be longer than the lease time (Section 4.1). Shorter TTLs will result in more frequent data refreshes; this increases latency on the DNS-SD client side, increases load on any caching resolvers and on the authoritative server, and also increases network load, which may be an issue for constrained networks. Longer TTLs will increase the likelihood that data in caches will be stale. TTL minimums and maximums SHOULD be configurable by the operator of the SRP server.

4. Maintenance

4.1. Cleaning up stale data

Because the DNS-SD registration protocol is automatic, and not managed by humans, some additional bookkeeping is required. When an update is constructed by the SRP client, it **MUST** include an EDNS(0) Update Lease Option [I-D.sekar-dns-ul]. The Update Lease Option contains two lease times: the Lease Time and the Key Lease Time.

These leases are promises, similar to DHCP leases [RFC2131], from the SRP client that it will send a new update for the service registration before the lease time expires. The Lease time is chosen to represent the time after the update during which the registered records other than the KEY record should be assumed to be valid. The Key Lease time represents the time after the update during which the KEY record should be assumed to be valid.

The reasoning behind the different lease times is discussed in the section on first-come, first-served naming (Section 2.2.4.1). SRP servers may be configured with limits for these values. A default limit of two hours for the Lease and 14 days for the SIG(0) KEY are currently thought to be good choices. Constrained devices with limited battery that wake infrequently are likely to request longer leases; servers that support such devices may need to set higher limits. SRP clients that are going to continue to use names on which they hold leases should update well before the lease ends, in case the registration service is unavailable or under heavy load.

The lease time applies specifically to the host. All service instances, and all service entries for such service instances, depend on the host. When the lease on a host expires, the host and all services that reference it **MUST** be removed at the same time—it is never valid for a service instance to remain when the host it references has been removed. If the KEY record for the host is to remain, the KEY record for any services that reference it **MUST** also remain. However, the service PTR record **MUST** be removed, since it has no key associated with it, and since it is never valid to have a service PTR record for which there is no service instance on the target of the PTR record.

SRP Servers **SHOULD** also track a lease time per service instance. The reason for doing this is that a client may re-register a host with a different set of services, and not remember that some different service instance had previously been registered. In this case, when that service instance lease expires, the SRP server **SHOULD** remove the service instance (although the KEY record for the service instance **SHOULD** be retained until the key lease on that service expires).

This is beneficial because if the SRP client continues to renew the host, but never mentions the stale service again, the stale service will continue to be advertised.

The SRP server MUST include an EDNS(0) Update Lease option in the response if the lease time proposed by the service has been shortened or lengthened. The service MUST check for the EDNS(0) Update Lease option in the response and MUST use the lease times from that option in place of the options that it sent to the server when deciding when to update its registration. The times may be shorter or longer than those specified in the SRP Update; the SRP client must honor them in either case.

SRP clients should assume that each lease ends N seconds after the update was first transmitted, where N is the lease duration. Servers should assume that each lease ends N seconds after the update that was successfully processed was received. Because the server will always receive the update after the SRP client sent it, this avoids the possibility of misunderstandings.

SRP servers MUST reject updates that do not include an EDNS(0) Update Lease option. Dual-use servers MAY accept updates that don't include leases, but SHOULD differentiate between SRP Updates and other updates, and MUST reject updates that would otherwise be SRP Updates if they do not include leases.

Lease times have a completely different function than TTLs. On an authoritative DNS server, the TTL on a resource record is a constant: whenever that RR is served in a DNS response, the TTL value sent in the answer is the same. The lease time is never sent as a TTL; its sole purpose is to determine when the authoritative DNS server will delete stale records. It is not an error to send a DNS response with a TTL of 'n' when the remaining time on the lease is less than 'n'.

5. Security Considerations

5.1. Source Validation

SRP Updates have no authorization semantics other than first-come, first-served. This means that if an attacker from outside of the administrative domain of the server knows the server's IP address, it can in principle send updates to the server that will be processed successfully. Servers should therefore be configured to reject updates from source addresses outside of the administrative domain of the server.

For updates sent to an anycast IP address of an SRP server, this validation must be enforced by every router on the path from the Constrained-Device Network to the unconstrained portion of the network. For TCP updates, the initial SYN-SYN+ACK handshake prevents updates being forged by an off-network attacker. In order to ensure that this handshake happens, SRP servers relying on three-way-handshake validation **MUST NOT** accept TCP Fast Open payloads. If the network infrastructure allows it, an SRP server **MAY** accept TCP Fast Open payloads if all such packets are validated along the path, and the network is able to reject this type of spoofing at all ingress points.

Note that these rules only apply to the validation of SRP Updates. A server that accepts updates from SRP clients may also accept other DNS updates, and those DNS updates may be validated using different rules. However, in the case of a DNS service that accepts SRP updates, the intersection of the SRP Update rules and whatever other update rules are present must be considered very carefully.

For example, a normal, authenticated DNS update to any RR that was added using SRP, but that is authenticated using a different key, could be used to override a promise made by the SRP Server to an SRP client, by replacing all or part of the service registration information with information provided by an authenticated DNS update client. An implementation that allows both kinds of updates should not allow DNS Update clients that are using different authentication and authorization credentials to update records added by SRP clients.

5.2. SRP Server Authentication

This specification does not provide a mechanism for validating responses from DNS servers to SRP clients. In the case of Constrained Network/Constrained Node clients, such validation isn't practical because there's no way to establish trust. In principle, a KEY RR could be used by a non-constrained SRP client to validate responses from the server, but this is not required, nor do we specify a mechanism for determining which key to use.

5.3. Required Signature Algorithm

For validation, SRP servers **MUST** implement the ECDSA_{P256}SHA256 signature algorithm. SRP servers **SHOULD** implement the algorithms specified in [RFC8624], Section 3.1, in the validation column of the table, that are numbered 13 or higher and have a "MUST", "RECOMMENDED", or "MAY" designation in the validation column of the table. SRP clients **MUST NOT** assume that any algorithm numbered lower than 13 is available for use in validating SIG(0) signatures.

6. Privacy Considerations

Because DNSSD SRP Updates can be sent off-link, the privacy implications of SRP are different than for multicast DNS responses. Host implementations that are using TCP SHOULD also use TLS if available. Server implementations MUST offer TLS support. The use of TLS with DNS is described in [RFC7858] and [RFC8310].

Hosts that implement TLS support SHOULD NOT fall back to TCP; since servers are required to support TLS, it is entirely up to the host implementation whether to use it.

Public keys can be used as identifiers to track hosts. SRP servers MAY elect not to return KEY records for queries for SRP registrations.

7. Delegation of 'service.arpa.'

In order to be fully functional, the owner of the 'arpa.' zone must add a delegation of 'service.arpa.' in the '.arpa.' zone [RFC3172]. This delegation should be set up as was done for 'home.arpa', as a result of the specification in Section 7 of [RFC8375].

8. IANA Considerations

8.1. Registration and Delegation of 'service.arpa' as a Special-Use Domain Name

IANA is requested to record the domain name 'service.arpa.' in the Special-Use Domain Names registry [SUDN]. IANA is requested, with the approval of IAB, to implement the delegation requested in Section 7.

IANA is further requested to add a new entry to the "Transport-Independent Locally-Served Zones" subregistry of the the "Locally-Served DNS Zones" registry [LSDZ]. The entry will be for the domain 'service.arpa.' with the description "DNS-SD Registration Protocol Special-Use Domain", listing this document as the reference.

8.2. 'dnssd-srp' Service Name

IANA is also requested to add a new entry to the Service Names and Port Numbers registry for dnssd-srp with a transport type of tcp. No port number is to be assigned. The reference should be to this document, and the Assignee and Contact information should reference the authors of this document. The Description should be as follows:

Availability of DNS Service Discovery Service Registration Protocol Service for a given domain is advertised using the "_dnssd-srp._tcp.<domain>" SRV record gives the target host and port where DNSSD Service Registration Service is provided for the named domain.

8.3. 'dnssd-srp-tls' Service Name

IANA is also requested to add a new entry to the Service Names and Port Numbers registry for dnssd-srp with a transport type of tcp. No port number is to be assigned. The reference should be to this document, and the Assignee and Contact information should reference the authors of this document. The Description should be as follows:

Availability of DNS Service Discovery Service Registration Protocol Service for a given domain over TLS is advertised using the "_dnssd-srp-tls._tcp.<domain>." SRV record gives the target host and port where DNSSD Service Registration Service is provided for the named domain.

8.4. Anycast Address

IANA is requested to allocate an IPv6 Anycast address from the IPv6 Special-Purpose Address Registry, similar to the Port Control Protocol anycast address, 2001:1::1. The value TBD should be replaced with the actual allocation in the table that follows. The values for the registry are:

Attribute	value
Address Block	2001:1::TBD/128
Name	DNS-SD Service Registration Protocol Anycast Address
RFC	[this document]
Allocation Date	[date of allocation]
Termination Date	N/A
Source	True
Destination	True
Forwardable	True
Global	True
Reserved-by-protocol	False

Table 1

9. Implementation Status

[Note to the RFC Editor: please remove this section prior to publication.]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 7942. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to RFC 7942, "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation

and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

There are two known independent implementations of SRP clients:

- * SRP Client for OpenThread:
<https://github.com/openthread/openthread/pull/6038>
- * mDNSResponder open source project: <https://github.com/Abhayakara/mdnsresponder>

There are two related implementations of an SRP server. One acts as a DNS Update proxy, taking an SRP Update and applying it to the specified DNS zone using DNS update. The other acts as an Advertising Proxy [I-D.sctl-advertising-proxy]. Both are included in the mDNSResponder open source project mentioned above.

10. Acknowledgments

Thanks to Toke Høiland-Jørgensen, Jonathan Hui, Esko Dijk, Kangping Dong and Abtin Keshavarzian for their thorough technical reviews. Thanks to Kangping and Abtin as well for testing the document by doing an independent implementation. Thanks to Tamara Kemper for doing a nice developmental edit, Tim Wattenberg for doing a SRP client proof-of-concept implementation at the Montreal Hackathon at IETF 102, and Tom Pusateri for reviewing during the hackathon and afterwards.

11. Normative References

- [I-D.sekar-dns-ul]
Cheshire, S. and T. Lemon, "An EDNS0 option to negotiate Leases on DNS Updates", Work in Progress, Internet-Draft, draft-sekar-dns-ul-03, 27 July 2021, <<https://datatracker.ietf.org/doc/html/draft-sekar-dns-ul-03>>.
- [RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, DOI 10.17487/RFC2132, March 1997, <<https://www.rfc-editor.org/info/rfc2132>>.
- [RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, DOI 10.17487/RFC2136, April 1997, <<https://www.rfc-editor.org/info/rfc2136>>.

- [RFC2539] Eastlake 3rd, D., "Storage of Diffie-Hellman Keys in the Domain Name System (DNS)", RFC 2539, DOI 10.17487/RFC2539, March 1999, <<https://www.rfc-editor.org/info/rfc2539>>.
- [RFC2931] Eastlake 3rd, D., "DNS Request and Transaction Signatures (SIG(0)s)", RFC 2931, DOI 10.17487/RFC2931, September 2000, <<https://www.rfc-editor.org/info/rfc2931>>.
- [RFC3172] Huston, G., Ed., "Management Guidelines & Operational Requirements for the Address and Routing Parameter Area Domain ("arpa")", BCP 52, RFC 3172, DOI 10.17487/RFC3172, September 2001, <<https://www.rfc-editor.org/info/rfc3172>>.
- [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>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC8106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 8106, DOI 10.17487/RFC8106, March 2017, <<https://www.rfc-editor.org/info/rfc8106>>.
- [RFC8375] Pfister, P. and T. Lemon, "Special-Use Domain 'home.arpa.'", RFC 8375, DOI 10.17487/RFC8375, May 2018, <<https://www.rfc-editor.org/info/rfc8375>>.
- [RFC8624] Wouters, P. and O. Sury, "Algorithm Implementation Requirements and Usage Guidance for DNSSEC", RFC 8624, DOI 10.17487/RFC8624, June 2019, <<https://www.rfc-editor.org/info/rfc8624>>.
- [RFC8765] Pusateri, T. and S. Cheshire, "DNS Push Notifications", RFC 8765, DOI 10.17487/RFC8765, June 2020, <<https://www.rfc-editor.org/info/rfc8765>>.
- [SUDN] "Special-Use Domain Names Registry", July 2012, <<https://www.iana.org/assignments/special-use-domain-names/special-use-domain-names.xhtml>>.
- [LSDZ] "Locally-Served DNS Zones Registry", July 2011, <<https://www.iana.org/assignments/locally-served-dns-zones/locally-served-dns-zones.xhtml>>.

12. Informative References

- [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>>.
- [RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <<https://www.rfc-editor.org/info/rfc2131>>.
- [RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, DOI 10.17487/RFC2181, July 1997, <<https://www.rfc-editor.org/info/rfc2181>>.
- [RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, DOI 10.17487/RFC2782, February 2000, <<https://www.rfc-editor.org/info/rfc2782>>.
- [RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, DOI 10.17487/RFC3007, November 2000, <<https://www.rfc-editor.org/info/rfc3007>>.
- [RFC6760] Cheshire, S. and M. Krochmal, "Requirements for a Protocol to Replace the AppleTalk Name Binding Protocol (NBP)", RFC 6760, DOI 10.17487/RFC6760, February 2013, <<https://www.rfc-editor.org/info/rfc6760>>.
- [RFC6761] Cheshire, S. and M. Krochmal, "Special-Use Domain Names", RFC 6761, DOI 10.17487/RFC6761, February 2013, <<https://www.rfc-editor.org/info/rfc6761>>.
- [RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <<https://www.rfc-editor.org/info/rfc6762>>.
- [RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <<https://www.rfc-editor.org/info/rfc7228>>.
- [RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <<https://www.rfc-editor.org/info/rfc8310>>.

- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.
- [RFC8766] Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", RFC 8766, DOI 10.17487/RFC8766, June 2020, <<https://www.rfc-editor.org/info/rfc8766>>.
- [I-D.cheshire-dnssd-roadmap]
Cheshire, S., "Service Discovery Road Map", Work in Progress, Internet-Draft, draft-cheshire-dnssd-roadmap-03, 23 October 2018, <<https://datatracker.ietf.org/doc/html/draft-cheshire-dnssd-roadmap-03>>.
- [I-D.cheshire-edns0-owner-option]
Cheshire, S. and M. Krochmal, "EDNS0 OWNER Option", Work in Progress, Internet-Draft, draft-cheshire-edns0-owner-option-01, 3 July 2017, <<https://datatracker.ietf.org/doc/html/draft-cheshire-edns0-owner-option-01>>.
- [I-D.sctl-advertising-proxy]
Cheshire, S. and T. Lemon, "Advertising Proxy for DNS-SD Service Registration Protocol", Work in Progress, Internet-Draft, draft-sctl-advertising-proxy-02, 12 July 2021, <<https://datatracker.ietf.org/doc/html/draft-sctl-advertising-proxy-02>>.
- [ZC] Cheshire, S. and D.H. Steinberg, "Zero Configuration Networking: The Definitive Guide", O'Reilly Media, Inc. , ISBN 0-596-10100-7, December 2005.

Appendix A. Testing using standard RFC2136-compliant servers

It may be useful to set up a DNS server for testing that does not implement SRP. This can be done by configuring the server to listen on the anycast address, or advertising it in the `_dnssd-srp._tcp.<zone>` SRV and `_dnssd-srp-tls._tcp.<zone>` record. It must be configured to be authoritative for "default.service.arpa", and to accept updates from hosts on local networks for names under "default.service.arpa" without authentication, since such servers will not have support for FCFS authentication (Section 2.2.4.1).

A server configured in this way will be able to successfully accept and process SRP Updates from services that send SRP updates. However, no prerequisites will be applied, and this means that the

test server will accept internally inconsistent SRP Updates, and will not stop two SRP Updates, sent by different services, that claim the same name(s), from overwriting each other.

Since SRP Updates are signed with keys, validation of the SIG(0) algorithm used by the client can be done by manually installing the client public key on the DNS server that will be receiving the updates. The key can then be used to authenticate the client, and can be used as a requirement for the update. An example configuration for testing SRP using BIND 9 is given in Appendix C.

Appendix B. How to allow services to update standard RFC2136-compliant servers

Ordinarily SRP Updates will fail when sent to an RFC 2136-compliant server that does not implement SRP because the zone being updated is "default.service.arpa", and no DNS server that is not an SRP server should normally be configured to be authoritative for "default.service.arpa". Therefore, a service that sends an SRP Update can tell that the receiving server does not support SRP, but does support RFC2136, because the RCODE will either be NOTZONE, NOTAUTH or REFUSED, or because there is no response to the update request (when using the anycast address)

In this case a service MAY attempt to register itself using regular RFC2136 DNS updates. To do so, it must discover the default registration zone and the DNS server designated to receive updates for that zone, as described earlier, using the `_dns-update._udp` SRV record. It can then make the update using the port and host pointed to by the SRV record, and should use appropriate prerequisites to avoid overwriting competing records. Such updates are out of scope for SRP, and a service that implements SRP MUST first attempt to use SRP to register itself, and should only attempt to use RFC2136 backwards compatibility if that fails. Although the owner name for the SRV record specifies the UDP protocol for updates, it is also possible to use TCP, and TCP should be required to prevent spoofing.

Appendix C. Sample BIND9 configuration for default.service.arpa.

```
zone "default.service.arpa." {  
    type master;  
    file "/etc/bind/master/service.db";  
    allow-update { key demo.default.service.arpa.; };  
};
```

Figure 1: Zone Configuration in named.conf

```

$ORIGIN .
$TTL 57600 ; 16 hours
default.service.arpa IN SOA      ns3.default.service.arpa.
                                postmaster.default.service.arpa. (
                                2951053287 ; serial
                                3600      ; refresh (1 hour)
                                1800      ; retry (30 minutes)
                                604800    ; expire (1 week)
                                3600      ; minimum (1 hour)
                                )
                                NS       ns3.default.service.arpa.
                                SRV 0 0 53 ns3.default.service.arpa.
$ORIGIN default.service.arpa.
$TTL 3600 ; 1 hour
_ipps._tcp PTR demo._ipps._tcp
$ORIGIN _ipps._tcp.default.service.arpa.
demo TXT "0"
SRV 0 0 9992 demo.default.service.arpa.
$ORIGIN _udp.default.service.arpa.
$TTL 3600 ; 1 hour
_dns-update PTR ns3.default.service.arpa.
$ORIGIN _tcp.default.service.arpa.
_dnssd-srp PTR ns3.default.service.arpa.
$ORIGIN default.service.arpa.
$TTL 300 ; 5 minutes
ns3 AAAA 2001:db8:0:1::1
$TTL 3600 ; 1 hour
demo AAAA 2001:db8:0:2::1
KEY 513 3 13 (
    qweEmaaQ0FAWok5//ftuQtZgiZoiFSUsm0srWREdywQU
    9dpvtOhrdKWUuPT3uEFF5TZU6B4q1z1I662GdaUwqg==
); alg = ECDSA256SHA256 ; key id = 15008
AAAA ::1

```

Figure 2: Example Zone file

Authors' Addresses

Ted Lemon
 Apple Inc.
 One Apple Park Way
 Cupertino, California 95014
 United States of America
 Email: mellon@fugue.com

Stuart Cheshire
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America
Phone: +1 408 974 3207
Email: cheshire@apple.com

Homenet
Internet-Draft
Intended status: Standards Track
Expires: May 6, 2021

D. Migault
Ericsson
R. Weber
Nominum
M. Richardson
Sandelman Software Works
R. Hunter
Globis Consulting BV
C. Griffiths

W. Cloetens
Deutsche Telekom
November 02, 2020

Simple Provisioning of Public Names for Residential Networks
draft-ietf-homenet-front-end-naming-delegation-12

Abstract

Home owners often have IPv6 devices that they wish to access over the Internet using names. It has been possible to register and populate a DNS Zone with names since DNS became a thing, but it has been an activity typically reserved for experts. This document automates the process through creation of a Homenet Naming Authority, whose responsibility is to select, sign and publish names to a set of publically visible servers.

The use of an outsourced primary DNS server deals with possible renumbering of the home network, and with possible denial of service attacks against the DNS infrastructure.

This document describes the mechanism that enables the Home Network Authority (HNA) to outsource the naming service to the DNS Outsourcing Infrastructure via a Distribution Master (DM).

In addition, this document deals with publication of a corresponding reverse zone.

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 <https://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 May 6, 2021.

Copyright Notice

Copyright (c) 2020 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 (<https://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
1.1. Selecting Names to Publish	6
1.2. Alternative solutions	6
2. Terminology	7
3. Architecture Description	9
3.1. Architecture Overview	9
3.2. Distribution Master Communication Channels	11
4. Control Channel between Homenet Naming Authority (HNA) and Distribution Master (DM)	13
4.1. Information to build the Public Homenet Zone.	13
4.2. Information to build the DNSSEC chain of trust	13
4.3. Information to set the Synchronization Channel	14
4.4. Deleting the delegation	14
4.5. Messages Exchange Description	14
4.5.1. Retrieving information for the Public Homenet Zone.	15
4.5.2. Providing information for the DNSSEC chain of trust	16
4.5.3. Providing information for the Synchronization Channel	16
4.5.4. HNA instructing deleting the delegation	17
4.6. Securing the Control Channel between Homenet Naming Authority (HNA) and Distribution Master (DM)	17
4.7. Implementation Concerns	18
5. DM Synchronization Channel between HNA and DM	19
5.1. Securing the Synchronization Channel between HNA and DM	20
6. DM Distribution Channel	20

7. HNA Security Policies	21
8. DNSSEC compliant Homenet Architecture	21
9. Homenet Reverse Zone Channels Configuration	21
10. Homenet Public Zone Channel Configurations	22
11. Renumbering	23
11.1. Hidden Primary	24
11.2. Distribution Master	25
12. Privacy Considerations	26
13. Security Considerations	27
13.1. HNA DMand RDM channels	27
13.2. Names are less secure than IP addresses	27
13.3. Names are less volatile than IP addresses	27
13.4. DNS Reflection Attacks	28
13.5. Reflection Attack involving the Hidden Primary	28
13.6. Reflection Attacks involving the DM	30
13.7. Reflection Attacks involving the Public Authoritative Servers	30
13.8. Flooding Attack	31
13.9. Replay Attack	31
14. Data Model for Outsourced information	32
15. IANA Considerations	32
16. Acknowledgment	32
17. References	32
17.1. Normative References	33
17.2. Informative References	36
Appendix A. Envisioned deployment scenarios	38
A.1. CPE Vendor	38
A.2. Agnostic CPE	38
Appendix B. Example: Homenet Zone	39
Appendix C. Example: HNA necessary parameters for outsourcing .	41
Appendix D. Example: A manufacturer provisioned HNA product flow	42
Authors' Addresses	43

1. Introduction

The Homenet Naming Authority (HNA) is responsible for making devices within the home network accessible by name within the home network as well as from outside the home network (e.g. the Internet). IPv6 connectivity provides the possibility of global end to end IP connectivity. End users will be able to transparently make use of this connectivity if they can use names to access the services they want from their home network.

The use of a DNS zone for each home network is a reasonable and scalable way to make the set of public names visible. There are a number of ways to populate such a zone. This specification proposes a way based on a number of assumptions about typical home networks.

1. The names of the devices accessible from the Internet are stored in the Public Homenet Zone, served by a DNS authoritative server.
2. It is unlikely that home networks will contain sufficiently robust platforms designed to host a service such as the DNS on the Internet and as such would expose the home network to DDoS attacks.
3. [RFC7368] emphasizes that the home network is subject to connectivity disruptions with the ISP. But, names used within the home MUST be resilient against such disruption.

This specification makes the public names resolvable within both the home network and on the Internet, even when there are disruptions.

This is achieved by having a device inside the home network that builds, signs, publishes, and manages a Public Homenet Zone, thus providing bindings between public names, IP addresses, and other RR types.

The management of the names can be a role that the Customer Premises Equipment (CPE) does. Other devices in the home network could fulfill this role e.g. a NAS server, but for simplicity, this document assumes the function is located on one of the CPE devices.

The homenet architecture [RFC7368] makes it clear that a home network may have multiple CPEs. The management of the Public Homenet Zone involves DNS specific mechanisms that cannot be distributed over multiple servers (primary server), when multiple nodes can potentially manage the Public Homenet Zone, a single node needs to be selected per outsourced zone. This selected node is designated as providing the Homenet Naming Authority (HNA) function.

The process by which a single HNA is selected per zone is not in scope for this document. It is envisioned that a future document will describe an HNCP mechanism to elect the single HNA.

CPEs, which may host the HNA function, as well as home network devices, are usually low powered devices not designed for terminating heavy traffic. As a result, hosting an authoritative DNS service visible to the Internet may expose the home network to resource exhaustion and other attacks. On the other hand, if the only copy of the public zone is on the Internet, then Internet connectivity disruptions would make the names unavailable inside the homenet.

In order to avoid resource exhaustion and other attacks, this document describes an architecture that outsources the authoritative naming service of the home network. More specifically, the HNA

builds the Public Homenet Zone and outsources it to an DNS Outsourcing Infrastructure (DOI) via a Distribution Master (DM). The DNS Outsourcing Infrastructure (DOI) is in charge of publishing the corresponding Public Homenet Zone on the Internet. The transfer of DNS zone information is achieved using standard DNS mechanisms involving primary and secondary DNS servers, with the HNA hosted primary being a stealth primary, and the Distribution Master a secondary.

Section 3.1 provides an architecture description that describes the relation between the HNA and the Outsourcing Architecture. In order to keep the Public Homenet Zone up-to-date Section 5 describes how the HNA and the DNS Outsourcing Infrastructure synchronizes the Public Homenet Zone.

The proposed architecture is explicitly designed to enable fully functional DNSSEC, and the Public Homenet Zone is expected to be signed with a secure delegation. DNSSEC key management and zone signing is handled by the HNA.

Section 10 discusses management and configuration of the Public Homenet Zone. It shows that the HNA configuration of the Outsourcing infrastructure can involve no or little interaction with the end user. More specifically, it shows that the existence of an account in the DOI is sufficient for the DOI to push the necessary configuration.

Section 9 discusses management of one or more reverse zones. It shows that management of the reverse zones can be entirely automated and benefit from a pre-established relation between the ISP and the home network. Note that such scenarios may also be met for the Public Homenet Zone, but not necessarily.

Section 11 discusses how renumbering should be handled. Finally, Section 12 and Section 13 respectively discuss privacy and security considerations when outsourcing the Public Homenet Zone.

The Public Homenet Zone is expected to contain public information only in a single universal view. This document does not define how the information required to construct this view is derived.

It is also not in the scope of this document to define names for exclusive use within the boundaries of the local home network. Instead, local scope information is expected to be provided to the home network using local scope naming services. mDNS [RFC6762] DNS-SD [RFC6763] are two examples of these services. Currently mDNS is limited to a single link network. However, future protocols and

architectures [I-D.ietf-homenet-simple-naming] are expected to leverage this constraint as pointed out in [RFC7558].

1.1. Selecting Names to Publish

While this document does not create any normative mechanism by which the selection of names to publish, this document anticipates that the home network administrator (a human), will be presented with a list of current names and addresses present on the inside of the home network.

The administrator would mark which devices (by name), are to be published. The HNA would then collect the IPv6 address(es) associated with that device, and put the name into the Public Homenet Zone. The address of the device can be collected from a number of places: mDNS [RFC6762], DHCP [RFC6644], UPnP, PCP [RFC6887], or manual configuration.

A device may have a Global Unicast Address (GUA), a Unique Local IPv6 Address (ULA), as well IPv6-Link-Local addresses, IPv4-Link-Local Addresses, and RFC1918 addresses. Of these the link-local are never useful for the Public Zone, and should be omitted. The IPv6 ULA and the RFC1918 addresses may be useful to publish, if the home network environment features a VPN that would allow the home owner to reach the network.

The IPv6 ULA addressees are significantly safer to publish, as the RFC1918 addressees are likely to be confusing to any other entity.

In general, one expects the GUA to be the default address to be published. However, during periods when the home network has connectivity problems, the ULA and RFC1918 addressees can be used inside the home, and the mapping from public name to locally useful location address would permit many services secured with HTTPS to continue to operate.

1.2. Alternative solutions

An alternative existing solution in IPv4 is to have a single zone, where a host uses a RESTful HTTP service to register a single name into a common public zone. This is often called "Dynamic DNS", and there are a number of commercial providers, including Dyn, Gandi etc. These solutions were typically used by a host behind the CPE to make its CPE IPv4 address visible, usually in order to enable incoming connections.

For a small number (one to three) of hosts, use of such a system provides an alternative to the architecture described in this document.

The alternative does suffer from some severe limitations:

- o the CPE/HNA router is unaware of the process, and cannot respond to queries for these names when there are disruptions in connectivity. This makes the home user or application dependent on having to resolve different names in the event of outages or disruptions.
- o the CPE/HNA router cannot control the process. Any host can do this regardless of whether or not the home network administrator wants the name published or not. There is therefore no possible audit trail.
- o the credentials for the dynamic DNS server need to be securely transferred to all hosts that wish to use it. This is not a problem for a technical user to do with one or two hosts, but it does not scale to multiple hosts and becomes a problem for non-technical users.
- o "all the good names are taken" - current services put everyone's names into some small set of zones, and there are often conflicts. Distinguishing similar names by delegation of zones was among the primary design goals of the DNS system.
- o The RESTful services do not always support all RR types. The homenet user is dependent on the service provider supporting new types. By providing full DNS delegation, this document enables all RR types and also future extensions.

There is no technical reason why a RESTful cloud service could not provide solutions to many of these problems, but this document describes a DNS based solution.

2. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Customer Premises Equipment: (CPE) is a router providing connectivity to the home network.

Homenet Zone: is the DNS zone for use within the boundaries of the home network: home.arpa, see [RFC8375]). This zone is not considered public and is out of scope for this document.

Registered Homenet Domain: is the Domain Name associated with the home network.

Public Homenet Zone: contains the names in the home network that are expected to be publicly resolvable on the Internet.

Homenet Naming Authority: (HNA) is a function responsible for managing the Public Homenet Zone. This includes populating the Public Homenet Zone, signing the zone for DNSSEC, as well as managing the distribution of that Homenet Zone to the Outsourcing Infrastructure.

DNS Outsourcing Infrastructure (DOI): is the infrastructure responsible for receiving the Public Homenet Zone and publishing it on the Internet. It is mainly composed of a Distribution Master and Public Authoritative Servers.

Public Authoritative Servers: are the authoritative name servers for the Public Homenet Zone. Name resolution requests for the Homenet Domain are sent to these servers. For resiliency the Public Homenet Zone SHOULD be hosted on multiple servers.

Homenet Authoritative Servers: are authoritative name servers within the Homenet network.

Distribution Master (DM): is the (set of) server(s) to which the HNA synchronizes the Public Homenet Zone, and which then distributes the relevant information to the Public Authoritative Servers.

Homenet Reverse Zone: The reverse zone file associated with the Public Homenet Zone.

Reverse Public Authoritative Servers: equivalent to Public Authoritative Servers specifically for reverse resolution.

Reverse Distribution Master: equivalent to Distribution Master specifically for reverse resolution.

Homenet DNSSEC Resolver: a resolver that performs a DNSSEC resolution on the home network for the Public Homenet Zone. The resolution is performed requesting the Homenet Authoritative Servers.

DNSSEC Resolver: a resolver that performs a DNSSEC resolution on the Internet for the Public Homenet Zone. The resolution is performed requesting the Public Authoritative Servers.

3. Architecture Description

This section provides an overview of the architecture for outsourcing the authoritative naming service from the HNA to the DNS Outsourcing Infrastructure in Section 3.1. Section Appendix B and Appendix C illustrates this architecture with the example of a Public Homenet Zone as well as necessary parameter to configure the HNA.

3.1. Architecture Overview

Figure 1 illustrates the architecture where the HNA outsources the publication of the Public Homenet Zone to the DNS Outsourcing Infrastructure (DOI).

The Public Homenet Zone is identified by the Registered Homenet Domain Name - myhome.isp.example.

The ".local" as well as ".home.arpa" are explicitly not considered as Public Homenet zones.

The HNA SHOULD build the Public Homenet Zone in a single view populated with all resource records that are expected to be published on the Internet.

As explained in {#selectingnames}, how the Public Homenet Zone is populated is out of the scope of this document.

The HNA also signs the Public Homenet Zone. The HNA handles all operations and keying material required for DNSSEC, so there is no provision made in this architecture for transferring private DNSSEC related keying material between the HNA and the DM.

Once the Public Homenet Zone has been built, the HNA outsources it to the DNS Outsourcing Infrastructure as described in Figure 1.

The HNA acts as a hidden primary while the DM behaves as a secondary responsible to distribute the Public Homenet Zone to the multiple Public Authoritative Servers that DNS Outsourcing Infrastructure is responsible for.

The DM has 3 communication channels:

- o a DM Control Channel (see section Section 4) to configure the HNA and the Outsourcing Infrastructure,

- o a DM Synchronization Channel (see section Section 5 to synchronize the Public Homenet Zone on the HNA and on the DM.
- o one or more Distribution Channels (see section Section 6 that distributes the Public Homenet Zone from the DM to the Public Authoritative Server serving the Public Homenet Zone on the Internet.

There MAY be multiple DM's, and multiple servers per DM. This text assumes a single DM server for simplicity, but there is no reason why each channel need to be implemented on the same server, or indeed use the same code base.

It is important to note that while the HNA is configured as an authoritative server, it is not expected to answer to DNS requests from the public Internet for the Public Homenet Zone. The function of the HNA is limited to building the zone and synchronization with the DM.

The addresses associated with the HNA SHOULD NOT be mentioned in the NS records of the Public Homenet zone, unless additional security provisions necessary to protect the HNA from external attack have been taken.

The DNS Outsourcing Infrastructure is also responsible for ensuring the DS record has been updated in the parent zone.

Resolution is performed by the DNSSEC resolvers. When the resolution is performed outside the home network, the DNSSEC Resolver resolves the DS record on the Global DNS and the name associated to the Public Homenet Zone (example.com) on the Public Authoritative Servers.

When the resolution is performed from within the home network, the Homenet DNSSEC Resolver may proceed similarly. On the other hand, to provide resilience to the Public Homenet Zone in case of disruption, the Homenet DNSSEC Resolver SHOULD be able to perform the resolution on the authoritative name service of the home network implemented by the Homenet Authoritative Servers. These servers are not expected to be mentioned in the Public Homenet Zone, nor to be accessible from the Internet. As such their information as well as the corresponding signed DS record MAY be provided by the HNA to the Homenet DNSSEC Resolvers, e.g., using HNCP. Such configuration is outside the scope of this document.

How the Homenet Authoritative Servers are provisioned is also out of scope of this specification. It could be implemented using primary secondaries servers, or via rsync. In some cases, the HNA and Homenet Authoritative Servers may be combined together which would

result in a common instantiation of an authoritative server on the WAN and inner interface. Other mechanisms may also be used.

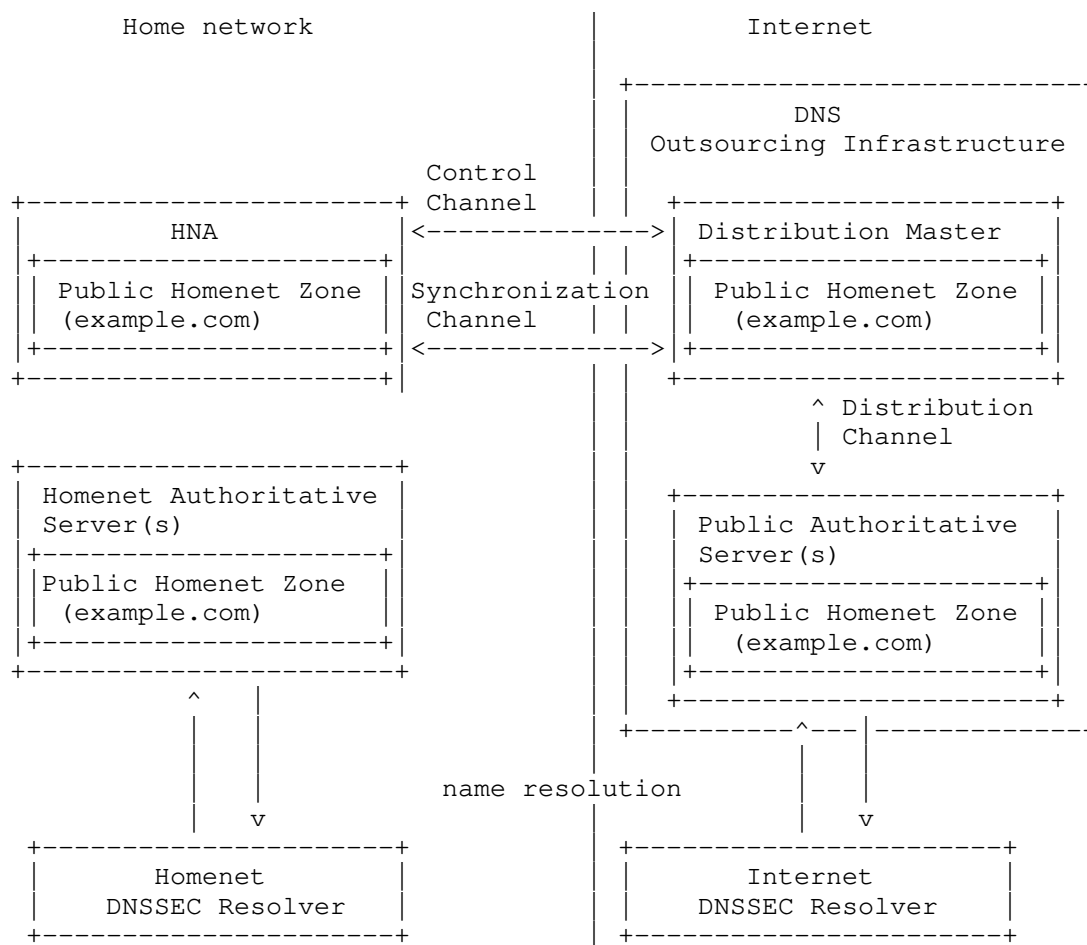


Figure 1: Homenet Naming Architecture Name Resolution

3.2. Distribution Master Communication Channels

This section details the interfaces and channels of the DM, that is the Control Channel, the Synchronization Channel and the Distribution Channel.

The Control Channel and the Synchronization Channel are the interfaces used between the HNA and the DNS Outsourcing Infrastructure. The entity within the DNS Outsourcing Infrastructure responsible to handle these communications is the DM and

communications between the HNA and the DM SHOULD be protected and mutually authenticated. While section Section 4.6 discusses in more depth the different security protocols that could be used to secure, this specification RECOMMENDS the use of TLS with mutually authentication based on certificates to secure the channel between the HNA and the DM.

The Control Channel is used to set up the Synchronization Channel. We assume that the HNA initiates the Control Channel connection with the DM and as such has a prior knowledge of the DM identity (X509 certificate), the IP address and port to use and protocol to set secure session. We also assume the DM has knowledge of the identity of the HNA (X509 certificate) as well as the Registered Homenet Domain. For more detail to see how this can be achieved, please see section Section 10.

The information exchanged between the HNA and the DM is using DNS messages. DNS messages can be protected using various kind of transport layers, among others, UDP:53/DTLS, TLS/TCP:53, HTTPS:443.

There was consideration to using a standard TSIG [RFC2845] or SIG(0) [RFC2931] to perform a dynamic DNS update to the DM. There are a number of issues with this.

The main one is that the Dynamic DNS update would also update the parent zone's (NS, DS and associated A or AAAA records) while the goal is to update the Distribution Master's configuration files. The visible NS records SHOULD remain pointing at the cloud provider's anycast addresses. Revealing the address of the HNA in the DNS is not desirable. Please see section Section 4.2 for more details.

This specification also assumes:

- o the DM serves both the Control Channel and Synchronization Channel on a single IP address, single port and with a single transport protocol.
- o the HNA uses a single IP address for both the Control and Synchronization channel by default. However, the HNA MAY use distinct IP addresses - see section Section 5 and section {sec-sync-info} for more details.

The Distribution Channel is internal to the DNS Outsourcing Infrastructure and as such is not the primary concern of this specification.

4. Control Channel between Homenet Naming Authority (HNA) and Distribution Master (DM)

The DM Control Channel is used by the HNA and the DNS Outsourcing Infrastructure to exchange information related to the configuration of the delegation which includes:

4.1. Information to build the Public Homenet Zone.

When the HNA builds the public zone, it must include information that it retrieves from the DM relating to how the zone is to be published.

The information includes at least names and IP addresses of the Public Authoritative Name Servers. In term of RRset information this includes:

- o the MNAME of the SOA,
- o the NS and associated A and AAA RRsets of the name servers.

Optionally the DNS Outsourcing Infrastructure MAY also provide operational parameters such as other fields of SOA (SERIAL, RNAME, REFRESH, RETRY, EXPIRE and MINIMUM). As the information is necessary for the HNA to proceed and the information is associated to the Outsourcing Infrastructure, this information exchange is mandatory.

4.2. Information to build the DNSSEC chain of trust

The HNA SHOULD provide the hash of the KSK (DS RRset), so the that DNS Outsourcing Infrastructure provides this value to the parent zone. A common deployment use case is that the Outsourcing Infrastructure is the registrar of the Registered Homenet Domain, and as such, its relationship with the registry of the parent zone enables it to update the parent zone. When such relation exists, the HNA should be able to request the DNS Outsourcing Infrastructure to update the DS RRset in the parent zone. A direct update is especially necessary to initialize the chain of trust.

Though the HNA may also later directly update the values of the DS via the Control Channel, it is RECOMMENDED to use other mechanisms such as CDS and CDNSKEY [RFC7344] for transparent updates during key roll overs.

As some deployment may not provide an DNS Outsourcing Infrastructure that will be able to update the DS in the parent zone, this information exchange is OPTIONAL.

By accepting the DS RR, the DM commits in taking care of advertising the DS to the parent zone. Upon refusal, the DM clearly indicates it does not have the capacity to proceed to the update.

4.3. Information to set the Synchronization Channel

That information sets the primary/secondary relation between the HNA and the DM. The HNA works as a primary authoritative DNS server, and MUST provide the corresponding IP address.

The specified IP address on the HNA side and the currently used IP address of the DM defines the IP addresses involved in the Synchronization Channel. Ports and transport protocol are the same as those used by the Control Channel. By default, the same IP address used by the HNA is considered by the DM. Exchange of this information is OPTIONAL.

4.4. Deleting the delegation

The purpose of the previous sections were to exchange information in order to set a delegation. The HNA MUST also be able to delete a delegation with a specific DM. Upon an instruction of deleting the delegation, the DM MUST stop serving the Public Homenet Zone.

4.5. Messages Exchange Description

There are multiple ways these information could be exchanged between the HNA and the DM. This specification defines a mechanism that re-use the DNS exchanges format. The intention is to reuse standard libraries especially to check the format of the exchanged fields as well as to minimize the additional libraries needed for the HNA. The re-use of DNS exchanges achieves these goals. Note that while information is provided using DNS exchanges, the exchanged information is not expected to be set in any zone file, instead this information is expected to be processed appropriately.

The Control Channel is not expected to be a long term session. After a predefined timer the Control Channel is expected to be terminated. The Control Channel MAY Be re-opened at any time later.

The provisioning process SHOULD provide a method of securing the control channel, so that the content of messages can be authenticated. This authentication MAY be based on certificates for both the DM and each HNA. The DM may also create the initial configuration for the delegation zone in the parent zone during the provisioning process.

4.5.1. Retrieving information for the Public Homenet Zone.

The information provided by the DM to the HNA is retrieved by the HNA with an AXFR exchange. The AXFR message enables the response to contain any type of RRsets. The response might be extended in the future if additional information will be needed. Alternatively, the information provided by the HNA to the DM is pushed by the HNA via a DNS update exchange.

To retrieve the necessary information to build the Public Homenet Zone, the HNA MUST send an DNS request of type AXFR associated to the Registered Homenet Domain. The DM MUST respond with a zone template. The zone template MUST contain a RRset of type SOA, one or multiple RRset of type NS and zero or more RRset of type A or AAAA.

- o The SOA RR is used to indicate to the HNA the value of the MNAME of the Public Homenet Zone.
- o The NAME of the SOA RR MUST be the Registered Homenet Domain.
- o The MNAME value of the SOA RDATA is the value provided by the DNS Outsourcing Infrastructure to the HNA.
- o Other RDATA values (RNAME, REFRESH, RETRY, EXPIRE and MINIMUM) are provided by the DNS Outsourcing Infrastructure as suggestions.

The NS RRsets are used to carry the Public Authoritative Servers of the DNS Outsourcing Infrastructure. Their associated NAME MUST be the Registered Homenet Domain.

The TTL and RDATA are those expected to be published on the Public Homenet Zone. The RRsets of Type A and AAAA MUST have their NAME matching the NSDNAME of one of the NS RRsets.

Upon receiving the response, the HNA MUST validate format and properties of the SOA, NS and A or AAAA RRsets. If an error occurs, the HNA MUST stop proceeding and MUST report an error. Otherwise, the HNA builds the Public Homenet Zone by setting the MNAME value of the SOA as indicated by the SOA provided by the AXFR response. The HNA SHOULD set the value of NAME, REFRESH, RETRY, EXPIRE and MINIMUM of the SOA to those provided by the AXFR response. The HNA MUST insert the NS and corresponding A or AAAA RRset in its Public Homenet Zone. The HNA MUST ignore other RRsets. If an error message is returned by the DM, the HNA MUST proceed as a regular DNS resolution. Error messages SHOULD be logged for further analysis. If the resolution does not succeed, the outsourcing operation is aborted and the HNA MUST close the Control Channel.

4.5.2. Providing information for the DNSSEC chain of trust

To provide the DS RRset to initialize the DNSSEC chain of trust the HNA MAY send a DNS UPDATE [RFC2136] message.

1. The NAME in the SOA MUST be set to the parent zone of the Registered Homenet Domain - that is where the DS records should be inserted.
2. The DS RRset MUST be placed in the Update section of the UPDATE query, and the NAME SHOULD be set to the Registered Homenet Domain.
3. The RDATA of the DS RR SHOULD correspond to the DS record to be inserted in the parent zone.
 - o A NOERROR response from the MD is a commitment to update the parent zone with the provided DS.
 - o An error indicates the MD will not update the DS, and other method should be used by the HNA.

4.5.3. Providing information for the Synchronization Channel

To provide the IP address of the primary, the HNA MAY send a DNS UPDATE message.

1. The NAME in the SOA MUST be the parent zone of the Registered Homenet Domain.
2. The Update section MUST be a RRset of Type NS.
3. The RDATA MUST be a RRset of type A or AAAA that designates the IP addresses associated to the primary.
4. There may be multiple IP addresses.
5. These IP addresses MUST be provided in the additional section.

The reason to provide these IP addresses is that it is NOT RECOMMENDED to publish these IP addresses. As a result, it is not expected to resolve them.

- o A NOERROR response indicates the DM has configured the secondary and is committed to serve as a secondary.
- o An error indicates the DM is not configured as a secondary.

The regular DNS error message SHOULD be returned to the HNA when an error occurs. In particular a FORMERR is returned when a format error is found, this includes when unexpected RRSets are added or when RRSets are missing.

- o A SERVFAIL error is returned when a internal error is encountered.
- o A NOTZONE error is returned when update and Zone sections are not coherent, a NOTAUTH error is returned when the DM is not authoritative for the Zone section.
- o A REFUSED error is returned when the DM refuses to proceed to the configuration and the requested action.

4.5.4. HNA instructing deleting the delegation

To instruct to delete the delegation the HNA MAY send a DNS UPDATE Delete message.

1. The NAME in the SOA MUST be the parent zone of the Registered Homenet Domain.
2. The Update section MUST be a RRset of Type NS.
3. The NAME associated to the NS RRset MUST be the Registered Domain Name.

As indicated by [RFC2136] section 2.5.2 the delete instruction is set by setting the TTL to 0, the Class to ANY, the RDLENGTH to 0 and the RDATA MUST be empty.

4.6. Securing the Control Channel between Homenet Naming Authority (HNA) and Distribution Master (DM)

The control channel between the HNA and the DM MUST be secured at both the HNA and the DM.

Secure protocols (like TLS [RFC8446] / DTLS [I-D.ietf-tls-dtls13]) SHOULD be used to secure the transactions between the DM and the HNA.

The advantage of TLS/DTLS is that this technology is widely deployed, and most of the devices already embed TLS/DTLS libraries, possibly also taking advantage of hardware acceleration. Further, TLS/DTLS provides authentication facilities and can use certificates to mutually authenticate the DM and HNA at the application layer, including available API. On the other hand, using TLS/DTLS requires implementing DNS exchanges over TLS/DTLS, as well as a new service port.

The HNA SHOULD authenticate inbound connections from the DM using standard mechanisms, such as a public certificate with baked-in root certificates on the HNA, or via DANE {!RFC6698}. The HNA is expected to be provisioned with a connection to the DM by the manufacturer, or during some user-initiated onboarding process, see Section 10.

The DM SHOULD authenticate the HNA and check that inbound messages are from the appropriate client. The DM MAY use a self-signed CA certificate mechanism per HNA, or public certificates for this purpose.

IPsec [RFC4301] and IKEv2 [RFC7296] were considered. They would need to operate in transport mode, and the authenticated end points would need to be visible to the applications, and this is not commonly available at the time of this writing.

A pure DNS solution using TSIG and/or SIG(0) to authenticate message was also considered. Section 10 envisions one mechanism would involve the end user, with a browser, signing up to a service provider, with a resulting OAUTH2 token to be provided to the HNA. A way to translate this OAUTH2 token from HTTPS web space to DNS SIG(0) space seems overly problematic, and so the enrollment protocol using web APIs was determined to be easier to implement at scale.

Note also that authentication of message exchanges between the HNA and the DM SHOULD NOT use the external IP address of the HNA to index the appropriate keys. As detailed in Section 11, the IP addresses of the DM and the Hidden Primary are subject to change, for example while the network is being renumbered. This means that the necessary keys to authenticate transaction SHOULD NOT be indexed using the IP address, and SHOULD be resilient to IP address changes.

4.7. Implementation Concerns

The Hidden Primary Server on the HNA differs from a regular authoritative server for the home network due to:

Interface Binding: the Hidden Primary Server will almost certainly listen on the WAN Interface, whereas a regular authoritative server for the home network would listen on the internal home network interface.

Limited exchanges: the purpose of the Hidden Primary Server is to synchronize with the DM, not to serve any zones to end users, or the public Internet.

As a result, exchanges are performed with specific nodes (the DM). Further, exchange types are limited. The only legitimate exchanges

are: NOTIFY initiated by the Hidden Primary and IXFR or AXFR exchanges initiated by the DM.

On the other hand, regular authoritative servers would respond to any hosts, and any DNS query would be processed. The HNA SHOULD filter IXFR/AXFR traffic and drop traffic not initiated by the DM. The HNA MUST listen for DNS on TCP and UDP and MUST at least allow SOA lookups of the Homenet Zone.

5. DM Synchronization Channel between HNA and DM

The DM Synchronization Channel is used for communication between the HNA and the DM for synchronizing the Public Homenet Zone. Note that the Control Channel and the Synchronization Channel are by construction different channels even though there they MAY use the same IP addresses. In fact the Control Channel is set between the HNA working as a client using port YYYY (a high range port) toward a service provided by the MD at port XX (well known port).

On the other hand, the Synchronization Channel is set between the DM working as a client using port ZZZZ (a high range port) toward a service a service provided by the HNA at port XX.

As a result, even though the same couple of IP addresses may be involved the Control Channel and the Synchronization Channel are always distinct channels.

Uploading and dynamically updating the zone file on the DM can be seen as zone provisioning between the HNA (Hidden Primary) and the DM (Secondary Server). This can be handled via AXFR + DNS UPDATE.

This document RECOMMENDS use of a primary / secondary mechanism instead of the use of DNS UPDATE. The primary / secondary mechanism is RECOMMENDED as it scales better and avoids DoS attacks. Note that even when UPDATE messages are used, these messages are using a distinct channel as those used to set the configuration.

Note that there is no standard way to distribute a DNS primary between multiple devices. As a result, if multiple devices are candidate for hosting the Hidden Primary, some specific mechanisms should be designed so the home network only selects a single HNA for the Hidden Primary. Selection mechanisms based on HNCP [RFC7788] are good candidates.

The HNA acts as a Hidden Primary Server, which is a regular authoritative DNS Server listening on the WAN interface.

The DM is configured as a secondary for the Homenet Domain Name. This secondary configuration has been previously agreed between the end user and the provider of the Outsourcing Infrastructure as part of either the provisioning or due to receipt of UPDATE messages on the DM Control Channel.

The Homenet Reverse Zone MAY also be updated either with DNS UPDATE [RFC2136] or using a primary / secondary synchronization.

5.1. Securing the Synchronization Channel between HNA and DM

The Synchronization Channel used standard DNS request.

First the primary notifies the secondary that the zone must be updated and eaves the secondary to proceed with the update when possible/convenient.

Then, a NOTIFY message is sent by the primary, which is a small packet that is less likely to load the secondary.

Finally, the AXFR [RFC1034] or IXFR [RFC1995] query performed by the secondary is a small packet sent over TCP (section 4.2 [RFC5936]), which mitigates reflection attacks using a forged NOTIFY.

The AXFR request from the DM to the HNA SHOULD be secured. DNS over TLS [RFC7858] is RECOMMENDED.

When using TLS, the HNA MAY authenticate inbound connections from the DM using standard mechanisms, such as a public certificate with baked-in root certificates on the HNA, or via DANE [RFC6698]

The HNA MAY apply a simple IP filter on inbound AXFR requests to ensure they only arrive from the DM Synchronization Channel. In this case, the HNA SHOULD regularly check (via DNS resolution) that the address of the DM in the filter is still valid.

6. DM Distribution Channel

The DM Distribution Channel is used for communication between the DM and the Public Authoritative Servers. The architecture and communication used for the DM Distribution Channels is outside the scope of this document, and there are many existing solutions available e.g. rsynch, DNS AXFR, REST, DB copy.

7. HNA Security Policies

This section details security policies related to the Hidden Primary / Secondary synchronization.

The Hidden Primary, as described in this document SHOULD drop any queries from the home network. This could be implemented via port binding and/or firewall rules. The precise mechanism deployed is out of scope of this document. The Hidden Primary SHOULD drop any DNS queries arriving on the WAN interface that are not issued from the DM. The Hidden Primary SHOULD drop any outgoing packets other than DNS NOTIFY query, SOA response, IXFR response or AXFR responses. The Hidden Primary SHOULD drop any incoming packets other than DNS NOTIFY response, SOA query, IXFR query or AXFR query. The Hidden Primary SHOULD drop any non protected IXFR or AXFR exchange, depending on how the synchronization is secured.

8. DNSSEC compliant Homenet Architecture

[RFC7368] in Section 3.7.3 recommends DNSSEC to be deployed on both the authoritative server and the resolver. The resolver side is out of scope of this document, and only the authoritative part of the server is considered.

This document assumes the HNA signs the Public Homenet Zone.

Secure delegation is achieved only if the DS RRset is properly set in the parent zone. Secure delegation is performed by the HNA or the DNS Outsourcing Infrastructures.

The DS RRset can be updated manually with nsupdate for example. This requires the HNA or the DNS Outsourcing Infrastructure to be authenticated by the DNS server hosting the parent of the Public Homenet Zone. Such a trust channel between the HNA and the parent DNS server may be hard to maintain with HNAs, and thus may be easier to establish with the DNS Outsourcing Infrastructure. In fact, the Public Authoritative Server(s) may use Automating DNSSEC Delegation Trust Maintenance [RFC7344].

9. Homenet Reverse Zone Channels Configuration

The Public Homenet Zone is associated to a Registered Homenet Domain and the ownership of that domain requires a specific registration from the end user as well as the HNA being provisioned with some authentication credentials. Such steps are mandatory unless the DNS Outsourcing Infrastructure has some other means to authenticate the HNA. Such situation may occur, for example, when the ISP provides the Homenet Domain as well as the DNS Outsourcing Infrastructure.

In this case, the HNA may be authenticated by the physical link layer, in which case the authentication of the HNA may be performed without additional provisioning of the HNA. While this may be not so common for the Public Homenet Zone, this situation is expected to be quite common for the Reverse Homenet Zone.

More specifically, a common case is that the upstream ISP provides the IPv6 prefix to the Homenet with a IA_PD [RFC8415] option and manages the DNS Outsourcing Infrastructure of the associated reverse zone. This leave place for setting up automatically the relation between HNA and the DNS Outsourcing infrastructure as described in [I-D.ietf-homenet-naming-architecture-dhc-options].

With this relation automatically configured, the synchronization between the Home network and the DNS Outsourcing Infrastructure happens similarly as for the Public Homenet Zone described earlier in this document.

Note that for home networks hosted by multiple ISPs, each ISP provides only the DNS Outsourcing Infrastructure of the reverse zones associated to the delegated prefix. It is also likely that the DNS exchanges will need to be performed on dedicated interfaces as to be accepted by the ISP. More specifically, the reverse zone associated to prefix 1 will not be possible to be performs by the HNA using an IP address that belongs to prefix 2. Such constraints does not raise major concerns either for hot standby or load sharing configuration.

With IPv6, the domain space for IP addresses is so large that reverse zone may be confronted with scalability issues. How the reverse zone is generated is out of scope of this document.

[I-D.howard-dnsop-ip6rdns] provides guidance on how to address scalability issues.

10. Homenet Public Zone Channel Configurations

This document does not deal with how the HNA is provisioned with a trusted relationship to the Distribution Master for the forward zone.

This section details what needs to be provisioned into the HNA and serves as a requirements statement for mechanisms.

The HNA needs to be provisioned with:

- o the Registered Domain (e.g., myhome.isp.example)
- o the contact info for the Distribution Master (DM), including the DNS name (FQDN), possibly including the IP literal, and a certificate (or anchor) to be used to authenticate the service

- o the DM transport protocol and port (the default is DNS over TLS, on port 853)
- o the HNA credentials used by the DM for its authentication.

The HNA will need to select an IP address for communication for the Synchronization Channel. This is typically the outside WAN address of the router, but could be an IPv6 LAN address in the case of a home with multiple ISPs (and multiple border routers). This is communicated in section BLAH when the NS and A record is communicated.

The above parameters MUST be provisioned for ISP-specific reverse zones, as per [I-D.ietf-homenet-naming-architecture-dhc-options]. ISP-specific forward zones MAY also be provisioned using [I-D.ietf-homenet-naming-architecture-dhc-options], but zones which are not related to a specific ISP zone (such as with a DNS provider) must be provisioned through other means.

Similarly, if the HNA is provided by a registrar, the HNA may be given configured to end user.

In the absence of specific pre-established relation, these pieces of information may be entered manually by the end user. In order to ease the configuration from the end user the following scheme may be implemented.

The HNA may present the end user a web interface where it provides the end user the ability to indicate the Registered Domain or the registrar for example a preselected list. Once the registrar has been selected, the HNA redirects the end user to that registrar in order to receive an access token. The access token will enable the HNA to retrieve the DM parameters associated to the Registered Domain. These parameters will include the credentials used by the HNA to establish the Control and Synchronization Channels.

Such architecture limits the necessary steps to configure the HNA from the end user.

11. Renumbering

This section details how renumbering is handled by the Hidden Primary server or the DM. Both types of renumbering are discussed i.e. "make-before-break" and "break-before-make" (aka flash renumbering).

In the make-before-break renumbering scenario, the new prefix is advertised, the network is configured to prepare the transition to

the new prefix. During a period of time, the two prefixes old and new coexist, before the old prefix is completely removed.

In the break-before-make renumbering scenario, the new prefix is advertised making the old prefix obsolete.

Renumbering has been extensively described in [RFC4192] and analyzed in [RFC7010] and the reader is expected to be familiar with them before reading this section.

11.1. Hidden Primary

In a renumbering scenario, the Hidden Primary is informed it is being renumbered. In most cases, this occurs because the whole home network is being renumbered. As a result, the Public Homenet Zone will also be updated. Although the new and old IP addresses may be stored in the Public Homenet Zone, we recommend that only the newly reachable IP addresses be published.

To avoid reachability disruption, IP connectivity information provided by the DNS SHOULD be coherent with the IP plane. In our case, this means the old IP address SHOULD NOT be provided via the DNS when it is not reachable anymore. Let for example TTL be the TTL associated with a RRset of the Public Homenet Zone, it may be cached for TTL seconds. Let T_NEW be the time the new IP address replaces the old IP address in the Homenet Zone, and T_OLD_UNREACHABLE the time the old IP is not reachable anymore.

In the case of the make-before-break, seamless reachability is provided as long as $T_OLD_UNREACHABLE - T_NEW > 2 * TTL$. If this is not satisfied, then devices associated with the old IP address in the home network may become unreachable for $2 * TTL - (T_OLD_UNREACHABLE - T_NEW)$. In the case of a break-before-make, $T_OLD_UNREACHABLE = T_NEW$, and the device may become unreachable up to $2 * TTL$.

Once the Public Homenet Zone file has been updated on the Hidden Primary, the Hidden Primary needs to inform the DNS Outsourcing Infrastructure that the Public Homenet Zone has been updated and that the IP address to use to retrieve the updated zone has also been updated. Both notifications are performed using regular DNS exchanges. Mechanisms to update an IP address provided by lower layers with protocols like SCTP [RFC4960], MOBIKE [RFC4555] are not considered in this document.

The Hidden Primary SHOULD inform the DM that the Public Homenet Zone has been updated by sending a NOTIFY payload with the new IP address. In addition, this NOTIFY payload SHOULD be authenticated using SIG(0) or TSIG. When the DM receives the NOTIFY payload, it MUST

authenticate it. Note that the cryptographic key used for the authentication SHOULD be indexed by the Registered Homenet Domain contained in the NOTIFY payload as well as the RRSIG. In other words, the IP address SHOULD NOT be used as an index.

If authentication succeeds, the DM MUST also notice the IP address has been modified and perform a reachability check before updating its primary configuration. The routability check MAY be performed by sending a SOA request to the Hidden Primary using the source IP address of the NOTIFY. This exchange is also secured, and if an authenticated response is received from the Hidden Primary with the new IP address, the DM SHOULD update its configuration file and retrieve the Public Homenet Zone using an AXFR or a IXFR exchange.

Note that the primary reason for providing the IP address is that the Hidden Primary is not publicly announced in the DNS. If the Hidden Primary were publicly announced in the DNS, then the IP address update could have been performed using the DNS as described in Section 11.2.

11.2. Distribution Master

Renumbering of the Distribution Master results in it changing its IP address. As the DM is a secondary, the destination of DNS NOTIFY payloads MUST be changed, and any configuration/firewalling that restricts DNS AXFR/IXFR operations MUST be updated.

If the DM is configured in the Hidden Primary configuration file using a FQDN, then the update of the IP address is performed by DNS. More specifically, before sending the NOTIFY, the Hidden Primary performs a DNS resolution to retrieve the IP address of the secondary.

As described in Section 11.1, the DM DNS information SHOULD be coherent with the IP plane. The TTL of the Distribution Master name SHOULD be adjusted appropriately prior to changing the IP address.

Some DNS infrastructure uses the IP address to designate the secondary, in which case, other mechanisms must be found. A reason for using IP addresses instead of names is generally to reach an internal interface that is not designated by a FQDN, and to avoid potential bootstrap problems. Such scenarios are considered as out of scope in the case of home networks.

12. Privacy Considerations

Outsourcing the DNS Authoritative service from the HNA to a third party raises a few privacy related concerns.

The Public Homenet Zone lists the names of services hosted in the home network. Combined with blocking of AXFR queries, the use of NSEC3 [RFC5155] (vs NSEC [RFC4034]) prevents an attacker from being able to walk the zone, to discover all the names. However, the attacker may be able to walk the reverse DNS zone, or use other reconnaissance techniques to learn this information as described in [RFC7707].

In general a home owner is expected only to publish names for which there is some need to be able to reference externally. Publication of the name does not imply that the service is necessarily reachable from any or all parts of the Internet. [RFC7084] mandates that the outgoing-only policy [RFC6092] be available, and in many cases it is configured by default. A well designed User Interface would combine a policy for making a service public by a name with a policy on who may access it.

In many cases, the home owner wishes to publish names for services that only they will be able to access. The access control may consist of an IP source address range, or access may be restricted via some VPN functionality. The purpose of publishing the name is so that the service may be access by the same name both within the home, and outside the home. Sending traffic to the relevant IPv6 address causes the relevant VPN policy to be enacted upon.

While the problem of getting access to internal names has been solved in Enterprise configurations with a split-DNS, and such a thing could be done in the home, many recent improvements to VPN user interfaces make it more likely that an individual might have multiple connections configured. For instance, an adult child checking on the state of a home automation system for a parent.

In addition to the Public Homenet Zone, pervasive DNS monitoring can also monitor the traffic associated with the Public Homenet Zone. This traffic may provide an indication of the services an end user accesses, plus how and when they use these services. Although, caching may obfuscate this information inside the home network, it is likely that outside your home network this information will not be cached.

13. Security Considerations

The Homenet Naming Architecture described in this document solves exposing the HNA's DNS service as a DoS attack vector.

13.1. HNA DMand RDM channels

The HNA DM channels are specified to include their own security mechanisms that are designed to provide the minimum attack surface, and to authenticate transactions where necessary.

Note that in the case of the Reverse Homenet Zone, the data is less subject to attacks than in the Public Homenet Zone. In addition, the HNA and the DM MAY belong to the same administrative domain, i.e. the ISP. More specifically, the WAN interface is located in the ISP network. As a result, if provisioned using DHCPv6, the security credential may not even transit in the home network. On the other hand, if the HNA is not hosted at the border of the home network, the credential may rely on the security associated to DHCPv6. Even if HNA and DM are in the same administrative domain it is strongly RECOMMENDED to use a secure channel.

The security of these channels heavily relies on TLS and the DM or RDM is authenticated by its certificate. To ensure the multiple TLS session are continuously authenticating the same entity, TLS may take advantage of second factor authentication as described in [RFC8672].

13.2. Names are less secure than IP addresses

This document describes how an end user can make their services and devices from his home network reachable on the Internet by using names rather than IP addresses. This exposes the home network to attackers, since names are expected to include less entropy than IP addresses. In fact, with IP addresses, the Interface Identifier is 64 bits long leading to up to 2^{64} possibilities for a given subnetwork. This is not to mention that the subnet prefix is also of 64 bits long, thus providing up to 2^{64} possibilities. On the other hand, names used either for the home network domain or for the devices present less entropy (livebox, router, printer, nicolas, jennifer, ...) and thus potentially exposes the devices to dictionary attacks.

13.3. Names are less volatile than IP addresses

IP addresses may be used to locate a device, a host or a service. However, home networks are not expected to be assigned a time invariant prefix by ISPs. As a result, observing IP addresses only

provides some ephemeral information about who is accessing the service. On the other hand, names are not expected to be as volatile as IP addresses. As a result, logging names over time may be more valuable than logging IP addresses, especially to profile an end user's characteristics.

PTR provides a way to bind an IP address to a name. In that sense, responding to PTR DNS queries may affect the end user's privacy. For that reason end users may choose not to respond to PTR DNS queries and MAY instead return a NXDOMAIN response.

13.4. DNS Reflection Attacks

An attacker performs a reflection attack when it sends traffic to one or more intermediary nodes (reflectors), that in turn send back response traffic to the victim. Motivations for using an intermediary node might be anonymity of the attacker, as well as amplification of the traffic. Typically, when the intermediary node is a DNSSEC server, the attacker sends a DNSSEC query and the victim is likely to receive a DNSSEC response. This section analyzes how the different components may be involved as a reflector in a reflection attack. Section 13.5 considers the Hidden Primary, Section 13.6 the Synchronization Server, and Section 13.7 the Public Authoritative Server(s).

13.5. Reflection Attack involving the Hidden Primary

With the specified architecture, the Hidden Primary is only expected to receive DNS queries of type SOA, AXFR or IXFR. This section analyzes how these DNS queries may be used by an attacker to perform a reflection attack.

DNS queries of type AXFR and IXFR use TCP and as such are less subject to reflection attacks. This makes SOA queries the only remaining practical vector of attacks for reflection attacks, based on UDP.

SOA queries are not associated with a large amplification factor compared to queries of type "ANY" or to query of non existing FQDNs. This reduces the probability a DNS query of type SOA will be involved in a DDoS attack.

SOA queries are expected to follow a very specific pattern, which makes rate limiting techniques an efficient way to limit such attacks, and associated impact on the naming service of the home network.

Motivations for such a flood might be a reflection attack, but could also be a resource exhaustion attack performed against the Hidden Primary. The Hidden Primary only expects to exchange traffic with the DM, that is its associated secondary. Even though secondary servers may be renumbered as mentioned in Section 11, the Hidden Primary is likely to perform a DNSSEC resolution and find out the associated secondary's IP addresses in use. As a result, the Hidden Primary is likely to limit the origin of its incoming traffic based on the origin IP address.

With filtering rules based on IP address, SOA flooding attacks are limited to forged packets with the IP address of the secondary server. In other words, the only victims are the Hidden Primary itself or the secondary. There is a need for the Hidden Primary to limit that flood to limit the impact of the reflection attack on the secondary, and to limit the resource needed to carry on the traffic by the HNA hosting the Hidden Primary. On the other hand, mitigation should be performed appropriately, so as to limit the impact on the legitimate SOA sent by the secondary.

The main reason for the DM sending a SOA query is to update the SOA RRset after the TTL expires, to check the serial number upon the receipt of a NOTIFY query from the Hidden Primary, or to re-send the SOA request when the response has not been received. When a flood of SOA queries is received by the Hidden Primary, the Hidden Primary may assume it is involved in an attack.

There are few legitimate time slots when the secondary is expected to send a SOA query. Suppose T_{NOTIFY} is the time a NOTIFY is sent by the Hidden Primary, T_{SOA} the last time the SOA has been queried, T_{TTL} the TTL associated to the SOA, and T_{REFRESH} the refresh time defined in the SOA RRset. The specific time SOA queries are expected can be for example T_{NOTIFY} , $T_{\text{SOA}} + 2/3 \text{ TTL}$, $T_{\text{SOA}} + \text{TTL}$, $T_{\text{SOA}} + T_{\text{REFRESH}}$, and. Outside a few minutes following these specific time slots, the probability that the HNA discards a legitimate SOA query is very low. Within these time slots, the probability the secondary may have its legitimate query rejected is higher. If a legitimate SOA is discarded, the secondary will re-send SOA query every "retry time" second until "expire time" seconds occurs, where "retry time" and "expire time" have been defined in the SOA.

As a result, it is RECOMMENDED to set rate limiting policies to protect HNA resources. If a flood lasts more than the expired time defined by the SOA, it is RECOMMENDED to re-initiate a synchronization between the Hidden Primary and the secondaries.

13.6. Reflection Attacks involving the DM

The DM acts as a secondary coupled with the Hidden Primary. The secondary expects to receive NOTIFY query, SOA responses, AXFR and IXFR responses from the Hidden Primary.

Sending a NOTIFY query to the secondary generates a NOTIFY response as well as initiating an SOA query exchange from the secondary to the Hidden Primary. As mentioned in [RFC1996], this is a known "benign denial of service attack". As a result, the DM SHOULD enforce rate limiting on sending SOA queries and NOTIFY responses to the Hidden Primary. Most likely, when the secondary is flooded with valid and signed NOTIFY queries, it is under a replay attack which is discussed in Section 13.9. The key thing here is that the secondary is likely to be designed to be able to process much more traffic than the Hidden Primary hosted on a HNA.

This paragraph details how the secondary may limit the NOTIFY queries. Because the Hidden Primary may be renumbered, the secondary SHOULD NOT perform permanent IP filtering based on IP addresses. In addition, a given secondary may be shared among multiple Hidden Primaries which make filtering rules based on IP harder to set. The time at which a NOTIFY is sent by the Hidden Primary is not predictable. However, a flood of NOTIFY messages may be easily detected, as a NOTIFY originated from a given Homenet Zone is expected to have a very limited number of unique source IP addresses, even when renumbering is occurring. As a result, the secondary, MAY rate limit incoming NOTIFY queries.

On the Hidden Primary side, it is recommended that the Hidden Primary sends a NOTIFY as long as the zone has not been updated by the secondary. Multiple SOA queries may indicate the secondary is under attack.

13.7. Reflection Attacks involving the Public Authoritative Servers

Reflection attacks involving the Public Authoritative Server(s) are similar to attacks on any DNS Outsourcing Infrastructure. This is not specific to the architecture described in this document, and thus are considered as out of scope.

In fact, one motivation of the architecture described in this document is to expose the Public Authoritative Server(s) to attacks instead of the HNA, as it is believed that the Public Authoritative Server(s) will be better able to defend itself.

13.8. Flooding Attack

The purpose of flooding attacks is mostly resource exhaustion, where the resource can be bandwidth, memory, or CPU for example.

One goal of the architecture described in this document is to limit the surface of attack on the HNA. This is done by outsourcing the DNS service to the Public Authoritative Server(s). By doing so, the HNA limits its DNS interactions between the Hidden Primary and the DM. This limits the number of entities the HNA interacts with as well as the scope of DNS exchanges - NOTIFY, SOA, AXFR, IXFR.

The use of an authenticated channel with SIG(0) or TSIG between the HNA and the DM, enables detection of illegitimate DNS queries, so appropriate action may be taken - like dropping the queries. If signatures are validated, then most likely, the HNA is under a replay attack, as detailed in Section 13.9

In order to limit the resource required for authentication, it is recommended to use TSIG that uses symmetric cryptography over SIG(0) that uses asymmetric cryptography.

13.9. Replay Attack

Replay attacks consist of an attacker either resending or delaying a legitimate message that has been sent by an authorized user or process. As the Hidden Primary and the DM use an authenticated channel, replay attacks are mostly expected to use forged DNS queries in order to provide valid traffic.

From the perspective of an attacker, using a correctly authenticated DNS query may not be detected as an attack and thus may generate a response. Generating and sending a response consumes more resources than either dropping the query by the defender, or generating the query by the attacker, and thus could be used for resource exhaustion attacks. In addition, as the authentication is performed at the DNS layer, the source IP address could be impersonated in order to perform a reflection attack.

Section 13.4 details how to mitigate reflection attacks and Section 13.8 details how to mitigate resource exhaustion. Both sections assume a context of DoS with a flood of DNS queries. This section suggests a way to limit the attack surface of replay attacks.

As SIG(0) and TSIG use inception and expiration time, the time frame for replay attack is limited. SIG(0) and TSIG recommends a fudge value of 5 minutes. This value has been set as a compromise between possibly loose time synchronization between devices and the valid

lifetime of the message. As a result, better time synchronization policies could reduce the time window of the attack.

14. Data Model for Outsourced information

The following is an abridged example of a set of data that represents the needed configuration parameters for outsourcing.

```
{
  "dm_notify" : "2001:db8:1f15:62e:21c::2",
  "dm_acl"    : "2001:db8:1f15:62e:21c::/64",
  "dm_ctrl"   : "192.168.1.18",
  "dm_port"   : "4433",
  "ns_list"   : [ "ns1.publicdns.example", "ns2.publicdns.example"],
  "zone"      : "daniel.homenetdns.example",
  "auth_method" : "certificate",
  "hna_certificate": "-----BEGIN CERTIFICATE-----\nMIIDTjCCFGy....",
  "hna_key"    : "-----BEGIN RSA PRIVATE KEY-----\nMIIEowICAQE...."
}
```

Here goes a YANG MODULE description of the above.

15. IANA Considerations

This document has no actions for IANA.

16. Acknowledgment

The authors wish to thank Philippe Lemordant for its contributions on the early versions of the draft; Ole Troan for pointing out issues with the IPv6 routed home concept and placing the scope of this document in a wider picture; Mark Townsley for encouragement and injecting a healthy debate on the merits of the idea; Ulrik de Bie for providing alternative solutions; Paul Mockapetris, Christian Jacquenet, Francis Dupont and Ludovic Eschard for their remarks on HNA and low power devices; Olafur Gudmundsson for clarifying DNSSEC capabilities of small devices; Simon Kelley for its feedback as dnsmasq implementer; Andrew Sullivan, Mark Andrew, Ted Lemon, Mikael Abrahamson, and Ray Bellis for their feedback on handling different views as well as clarifying the impact of outsourcing the zone signing operation outside the HNA; Mark Andrew and Peter Koch for clarifying the renumbering.

17. References

17.1. Normative References

- [RFC1033] Lottor, M., "Domain Administrators Operations Guide", RFC 1033, DOI 10.17487/RFC1033, November 1987, <<https://www.rfc-editor.org/info/rfc1033>>.
- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [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>>.
- [RFC1995] Ohta, M., "Incremental Zone Transfer in DNS", RFC 1995, DOI 10.17487/RFC1995, August 1996, <<https://www.rfc-editor.org/info/rfc1995>>.
- [RFC1996] Vixie, P., "A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY)", RFC 1996, DOI 10.17487/RFC1996, August 1996, <<https://www.rfc-editor.org/info/rfc1996>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, DOI 10.17487/RFC2136, April 1997, <<https://www.rfc-editor.org/info/rfc2136>>.
- [RFC2142] Crocker, D., "Mailbox Names for Common Services, Roles and Functions", RFC 2142, DOI 10.17487/RFC2142, May 1997, <<https://www.rfc-editor.org/info/rfc2142>>.
- [RFC2308] Andrews, M., "Negative Caching of DNS Queries (DNS NCACHE)", RFC 2308, DOI 10.17487/RFC2308, March 1998, <<https://www.rfc-editor.org/info/rfc2308>>.
- [RFC2845] Vixie, P., Gudmundsson, O., Eastlake 3rd, D., and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", RFC 2845, DOI 10.17487/RFC2845, May 2000, <<https://www.rfc-editor.org/info/rfc2845>>.
- [RFC2931] Eastlake 3rd, D., "DNS Request and Transaction Signatures (SIG(0)s)", RFC 2931, DOI 10.17487/RFC2931, September 2000, <<https://www.rfc-editor.org/info/rfc2931>>.

- [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <<https://www.rfc-editor.org/info/rfc4034>>.
- [RFC4192] Baker, F., Lear, E., and R. Droms, "Procedures for Renumbering an IPv6 Network without a Flag Day", RFC 4192, DOI 10.17487/RFC4192, September 2005, <<https://www.rfc-editor.org/info/rfc4192>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC4555] Eronen, P., "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", RFC 4555, DOI 10.17487/RFC4555, June 2006, <<https://www.rfc-editor.org/info/rfc4555>>.
- [RFC4960] Stewart, R., Ed., "Stream Control Transmission Protocol", RFC 4960, DOI 10.17487/RFC4960, September 2007, <<https://www.rfc-editor.org/info/rfc4960>>.
- [RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS Security (DNSSEC) Hashed Authenticated Denial of Existence", RFC 5155, DOI 10.17487/RFC5155, March 2008, <<https://www.rfc-editor.org/info/rfc5155>>.
- [RFC5936] Lewis, E. and A. Hoenes, Ed., "DNS Zone Transfer Protocol (AXFR)", RFC 5936, DOI 10.17487/RFC5936, June 2010, <<https://www.rfc-editor.org/info/rfc5936>>.
- [RFC6092] Woodyatt, J., Ed., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, DOI 10.17487/RFC6092, January 2011, <<https://www.rfc-editor.org/info/rfc6092>>.
- [RFC6644] Evans, D., Droms, R., and S. Jiang, "Rebind Capability in DHCPv6 Reconfigure Messages", RFC 6644, DOI 10.17487/RFC6644, July 2012, <<https://www.rfc-editor.org/info/rfc6644>>.
- [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>>.
- [RFC6887] Wing, D., Ed., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, DOI 10.17487/RFC6887, April 2013, <<https://www.rfc-editor.org/info/rfc6887>>.
- [RFC7010] Liu, B., Jiang, S., Carpenter, B., Venaas, S., and W. George, "IPv6 Site Renumbering Gap Analysis", RFC 7010, DOI 10.17487/RFC7010, September 2013, <<https://www.rfc-editor.org/info/rfc7010>>.
- [RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, DOI 10.17487/RFC7084, November 2013, <<https://www.rfc-editor.org/info/rfc7084>>.
- [RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <<https://www.rfc-editor.org/info/rfc7296>>.
- [RFC7344] Kumari, W., Gudmundsson, O., and G. Barwood, "Automating DNSSEC Delegation Trust Maintenance", RFC 7344, DOI 10.17487/RFC7344, September 2014, <<https://www.rfc-editor.org/info/rfc7344>>.
- [RFC7368] Chown, T., Ed., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", RFC 7368, DOI 10.17487/RFC7368, October 2014, <<https://www.rfc-editor.org/info/rfc7368>>.
- [RFC7558] Lynn, K., Cheshire, S., Blanchet, M., and D. Migault, "Requirements for Scalable DNS-Based Service Discovery (DNS-SD) / Multicast DNS (mDNS) Extensions", RFC 7558, DOI 10.17487/RFC7558, July 2015, <<https://www.rfc-editor.org/info/rfc7558>>.
- [RFC7707] Gont, F. and T. Chown, "Network Reconnaissance in IPv6 Networks", RFC 7707, DOI 10.17487/RFC7707, March 2016, <<https://www.rfc-editor.org/info/rfc7707>>.
- [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>>.

- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8375] Pfister, P. and T. Lemon, "Special-Use Domain 'home.arpa.'", RFC 8375, DOI 10.17487/RFC8375, May 2018, <<https://www.rfc-editor.org/info/rfc8375>>.
- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.

17.2. Informative References

- [I-D.howard-dnsop-ip6rdns]
Howard, L., "Reverse DNS in IPv6 for Internet Service Providers", draft-howard-dnsop-ip6rdns-00 (work in progress), June 2014.
- [I-D.ietf-homenet-naming-architecture-dhc-options]
Migault, D., Weber, R., Mrugalski, T., Griffiths, C., and W. Cloetens, "DHCPv6 Options for Home Network Naming Authority", draft-ietf-homenet-naming-architecture-dhc-options-08 (work in progress), October 2020.
- [I-D.ietf-homenet-simple-naming]
Lemon, T., Migault, D., and S. Cheshire, "Homenet Naming and Service Discovery Architecture", draft-ietf-homenet-simple-naming-03 (work in progress), October 2018.
- [I-D.ietf-tls-dtls13]
Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", draft-ietf-tls-dtls13-38 (work in progress), May 2020.

[I-D.richardson-homerouter-provisioning]

Richardson, M., "Provisioning Initial Device Identifiers into Home Routers", draft-richardson-homerouter-provisioning-00 (work in progress), November 2020.

[RFC8672] Sheffer, Y. and D. Migault, "TLS Server Identity Pinning with Tickets", RFC 8672, DOI 10.17487/RFC8672, October 2019, <<https://www.rfc-editor.org/info/rfc8672>>.

Appendix A. Envisioned deployment scenarios

A number of deployment have been envisioned, this section aims at providing a brief description. The use cases are not limitations and this section is not normative.

A.1. CPE Vendor

A specific vendor with specific relations with a registrar or a registry may sell a CPE that is provisioned with provisioned domain name. Such domain name does not need to be necessary human readable.

One possible way is that the vendor also provisions the HNA with a private and public keys as well as a certificate. Note that these keys are not expected to be used for DNSSEC signing. Instead these keys are solely used by the HNA to proceed to the authentication. Normally the keys should be necessary and sufficient to proceed to the authentication. The reason to combine the domain name and the key is that outsourcing infrastructure are likely handle names better than keys and that domain names might be used as a login which enables the key to be regenerated.

When the home network owner plugs the CPE at home, the relation between HNA and DM is expected to work out-of-the-box.

A.2. Agnostic CPE

An CPE that is not preconfigured may also take advantage to the protocol defined in this document but some configuration steps will be needed.

1. The owner of the home network buys a domain name to a registrar, and as such creates an account on that registrar
2. Either the registrar is also providing the outsourcing infrastructure or the home network needs to create a specific account on the outsourcing infrastructure. * If the outsourcing provider is the registrar, the outsourcing has by design a proof of ownership of the domain name by the homenet owner. In this case, it is expected the infrastructure provides the necessary parameters to the home network owner to configure the HNA. A good way to provide the parameters would be the home network be able to copy/paste a JSON object. What matters at that point is the outsourcing infrastructure being able to generate authentication credentials for the HNA to authenticate itself to the outsourcing infrastructure. This obviously requires the home network to provide the public key generated by the HNA in a CSR.

- o If the outsourcing infrastructure is not the registrar, then the proof of ownership needs to be established using protocols like ACME for example that will end in the generation of a certificate. ACME is used here to the purpose of automating the generation of the certificate, the CA may be a specific CA or the outsourcing infrastructure. With that being done, the outsourcing infrastructure has a roof of ownership and can proceed as above.

Appendix B. Example: Homenet Zone

This section is not normative and intends to illustrate how the HNA builds the Homenet Zone.

As depicted in Figure 1, the Public Homenet Zone is hosted on the Public Authoritative Server(s), whereas the Homenet Zone is hosted on the HNA. This section considers that the HNA builds the zone that will be effectively published on the Public Authoritative Server(s). In other words "Homenet to Public Zone transformation" is the identity also commonly designated as "no operation" (NOP).

In that case, the Homenet Zone should configure its Name Server RRset (NS) and Start of Authority (SOA) with the values associated with the Public Authoritative Server(s). This is illustrated in Figure 2. public.primary.example.net is the FQDN of the Public Authoritative Server(s), and IP1, IP2, IP3, IP4 are the associated IP addresses. Then the HNA should add the additional new nodes that enter the home network, remove those that should be removed, and sign the Homenet Zone.

```

$ORIGIN example.com
$TTL 1h

@ IN SOA public.primary.example.net
    hostmaster.example.com. (
        2013120710 ; serial number of this zone file
        1d         ; secondary refresh
        2h         ; secondary retry time in case of a problem
        4w         ; secondary expiration time
        1h         ; maximum caching time in case of failed
                   ; lookups
    )

@ NS public.authoritative.servers.example.net

public.primary.example.net A @IP1
public.primary.example.net A @IP2
public.primary.example.net AAAA @IP3
public.primary.example.net AAAA @IP4

```

Figure 2: Homenet Zone

The SOA RRset is defined in [RFC1033], [RFC1035] and [RFC2308]. This SOA is specific, as it is used for the synchronization between the Hidden Primary and the DM and published on the DNS Public Authoritative Server(s)...

- o MNAME: indicates the primary. In our case the zone is published on the Public Authoritative Server(s), and its name MUST be included. If multiple Public Authoritative Server(s) are involved, one of them MUST be chosen. More specifically, the HNA MUST NOT include the name of the Hidden Primary.
- o RNAME: indicates the email address to reach the administrator. [RFC2142] recommends using hostmaster@domain and replacing the '@' sign by '.'.
- o REFRESH and RETRY: indicate respectively in seconds how often secondaries need to check the primary, and the time between two refresh when a refresh has failed. Default values indicated by [RFC1033] are 3600 (1 hour) for refresh and 600 (10 minutes) for retry. This value might be too long for highly dynamic content. However, the Public Authoritative Server(s) and the HNA are expected to implement NOTIFY [RFC1996]. So whilst shorter refresh timers might increase the bandwidth usage for secondaries hosting large number of zones, it will have little practical impact on the elapsed time required to achieve synchronization between the

Outsourcing Infrastructure and the Hidden Master. As a result, the default values are acceptable.

- o EXPIRE: is the upper limit data SHOULD be kept in absence of refresh. The default value indicated by [RFC1033] is 3600000 (approx. 42 days). In home network architectures, the HNA provides both the DNS synchronization and the access to the home network. This device may be plugged and unplugged by the end user without notification, thus we recommend a long expiry timer.
- o MINIMUM: indicates the minimum TTL. The default value indicated by [RFC1033] is 86400 (1 day). For home network, this value MAY be reduced, and 3600 (1 hour) seems more appropriate.

Appendix C. Example: HNA necessary parameters for outsourcing

This section specifies the various parameters required by the HNA to configure the naming architecture of this document. This section is informational, and is intended to clarify the information handled by the HNA and the various settings to be done.

The HNA needs to be configured with the following parameters. These parameters are necessary to establish a secure channel between the HNA and the DM as well as to specify the DNS zone that is in the scope of the communication:

Distribution Master notification address (dm_notify): The associated FQDNs or IP addresses of the DM to which DNS notifies should be sent. IP addresses are optional and the FQDN is sufficient and preferred. If there are concerns about the security of the name to IP translation, then DNSSEC should be employed.

Authentication Method ("method"): How the HNA authenticates itself to the DM. This specification defines only "certificate"

Authentication data ("hna_certificate", "hna_key"): While a PSK can be used as part of TSIG authentication, it has poor security properties and is hard to scale. Better solutions use public key mechanisms, leveraging private keys built into the HNA.

Public Authoritative Server(s) (dm_ctrl and dm_port): The FQDN or IP addresses of the Public Authoritative Server(s) to which control messages will be sent. IP addresses are optional and the FQDN is sufficient.

(XXX? what? It MAY correspond to the data that will be sent in the NS RRsets and SOA of the Homenet Zone.)

Registered Homenet Domain (???): The domain name used to establish the secure channel. This name is used by the DM and the HNA for the primary / secondary configuration as well as to index the NOTIFY queries of the HNA when the HNA has been renumbered.

Registered Homenet Domain (zone): The Domain Name of the zone. Multiple Registered Homenet Domains may be provided. This will generate the creation of multiple Public Homenet Zones.

Public Authoritative Server (ns-list): The Public Authoritative Server(s) associated with the Registered Homenet Domain. Multiple Public Authoritative Server(s) may be provided.

For forward zones, the relationship between the HNA and the forward zone provider may be the result of a number of transactions:

1. The forward zone outsourcing may be provided by the maker of the Homenet router. In this case, the identity and authorization could be built in the device at manufacturer provisioning time. The device would need to be provisioned with a device-unique credential, and it is likely that the Registered Homenet Domain would be derived from a public attribute of the device, such as a serial number.
2. The forward zone outsourcing may be provided by the Internet Service Provider. In this case, the use of [I-D.ietf-homenet-naming-architecture-dhc-options] to provide the credentials is appropriate.
3. The forward zone may be outsourced to a third party, such as a domain registrar. In this case, the use of the JSON-serialized YANG data model described in section Section 14 is appropriate, as it can easily be copy and pasted by the user, or downloaded as part of a web transaction.

For reverse zones, the relationship is always with the upstream ISP (although there may be more than one), and so [I-D.ietf-homenet-naming-architecture-dhc-options] is always the appropriate interface.

Appendix D. Example: A manufacturer provisioned HNA product flow

This scenario is one where a homenet router device manufacturer decides to offer DNS hosting as a value add.

[I-D.richardson-homerouter-provisioning] describes a process for a home router credential provisioning system. The outline of it is that near the end of the manufacturing process, as part of the

firmware loading, the manufacturer provisions a private key and certificate into the device.

In addition to having a asymmetric credential known to the manufacturer, the device also has been provisioned with an agreed upon name. In the example in the above document, the name "n8d234f.r.example.net" has already been allocated and confirmed with the manufacturer.

The HNA can use the above domain for itself. It is not very pretty or personal, but if the owner wishes a better name, they can arrange for it.

The configuration would look like:

```
{
  "dm_notify" : "2001:db8:1234:111:222::2",
  "dm_acl"    : "2001:db8:1234:111:222::/64",
  "dm_ctrl"   : "manufacturer.example.net",
  "dm_port"   : "4433",
  "ns_list"   : [ "ns1.publicdns.example", "ns2.publicdns.example"],
  "zone"      : "n8d234f.r.example.net",
  "auth_method" : "certificate",
  "hna_certificate": "-----BEGIN CERTIFICATE-----\nMIIDTjCCFGy....",
}
```

The dm_ctrl and dm_port values would be built into the firmware.

Authors' Addresses

Daniel Migault
Ericsson
8275 Trans Canada Route
Saint Laurent, QC 4S 0B6
Canada

Email: daniel.migault@ericsson.com

Ralf Weber
Nominum
2000 Seaport Blvd
Redwood City 94063
US

Email: ralf.weber@nominum.com

Michael Richardson
Sandelman Software Works
470 Dawson Avenue
Ottawa, ON K1Z 5V7
Canada

EMail: mcr+ietf@sandelman.ca

Ray Hunter
Globis Consulting BV
Weegschaalstraat 3
Eindhoven 5632CW
NL

EMail: v6ops@globis.net

Chris Griffiths

EMail: cgriffiths@gmail.com

Wouter Cloetens
Deutsche Telekom

EMail: wouter.cloetens@external.telekom.de

Homenet
Internet-Draft
Intended status: Standards Track
Expires: November 14, 2021

D. Migault
Ericsson
R. Weber
Nominum
M. Richardson
Sandelman Software Works
R. Hunter
Globis Consulting BV
May 13, 2021

Simple Provisioning of Public Names for Residential Networks
draft-ietf-homenet-front-end-naming-delegation-15

Abstract

Home owners often have IPv6 devices that they wish to access over the Internet using names.

Outsourcing the DNS servers to a DNS infrastructure protects against possible DDoS attacks as well as sudden renumbering of the home network. It has been possible to register and populate a DNS Zone with names since DNS became a thing, but it has been an activity typically reserved for experts. This document automates the process through creation of a Homenet Naming Authority (HNA), whose responsibility is to select, sign and publish names to a set of publicly visible servers.

This document describes the mechanism that enables the HNA to outsource the naming service to the DNS Outsourcing Infrastructure (DOI) via a Distribution Manager (DM).

In addition, this document deals with publication of a corresponding reverse zone.

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 <https://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 November 14, 2021.

Copyright Notice

Copyright (c) 2021 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 (<https://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
1.1. Selecting Names to Publish	5
1.2. Alternative solutions	6
2. Terminology	7
3. Architecture Description	8
3.1. Architecture Overview	8
3.2. Distribution Manager Communication Channels	11
4. Control Channel	12
4.1. Information to Build the Public Homenet Zone	13
4.2. Information to build the DNSSEC chain of trust	13
4.3. Information to set the Synchronization Channel	14
4.4. Deleting the delegation	14
4.5. Messages Exchange Description	14
4.5.1. Retrieving information for the Public Homenet Zone.	14
4.5.2. Providing information for the DNSSEC chain of trust	15
4.5.3. Providing information for the Synchronization Channel	16
4.5.4. HNA instructing deleting the delegation	17
4.6. Securing the Control Channel	17
4.7. Implementation Concerns	18
5. Synchronization Channel	19
5.1. Securing the Synchronization Channel	20
6. DM Distribution Channel	20
7. HNA Security Policies	20
8. DNSSEC compliant Homenet Architecture	21
9. Homenet Reverse Zone Channels Configuration	21
10. Homenet Public Zone Channel Configurations	22
11. Renumbering	24
11.1. Hidden Primary	24
12. Privacy Considerations	25

13. Security Considerations	26
13.1. HNA DM channels	26
13.2. Names are less secure than IP addresses	26
13.3. Names are less volatile than IP addresses	27
14. Information Model for Outsourced information	27
14.1. Outsourced Information Model	28
15. IANA Considerations	30
16. Acknowledgment	30
17. Contributors	30
18. References	31
18.1. Normative References	31
18.2. Informative References	34
Appendix A. Envisioned deployment scenarios	36
A.1. CPE Vendor	36
A.2. Agnostic CPE	36
Appendix B. Example: A manufacturer provisioned HNA product flow	37
Authors' Addresses	38

1. Introduction

The Homenet Naming Authority (HNA) is responsible for making devices within the home network accessible by a public name within the home network as well as from outside the home network (e.g. the Internet). IPv6 connectivity provides the possibility of global end to end IP connectivity.

The use of a DNS zone for each home network is a reasonable and scalable way to make the set of public names visible. There are a number of ways to populate such a zone. This specification proposes a way based on a number of assumptions about typical home networks.

1. The names of the devices accessible from the Internet are stored in the Public Homenet Zone, served by a DNS authoritative server.
2. It is unlikely that home networks will contain sufficiently robust platforms designed to host and expose to the Internet a service such as the DNS and as such would expose the home network to DDoS attacks.
3. [RFC7368] emphasizes that the home network is subject to connectivity disruptions with the ISP. But, names used within the home MUST be resilient against such disruption.

This specification makes the public names resolvable within both the home network and on the Internet, even when there are disruptions.

This is achieved by having a function inside the home network that builds, signs, publishes, and manages a Public Homenet Zone, thus

providing bindings between public names, IP addresses, and other RR types.

The management of the names can be under the responsibility the Customer Premises Equipment (CPE) does. Other devices within the home network could fulfill this role e.g. a Network Attached Storage server, but for simplicity, this document assumes the function is located on one of the CPE devices.

The homenet architecture [RFC7368] makes it clear that a home network may have multiple CPEs. The management of the Public Homenet Zone involves DNS specific mechanisms that cannot be distributed over multiple servers (primary server), when multiple nodes can potentially manage the Public Homenet Zone, a single node needs to be selected per outsourced zone. This selected node is designated as providing the HNA function.

The process by which a single HNA is selected per zone is not in scope for this document. It is envisioned that a future document will describe an HNCP mechanism to elect the single HNA.

CPEs, which may host the HNA function are usually low powered devices not designed for supporting a heavy traffic. As a result, hosting an authoritative DNS service visible to the Internet may expose the home network to resource exhaustion and other attacks. On the other hand, if the only copy of the public zone is on the Internet, then Internet connectivity disruptions would make the names unavailable inside the homenet.

In order to avoid resource exhaustion and other attacks, this document describes in Section 3.1 an architecture that outsources the authoritative naming service of the home network. More specifically, the HNA builds the Public Homenet Zone and outsources it to a DNS Outsourcing Infrastructure (DOI) via a Distribution Manager (DM). The DOI is in charge of publishing the corresponding Public Homenet Zone on the Internet. The transfer of DNS zone information is achieved using standard DNS mechanisms involving primary and secondary DNS servers, with the HNA being a stealth primary, and the DM a secondary.

In order to keep the Public Homenet Zone up-to-date Section 5 describes how the HNA and the DOI synchronize the Public Homenet Zone.

The architecture is explicitly designed to enable fully functional DNSSEC, and the Public Homenet Zone is expected to be signed with a secure delegation. DNSSEC key management and zone signing are handled by the HNA.

Section 10 discusses management and configuration of the Public Homenet Zone. It shows that the HNA configuration of the DOI can involve no or little interaction with the end user. More specifically, it shows that the existence of an account in the DOI is sufficient for the DOI to push the necessary configuration. In addition, when the DOI and CPE are both managed by an ISP, the configuration can be entirely automated - see Section 9.

Section 9 discusses management of one or more reverse zones. It shows that management of the reverse zones can be entirely automated and benefit from a pre-established relation between the ISP and the home network. Note that such scenarios may also be met for the Public Homenet Zone.

Section 11 discusses how renumbering should be handled. Finally, Section 12 and Section 13 respectively discuss privacy and security considerations when outsourcing the Public Homenet Zone.

The Public Homenet Zone is expected to contain public information only in a single universal view. This document does not define how the information required to construct this view is derived.

It is also not in the scope of this document to define names for exclusive use within the boundaries of the local home network. Instead, local scope information is expected to be provided to the home network using local scope naming services. mDNS [RFC6762] and DNS-SD [RFC6763] are two examples of these services. Currently mDNS is limited to a single link network. However, future protocols and architectures [I-D.ietf-homenet-simple-naming] are expected to leverage this constraint as pointed out in [RFC7558].

1.1. Selecting Names to Publish

While this document does not create any normative mechanism by which the selection of names to publish, this document anticipates that the home network administrator (a human), will be presented with a list of current names and addresses present on the inside of the home network.

The administrator would mark which devices (by name), are to be published. The HNA would then collect the IPv6 address(es) associated with that device, and put the name into the Public Homenet Zone. The address of the device can be collected from a number of places: mDNS [RFC6762], DHCP [RFC6644], UPnP, PCP [RFC6887], or manual configuration.

A device may have a Global Unicast Address (GUA), a Unique Local IPv6 Address (ULA), as well as IPv6-Link-Local addresses, IPv4-Link-Local

Addresses, and RFC1918 addresses. Of these the link-local are never useful for the Public Zone, and should be omitted. The IPv6 ULA and the RFC1918 addresses may be useful to publish, if the home network environment features a VPN that would allow the home owner to reach the network.

The IPv6 ULA addressees are significantly safer to publish, as the RFC1918 addressees are likely to be confusing to any other entity.

In general, one expects the GUA to be the default address to be published. However, during periods when the home network has connectivity problems, the ULA and RFC1918 addressees can be used inside the home, and the mapping from public name to locally useful location address would permit many services secured with HTTPS to continue to operate.

1.2. Alternative solutions

An alternative existing solution in IPv4 is to have a single zone, where a host uses a RESTful HTTP service to register a single name into a common public zone. This is often called "Dynamic DNS", and there are a number of commercial providers, including Dyn, Gandi etc. These solutions were typically used by a host behind the CPE to make its CPE IPv4 address visible, usually in order to enable incoming connections. This is the most common scenario considered in this section, while some variant may also consider the client being hosted in the CPE.

For a very few number (one to three) of hosts, the use of such a system provides an alternative to the architecture described in this document. The alternative - even adapted to IPv6 and ignoring those associated to an IPv4 development - does suffer from some severe limitations:

- o the CPE/HNA router is unaware of the process, and cannot respond to queries for these names when there are disruptions in connectivity. This makes the home user or application dependent on having to resolve different names in the event of outages or disruptions.
- o the CPE/HNA router cannot control the process. Any host can do this regardless of whether or not the home network administrator wants the name published or not. There is therefore no possible audit trail.
- o the credentials for the dynamic DNS server need to be securely transferred to all hosts that wish to use it. This is not a problem for a technical user to do with one or two hosts, but it

does not scale to multiple hosts and becomes a problem for non-technical users.

- o "all the good names are taken" - current services put everyone's names into some small set of zones, and there are often conflicts. Distinguishing similar names by delegation of zones was among the primary design goals of the DNS system.
- o The RESTful services do not always support all RR types. The homenet user is dependent on the service provider supporting new types. By providing full DNS delegation, this document enables all RR types and also future extensions.

There is no technical reason why a RESTful service could not provide solutions to many of these problems, but this document describes a DNS-based solution.

2. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Customer Premises Equipment: (CPE) is a router providing connectivity to the home network.

Homenet Zone: is the DNS zone for use within the boundaries of the home network: 'home.arpa' (see [RFC8375]). This zone is not considered public and is out of scope for this document.

Registered Homenet Domain: is the domain name that is associated with the home network.

Public Homenet Zone: contains the names in the home network that are expected to be publicly resolvable on the Internet.

Homenet Naming Authority(HNA): is a function responsible for managing the Public Homenet Zone. This includes populating the Public Homenet Zone, signing the zone for DNSSEC, as well as managing the distribution of that Homenet Zone to the DNS Outsourcing Infrastructure (DOI).

DNS Outsourcing Infrastructure (DOI): is the infrastructure responsible for receiving the Public Homenet Zone and publishing it on the Internet. It is mainly composed of a Distribution Manager and Public Authoritative Servers.

Public Authoritative Servers: are the authoritative name servers for the Public Homenet Zone. Name resolution requests for the Homenet Domain are sent to these servers. For resiliency the Public Homenet Zone SHOULD be hosted on multiple servers.

Homenet Authoritative Servers: are authoritative name servers within the Homenet network.

Distribution Manager (DM): is the (set of) server(s) to which the HNA synchronizes the Public Homenet Zone, and which then distributes the relevant information to the Public Authoritative Servers.

Homenet Reverse Zone: The reverse zone file associated with the Public Homenet Zone.

Reverse Public Authoritative Servers: equivalent to Public Authoritative Servers specifically for reverse resolution.

Reverse Distribution Manager: equivalent to Distribution Manager specifically for reverse resolution.

Homenet DNSSEC Resolver: a resolver that performs a DNSSEC resolution on the home network for the Public Homenet Zone. The resolution is performed requesting the Homenet Authoritative Servers.

DNSSEC Resolver: a resolver that performs a DNSSEC resolution on the Internet for the Public Homenet Zone. The resolution is performed requesting the Public Authoritative Servers.

3. Architecture Description

This section provides an overview of the architecture for outsourcing the authoritative naming service from the HNA to the DOI. Note that Section 14 defines necessary parameter to configure the HNA.

3.1. Architecture Overview

Figure 1 illustrates the architecture where the HNA outsources the publication of the Public Homenet Zone to the DOI.

The Public Homenet Zone is identified by the Registered Homenet Domain Name - myhome.example. The ".local" as well as ".home.arpa" are explicitly not considered as Public Homenet zones and represented as Homenet Zone in Figure 1.

The HNA SHOULD build the Public Homenet Zone in a single view populated with all resource records that are expected to be published on the Internet. The HNA also signs the Public Homenet Zone. The HNA handles all operations and keying material required for DNSSEC, so there is no provision made in this architecture for transferring private DNSSEC related keying material between the HNA and the DM.

Once the Public Homenet Zone has been built, the HNA outsources it to the DOI as described in Figure 1. The HNA acts as a hidden primary [RFC8499] while the DM behaves as a secondary responsible to distribute the Public Homenet Zone to the multiple Public Authoritative Servers that DOI is responsible for. The DM has three communication channels:

- o a DM Control Channel (Section 4) to configure the HNA and the DOI,
- o a DM Synchronization Channel (Section 5) to synchronize the Public Homenet Zone on the HNA and on the DM,
- o one or more Distribution Channels (Section 6) that distribute the Public Homenet Zone from the DM to the Public Authoritative Server serving the Public Homenet Zone on the Internet.

There might be multiple DM's, and multiple servers per DM. This document assumes a single DM server for simplicity, but there is no reason why each channel needs to be implemented on the same server or use the same code base.

It is important to note that while the HNA is configured as an authoritative server, it is not expected to answer to DNS requests from the public Internet for the Public Homenet Zone. More specifically, the addresses associated with the HNA SHOULD NOT be mentioned in the NS records of the Public Homenet zone, unless additional security provisions necessary to protect the HNA from external attack have been taken.

The DOI is also responsible for ensuring the DS record has been updated in the parent zone.

Resolution is performed by the DNSSEC resolvers. When the resolution is performed outside the home network, the DNSSEC Resolver resolves the DS record on the Global DNS and the name associated to the Public Homenet Zone (myhome.example) on the Public Authoritative Servers.

When the resolution is performed from within the home network, the Homenet DNSSEC Resolver MAY proceed similarly. On the other hand, to provide resilience to the Public Homenet Zone in case of WAN connectivity disruption, the Homenet DNSSEC Resolver SHOULD be able

to perform the resolution on the Homenet Authoritative Servers. These servers are not expected to be mentioned in the Public Homenet Zone, nor to be accessible from the Internet. As such their information as well as the corresponding signed DS record MAY be provided by the HNA to the Homenet DNSSEC Resolvers, e.g., using HNCP [RFC7788]. Such configuration is outside the scope of this document. Since the scope of the Homenet Authoritative Servers is limited to the home network, these servers are expected to serve the Homenet Zone as represented in Figure 1.

How the Homenet Authoritative Servers are provisioned is also out of scope of this specification. It could be implemented using primary and secondary servers, or via rsync. In some cases, the HNA and Homenet Authoritative Servers may be combined together which would result in a common instantiation of an authoritative server on the WAN and inner homenet interface. Note that [RFC6092] REC-8 states this must not be the default configuration. Other mechanisms may also be used.

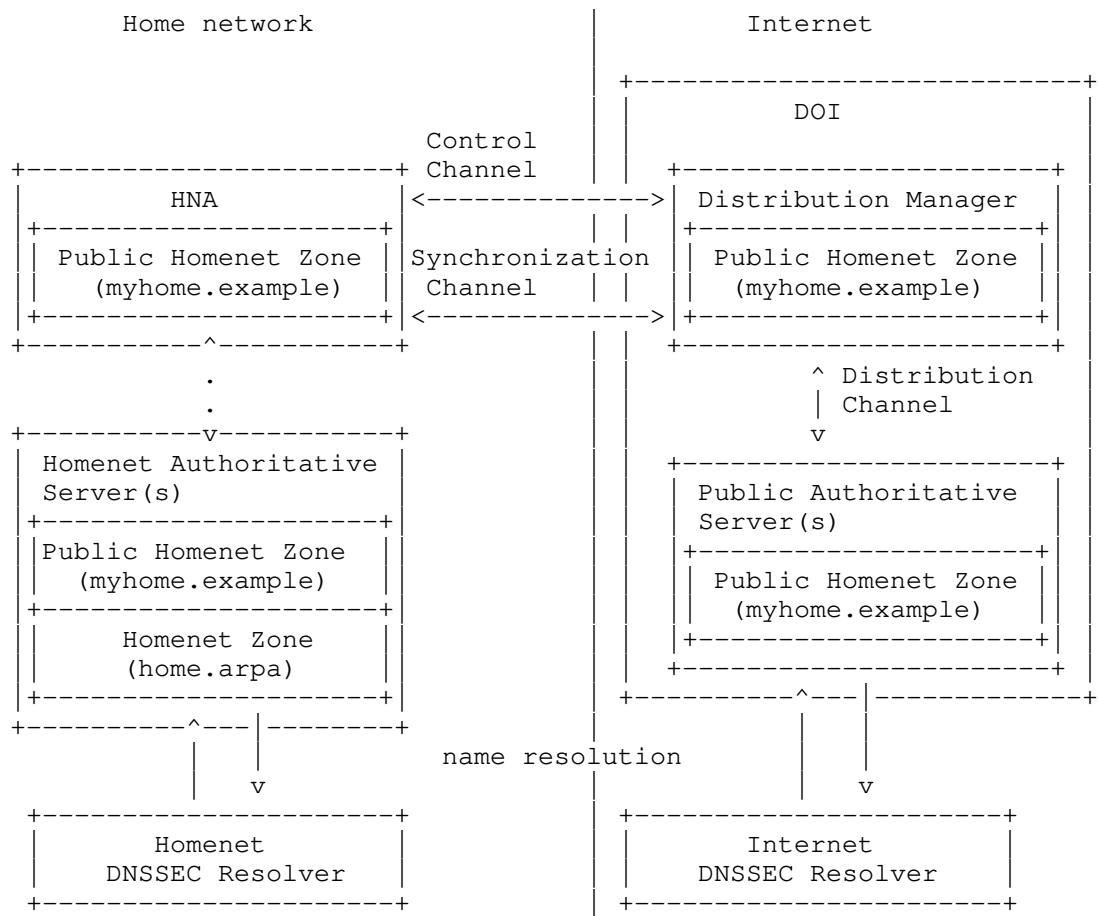


Figure 1: Homenet Naming Architecture

3.2. Distribution Manager Communication Channels

This section details the DM channels, that is the Control Channel, the Synchronization Channel and the Distribution Channel.

The Control Channel and the Synchronization Channel are the interfaces used between the HNA and the DOI. The entity within the DOI responsible to handle these communications is the DM and communications between the HNA and the DM MUST be protected and mutually authenticated. While Section 4.6 discusses in more depth the different security protocols that could be used to secure, it is RECOMMENDED to use TLS with mutually authentication based on certificates to secure the channel between the HNA and the DM.

The Control Channel is used to set up the Synchronization Channel. We assume that the HNA initiates the Control Channel connection with the DM and as such has a prior knowledge of the DM identity (X509 certificate), the IP address and port number to use and protocol to set secure session. We also assume the DM has knowledge of the identity of the HNA (X509 certificate) as well as the Registered Homenet Domain. For more detail to see how this can be achieved, please see Section 10.

The information exchanged between the HNA and the DM uses DNS messages protected by DNS over TLS (DoT) [RFC7858]. Other specifications may consider protecting DNS messages with other transport layers, among others, DNS over DTLS [RFC8094], or DNS over HTTPs (DoH) [RFC8484] or DNS over QUIC [I-D.ietf-dprive-dnsquic]. There was consideration to using a standard TSIG [RFC2845] or SIG(0) [RFC2931] to perform a dynamic DNS update to the DM. There are a number of issues with this. The first one is that TSIG or SIG(0) make scenarios where the end user needs to interact via its web browser more complex. More precisely, authorization and access control granted via OAUTH would be unnecessarily complex with TSIG or SIG(0).

The main issue is that the Dynamic DNS update would also update the parent zone's (NS, DS and associated A or AAAA records) while the goal is to update the DM configuration files. The visible NS records SHOULD remain pointing at the cloud provider's anycast addresses. Revealing the address of the HNA in the DNS is not desirable. Refer to Section 4.2 for more details.

This specification assumes:

- o the DM serves both the Control Channel and Synchronization Channel on a single IP address, single port and using a single transport protocol.
- o By default, the HNA uses a single IP address for both the Control and Synchronization channel. However, the HNA MAY use distinct IP addresses for the Control Channel and the Synchronization Channel - see Section 5 and Section 4.3 for more details.

The Distribution Channel is internal to the DOI and as such is not the primary concern of this specification.

4. Control Channel

The DM Control Channel is used by the HNA and the DOI to exchange information related to the configuration of the delegation which includes information to build the Public Homenet Zone (Section 4.1),

information to build the DNSSEC chain of trust (Section 4.2) and information to set the Synchronization Channel (Section 4.3).

4.1. Information to Build the Public Homenet Zone

When the HNA builds the Public Homenet Zone, it must include information that it retrieves from the DM relating to how the zone is to be published.

The information includes at least names and IP addresses of the Public Authoritative Name Servers. In term of RRset information this includes:

- o the MNAME of the SOA,
- o the NS and associated A and AAA RRsets of the name servers.

The DOI MAY also provide operational parameters such as other fields of SOA (SERIAL, RNAME, REFRESH, RETRY, EXPIRE and MINIMUM). As the information is necessary for the HNA to proceed and the information is associated to the DOI, this information exchange is mandatory.

4.2. Information to build the DNSSEC chain of trust

The HNA SHOULD provide the hash of the KSK (DS RRset), so the that DOI provides this value to the parent zone. A common deployment use case is that the DOI is the registrar of the Registered Homenet Domain, and as such, its relationship with the registry of the parent zone enables it to update the parent zone. When such relation exists, the HNA should be able to request the DOI to update the DS RRset in the parent zone. A direct update is especially necessary to initialize the chain of trust.

Though the HNA may also later directly update the values of the DS via the Control Channel, it is RECOMMENDED to use other mechanisms such as CDS and CDNSKEY [RFC7344] for transparent updates during key roll overs.

As some deployments may not provide a DOI that will be able to update the DS in the parent zone, this information exchange is OPTIONAL.

By accepting the DS RR, the DM commits in taking care of advertising the DS to the parent zone. Upon refusal, the DM clearly indicates it does not have the capacity to proceed to the update.

4.3. Information to set the Synchronization Channel

The HNA works as a primary authoritative DNS server, while the DM works like a secondary. As a result, the HNA MUST provide the IP address the DM is using to reach the HNA. The synchronization Channel will be set between that IP address and the IP address of the DM. By default, the IP address used by the HNA in the Control Channel is considered by the DM and the specification of the IP by the HNA is only OPTIONAL. The transport channel (including port number) is the same as the one used between the HNA and the DM for the Control Channel.

4.4. Deleting the delegation

The purpose of the previous sections were to exchange information in order to set a delegation. The HNA MUST also be able to delete a delegation with a specific DM. Upon an instruction of deleting the delegation, the DM MUST stop serving the Public Homenet Zone.

4.5. Messages Exchange Description

There are multiple ways this information could be exchanged between the HNA and the DM. This specification defines a mechanism that re-use the DNS exchanges format. The intention is to reuse standard libraries especially to check the format of the exchanged fields as well as to minimize the additional libraries needed for the HNA. The re-use of DNS exchanges achieves these goals. Note that while information is provided using DNS exchanges, the exchanged information is not expected to be set in any zone file, instead this information is uses as commands between the HNA and the DM.

The Control Channel is not expected to be a long term session. After a predefined timer - similar to those used for TCP - the Control Channel is expected to be terminated - by closing the transport channel. The Control Channel MAY be re-opened at any time later.

The provisioning process SHOULD provide a method of securing the Control Channel, so that the content of messages can be authenticated. This authentication MAY be based on certificates for both the DM and each HNA. The DM may also create the initial configuration for the delegation zone in the parent zone during the provisioning process.

4.5.1. Retrieving information for the Public Homenet Zone.

The information provided by the DM to the HNA is retrieved by the HNA with an AXFR exchange [RFC1034]. AXFR enables the response to contain any type of RRsets. The response might be extended in the

future if additional information will be needed. Alternatively, the information provided by the HNA to the DM is pushed by the HNA via a DNS update exchange [RFC2136].

To retrieve the necessary information to build the Public Homenet Zone, the HNA MUST send a DNS request of type AXFR associated to the Registered Homenet Domain. The DM MUST respond with a zone template. The zone template MUST contain a RRset of type SOA, one or multiple RRset of type NS and zero or more RRset of type A or AAAA.

- o The SOA RR indicates to the HNA the value of the MNAME of the Public Homenet Zone.
- o The NAME of the SOA RR MUST be the Registered Homenet Domain.
- o The MNAME value of the SOA RDATA is the value provided by the DOI to the HNA.
- o Other RDATA values (RNAME, REFRESH, RETRY, EXPIRE and MINIMUM) are provided by the DOI as suggestions.

The NS RRsets carry the Public Authoritative Servers of the DOI. Their associated NAME MUST be the Registered Homenet Domain.

The TTL and RDATA are those expected to be published on the Public Homenet Zone. The RRsets of Type A and AAAA MUST have their NAME matching the NSDNAME of one of the NS RRsets.

Upon receiving the response, the HNA MUST validate format and properties of the SOA, NS and A or AAAA RRsets. If an error occurs, the HNA MUST stop proceeding and MUST log an error. Otherwise, the HNA builds the Public Homenet Zone by setting the MNAME value of the SOA as indicated by the SOA provided by the AXFR response. The HNA SHOULD set the value of NAME, REFRESH, RETRY, EXPIRE and MINIMUM of the SOA to those provided by the AXFR response. The HNA MUST insert the NS and corresponding A or AAAA RRset in its Public Homenet Zone. The HNA MUST ignore other RRsets. If an error message is returned by the DM, the HNA MUST proceed as a regular DNS resolution. Error messages SHOULD be logged for further analysis. If the resolution does not succeed, the outsourcing operation is aborted and the HNA MUST close the Control Channel.

4.5.2. Providing information for the DNSSEC chain of trust

To provide the DS RRset to initialize the DNSSEC chain of trust the HNA MAY send a DNS update [RFC2136] message.

The DNS update message is composed of a Header section, a Zone section, a Pre-requisite section, and Update section and an additional section. The Zone section MUST set the ZNAME to the parent zone of the Registered Homenet Domain - that is where the DS records should be inserted. As described [RFC2136], ZTYPE is set to SOA and ZCLASS is set to the zone's class. The Pre-requisite section MUST be empty. The Update section is a DS RRset with its NAME set to the Registered Homenet Domain and the associated RDATA corresponds to the value of the DS. The Additional Data section MUST be empty.

Though the pre-requisite section MAY be ignored by the DM, this value is fixed to remain coherent with a standard DNS update.

Upon receiving the DNS update request, the DM reads the DS RRset in the Update section. The DM checks ZNAME corresponds to the parent zone. The DM SHOULD ignore non empty the Pre-requisite and Additional Data section. The DM MAY update the TTL value before updating the DS RRset in the parent zone. Upon a successful update, the DM should return a NOERROR response as a commitment to update the parent zone with the provided DS. An error indicates the MD does not update the DS, and other method should be used by the HNA.

The regular DNS error message SHOULD be returned to the HNA when an error occurs. In particular a FORMERR is returned when a format error is found, this includes when unexpected RRsets are added or when RRsets are missing. A SERVFAIL error is returned when a internal error is encountered. A NOTZONE error is returned when update and Zone sections are not coherent, a NOTAUTH error is returned when the DM is not authoritative for the Zone section. A REFUSED error is returned when the DM refuses to proceed to the configuration and the requested action.

4.5.3. Providing information for the Synchronization Channel

The default IP address used by the HNA for the Synchronization Channel is the IP address of the Control Channel. To provide a different IP address, the HNA MAY send a DNS Update message.

Similarly to the Section 4.5.2, the HNA MAY specify the IP address using a DNS update message. The Zone section sets its ZNAME to the parent zone of the Registered Homenet Domain, ZTYPE is set to SOA and ZCLASS is set to the zone's type. Pre-requisite is empty. The Update section is a RRset of type NS. The Additional Data section contains the RRsets of type A or AAAA that designates the IP addresses associated to the primary (or the HNA).

The reason to provide these IP addresses is to keep them unpublished and prevent them to be resolved.

Upon receiving the DNS update request, the DM reads the IP addresses and checks the ZNAME corresponds to the parent zone. The DM SHOULD ignore a non empty Pre-requisite section. The DM configures the secondary with the IP addresses and returns a NOERROR response to indicate it is committed to serve as a secondary.

Similarly to Section 4.5.2, DNS errors are used and an error indicates the DM is not configured as a secondary.

4.5.4. HNA instructing deleting the delegation

To instruct to delete the delegation the HNA SHOULD send a DNS UPDATE Delete message.

The Zone section sets its ZNAME to the Registered Homenet Domain, the ZTYPE to SOA and the ZCLASS to zone's type. The Pre-requisite section is empty. The Update section is a RRset of type NS with the NAME set to the Registered Domain Name. As indicated by [RFC2136] Section 2.5.2 the delete instruction is set by setting the TTL to 0, the Class to ANY, the RDLENGTH to 0 and the RDATA MUST be empty. The Additional Data section is empty.

Upon receiving the DNS update request, the DM checks the request and removes the delegation. The DM returns a NOERROR response to indicate the delegation has been deleted. Similarly to Section 4.5.2, DNS errors are used and an error indicates the delegation has not been deleted.

4.6. Securing the Control Channel

The control channel between the HNA and the DM MUST be secured at both the HNA and the DM.

Secure protocols (like TLS [RFC8446] SHOULD be used to secure the transactions between the DM and the HNA.

The advantage of TLS is that this technology is widely deployed, and most of the devices already embed TLS libraries, possibly also taking advantage of hardware acceleration. Further, TLS provides authentication facilities and can use certificates to mutually authenticate the DM and HNA at the application layer, including available API. On the other hand, using TLS requires implementing DNS exchanges over TLS, as well as a new service port.

The HNA SHOULD authenticate inbound connections from the DM using standard mechanisms, such as a public certificate with baked-in root certificates on the HNA, or via DANE [RFC6698]. The HNA is expected

to be provisioned with a connection to the DM by the manufacturer, or during some user-initiated onboarding process, see Section 10.

The DM SHOULD authenticate the HNA and check that inbound messages are from the appropriate client. The DM MAY use a self-signed CA certificate mechanism per HNA, or public certificates for this purpose.

IPsec [RFC4301] and IKEv2 [RFC7296] were considered. They would need to operate in transport mode, and the authenticated end points would need to be visible to the applications, and this is not commonly available at the time of this writing.

A pure DNS solution using TSIG and/or SIG(0) to authenticate message was also considered. Section 10 envisions one mechanism would involve the end user, with a browser, signing up to a service provider, with a resulting OAuth2 token to be provided to the HNA. A way to translate this OAuth2 token from HTTPS web space to DNS SIG(0) space seems overly problematic, and so the enrollment protocol using web APIs was determined to be easier to implement at scale.

Note also that authentication of message exchanges between the HNA and the DM SHOULD NOT use the external IP address of the HNA to index the appropriate keys. As detailed in Section 11, the IP addresses of the DM and the hidden primary are subject to change, for example while the network is being renumbered. This means that the necessary keys to authenticate transaction SHOULD NOT be indexed using the IP address, and SHOULD be resilient to IP address changes.

4.7. Implementation Concerns

The Hidden Primary Server on the HNA differs from a regular authoritative server for the home network due to:

Interface Binding: the Hidden Primary Server will almost certainly listen on the WAN Interface, whereas a regular Homenet Authoritative Servers would listen on the internal home network interface.

Limited exchanges: the purpose of the Hidden Primary Server is to synchronize with the DM, not to serve any zones to end users, or the public Internet.

As a result, exchanges are performed with specific nodes (the DM). Further, exchange types are limited. The only legitimate exchanges are: NOTIFY initiated by the Hidden Primary and IXFR or AXFR exchanges initiated by the DM.

On the other hand, regular authoritative servers would respond to any hosts, and any DNS query would be processed. The HNA SHOULD filter IXFR/AXFR traffic and drop traffic not initiated by the DM. The HNA MUST at least allow SOA lookups of the Homenet Zone.

5. Synchronization Channel

The DM Synchronization Channel is used for communication between the HNA and the DM for synchronizing the Public Homenet Zone. Note that the Control Channel and the Synchronization Channel are by construction different channels even though there they may use the same IP address. In fact the Control Channel is set between the HNA working as a client using port number YYYY (a high range port) toward a service provided by the DM at port number XX (well known port).

On the other hand, the Synchronization Channel is set between the DM working as a client using port ZZZZ (a high range port) toward a service a service provided by the HNA at port XX.

As a result, even though the same couple of IP addresses may be involved the Control Channel and the Synchronization Channel are always distinct channels.

Uploading and dynamically updating the zone file on the DM can be seen as zone provisioning between the HNA (Hidden Primary) and the DM (Secondary Server). This can be handled via AXFR + DNS Update.

The use of a primary / secondary mechanism is RECOMMENDED instead of the use of DNS Update. The primary / secondary mechanism is RECOMMENDED as it scales better and avoids DoS attacks. Note that even when UPDATE messages are used, these messages are using a distinct channel as those used to set the configuration.

Note that there is no standard way to distribute a DNS primary between multiple devices. As a result, if multiple devices are candidate for hosting the Hidden Primary, some specific mechanisms should be designed so the home network only selects a single HNA for the Hidden Primary. Selection mechanisms based on HNCP [RFC7788] are good candidates.

The HNA acts as a Hidden Primary Server, which is a regular authoritative DNS Server listening on the WAN interface.

The DM is configured as a secondary for the Registered Homenet Domain Name. This secondary configuration has been previously agreed between the end user and the provider of the DOI as part of either the provisioning or due to receipt of DNS Update messages on the DM Control Channel.

The Homenet Reverse Zone MAY also be updated either with DNS UPDATE [RFC2136] or using a primary / secondary synchronization.

5.1. Securing the Synchronization Channel

The Synchronization Channel uses standard DNS requests.

First the primary notifies the secondary that the zone must be updated and eaves the secondary to proceed with the update when possible/convenient.

Then, a NOTIFY message is sent by the primary, which is a small packet that is less likely to load the secondary.

Finally, the AXFR [RFC1034] or IXFR [RFC1995] query performed by the secondary is a small packet sent over TCP (Section 4.2 [RFC5936]), which mitigates reflection attacks using a forged NOTIFY.

The AXFR request from the DM to the HNA SHOULD be secured and the use of TLS is RECOMMENDED [I-D.ietf-dprive-xfr-over-tls]

When using TLS, the HNA MAY authenticate inbound connections from the DM using standard mechanisms, such as a public certificate with baked-in root certificates on the HNA, or via DANE [RFC6698]. In addition, to guarantee the DM remains the same across multiple TLS session, the HNA and DM MAY implement [RFC8672].

The HNA SHOULD apply an ACL on inbound AXFR requests to ensure they only arrive from the DM Synchronization Channel. In this case, the HNA SHOULD regularly check (via DNS resolution) that the address of the DM in the filter is still valid.

6. DM Distribution Channel

The DM Distribution Channel is used for communication between the DM and the Public Authoritative Servers. The architecture and communication used for the DM Distribution Channels is outside the scope of this document, and there are many existing solutions available e.g. rsynch, DNS AXFR, REST, DB copy.

7. HNA Security Policies

This section details security policies related to the Hidden Primary / Secondary synchronization.

The HNA, as Hidden Primary SHOULD drop any queries from the home network. This could be implemented via port binding and/or firewall rules. The precise mechanism deployed is out of scope of this

document. The Hidden Primary SHOULD drop any DNS queries arriving on the WAN interface that are not issued from the DM. The Hidden Primary SHOULD drop any outgoing packets other than DNS NOTIFY query, SOA response, IXFR response or AXFR responses. The Hidden Primary SHOULD drop any incoming packets other than DNS NOTIFY response, SOA query, IXFR query or AXFR query. The Hidden Primary SHOULD drop any non protected IXFR or AXFR exchange, depending on how the synchronization is secured.

8. DNSSEC compliant Homenet Architecture

[RFC7368] in Section 3.7.3 recommends DNSSEC to be deployed on both the authoritative server and the resolver. The resolver side is out of scope of this document, and only the authoritative part of the server is considered.

This document assumes the HNA signs the Public Homenet Zone.

Secure delegation is achieved only if the DS RRset is properly set in the parent zone. Secure delegation is performed by the HNA or the DOIs.

The DS RRset can be updated manually with nsupdate for example. This requires the HNA or the DOI to be authenticated by the DNS server hosting the parent of the Public Homenet Zone. Such a trust channel between the HNA and the parent DNS server may be hard to maintain with HNAs, and thus may be easier to establish with the DOI. In fact, the Public Authoritative Server(s) may use Automating DNSSEC Delegation Trust Maintenance [RFC7344].

9. Homenet Reverse Zone Channels Configuration

The Public Homenet Zone is associated to a Registered Homenet Domain and the ownership of that domain requires a specific registration from the end user as well as the HNA being provisioned with some authentication credentials. Such steps are mandatory unless the DOI has some other means to authenticate the HNA. Such situation may occur, for example, when the ISP provides the Homenet Domain as well as the DOI.

In this case, the HNA may be authenticated by the physical link layer, in which case the authentication of the HNA may be performed without additional provisioning of the HNA. While this may not be so common for the Public Homenet Zone, this situation is expected to be quite common for the Reverse Homenet Zone.

More specifically, a common case is that the upstream ISP provides the IPv6 prefix to the Homenet with a IA_PD [RFC8415] option and manages the DOI of the associated reverse zone.

This leave place for setting up automatically the relation between HNA and the DNS Outsourcing infrastructure as described in [I-D.ietf-homenet-naming-architecture-dhc-options].

In the case of the reverse zone, the DOI authenticates the source of the updates by IPv6 Access Control Lists. In the case of the reverse zone, the ISP knows exactly what addresses have been delegated. The HNA SHOULD therefore always originate Synchronization Channel updates from an IP address within the zone that is being updated.

For example, if the ISP has assigned 2001:db8:f00d::/64 to the WAN interface (by DHCPv6, or PPP/RA), then the HNA should originate Synchronization Channel updates from, for example, 2001:db8:f00d::2.

An ISP that has delegated 2001:db8:babe::/56 to the HNA via DHCPv6-PD, then HNA should originate Synchronization Channel updates an IP within that subnet, such as 2001:db8:babe:0001::2.

With this relation automatically configured, the synchronization between the Home network and the DOI happens similarly as for the Public Homenet Zone described earlier in this document.

Note that for home networks connected to by multiple ISPs, each ISP provides only the DOI of the reverse zones associated to the delegated prefix. It is also likely that the DNS exchanges will need to be performed on dedicated interfaces as to be accepted by the ISP. More specifically, the reverse zone associated to prefix 1 will not be possible to be performs by the HNA using an IP address that belongs to prefix 2. Such constraints does not raise major concerns either for hot standby or load sharing configuration.

With IPv6, the domain space for IP addresses is so large that reverse zone may be confronted with scalability issues. How the reverse zone is generated is out of scope of this document. [RFC8501] provides guidance on how to address scalability issues.

10. Homenet Public Zone Channel Configurations

This document does not deal with how the HNA is provisioned with a trusted relationship to the Distribution Manager for the forward zone.

This section details what needs to be provisioned into the HNA and serves as a requirements statement for mechanisms.

The HNA needs to be provisioned with:

- o the Registered Domain (e.g., myhome.isp.example)
- o the contact info for the Distribution Manager (DM), including the DNS name (FQDN), possibly including the IP literal, and a certificate (or anchor) to be used to authenticate the service
- o the DM transport protocol and port (the default is DNS over TLS, on port 853)
- o the HNA credentials used by the DM for its authentication.

The HNA will need to select an IP address for communication for the Synchronization Channel. This is typically the WAN address of the RG router, but could be an IPv6 LAN address in the case of a home with multiple ISPs (and multiple border routers). This is detailed in Section 4.5.3 when the NS and A or AAAA RRsets are communicated.

The above parameters MUST be be provisioned for ISP-specific reverse zones, as per [I-D.ietf-homenet-naming-architecture-dhc-options]. ISP-specific forward zones MAY also be provisioned using [I-D.ietf-homenet-naming-architecture-dhc-options], but zones which are not related to a specific ISP zone (such as with a DNS provider) must be provisioned through other means.

Similarly, if the HNA is provided by a registrar, the HNA may be handed pre-configured to end user.

In the absence of specific pre-established relation, these pieces of information may be entered manually by the end user. In order to ease the configuration from the end user the following scheme may be implemented.

The HNA may present the end user a web interface where it provides the end user the ability to indicate the Registered Homenet Domain or the registrar for example a preselected list. Once the registrar has been selected, the HNA redirects the end user to that registrar in order to receive a access token. The access token will enable the HNA to retrieve the DM parameters associated to the Registered Domain. These parameters will include the credentials used by the HNA to establish the Control and Synchronization Channels.

Such architecture limits the necessary steps to configure the HNA from the end user.

11. Renumbering

This section details how renumbering is handled by the Hidden Primary server or the DM. Both types of renumbering are discussed i.e. "make-before-break" and "break-before-make" (aka flash renumbering).

In the make-before-break renumbering scenario, the new prefix is advertised, the network is configured to prepare the transition to the new prefix. During a period of time, the two prefixes old and new coexist, before the old prefix is completely removed.

In the break-before-make renumbering scenario, the new prefix is advertised making the old prefix obsolete.

Renumbering has been extensively described in [RFC4192] and analyzed in [RFC7010] and the reader is expected to be familiar with them before reading this section.

11.1. Hidden Primary

In a renumbering scenario, the HNA or Hidden Primary is informed it is being renumbered. In most cases, this occurs because the whole home network is being renumbered. As a result, the Public Homenet Zone will also be updated. Although the new and old IP addresses may be stored in the Public Homenet Zone, we recommend that only the newly reachable IP addresses be published.

To avoid reachability disruption, IP connectivity information provided by the DNS SHOULD be coherent with the IP in use. In our case, this means the old IP address SHOULD NOT be provided via the DNS when it is not reachable anymore. Let for example TTL be the TTL associated with a RRset of the Public Homenet Zone, it may be cached for TTL seconds. Let T_NEW be the time the new IP address replaces the old IP address in the Homenet Zone, and T_OLD_UNREACHABLE the time the old IP is not reachable anymore.

In the case of the make-before-break, seamless reachability is provided as long as $T_OLD_UNREACHABLE - T_NEW > 2 * TTL$. If this is not satisfied, then devices associated with the old IP address in the home network may become unreachable for $2 * TTL - (T_OLD_UNREACHABLE - T_NEW)$. In the case of a break-before-make, $T_OLD_UNREACHABLE = T_NEW$, and the device may become unreachable up to $2 * TTL$. Of course if $T_NEW \geq T_OLD_UNREACHABLE$, the disruption is increased.

Once the Public Homenet Zone file has been updated on the Hidden Primary, the Hidden Primary needs to inform the DOI that the Public Homenet Zone has been updated and that the IP address to use to retrieve the updated zone has also been updated. Both notifications

are performed using regular DNS exchanges. Mechanisms to update an IP address provided by lower layers with protocols like SCTP [RFC4960], MOBIKE [RFC4555] are not considered in this document. Instead the IP address of the HNA is updated using the Synchronization Channel as described in Section 4.3.

12. Privacy Considerations

Outsourcing the DNS Authoritative service from the HNA to a third party raises a few privacy related concerns.

The Public Homenet Zone lists the names of services hosted in the home network. Combined with blocking of AXFR queries, the use of NSEC3 [RFC5155] (vs NSEC [RFC4034]) prevents an attacker from being able to walk the zone, to discover all the names. However, the attacker may be able to walk the reverse DNS zone, or use other reconnaissance techniques to learn this information as described in [RFC7707].

In general a home network owner is expected to publish only names for which there is some need to be able to reference externally. Publication of the name does not imply that the service is necessarily reachable from any or all parts of the Internet. [RFC7084] mandates that the outgoing-only policy [RFC6092] be available, and in many cases it is configured by default. A well designed User Interface would combine a policy for making a service public by a name with a policy on who may access it.

In many cases, the home network owner wishes to publish names for services that only they will be able to access. The access control may consist of an IP source address range, or access may be restricted via some VPN functionality. The purpose of publishing the name is so that the service may be access by the same name both within the home, and outside the home. Sending traffic to the relevant IPv6 address causes the relevant VPN policy to be enacted upon.

While the problem of getting access to internal names has been solved in Enterprise configurations with a split-DNS, and such a thing could be done in the home, many recent improvements to VPN user interfaces make it more likely that an individual might have multiple connections configured. For instance, an adult child checking on the state of a home automation system for a parent.

In addition to the Public Homenet Zone, pervasive DNS monitoring can also monitor the traffic associated with the Public Homenet Zone. This traffic may provide an indication of the services an end user accesses, plus how and when they use these services. Although,

caching may obfuscate this information inside the home network, it is likely that outside your home network this information will not be cached.

13. Security Considerations

This document exposes a mechanism that prevents the HNA from being exposed to the Internet and served DNS request from the Internet. These requests are instead served by the DOI. While this limits the level of exposure of the HNA, the HNA remains exposed to the Internet with communications with the DOI. This section analyses the attack surface associated to these communications. In addition, the DOI exposes data that are related to the home network. This section also analyses the implication of such exposure.

13.1. HNA DM channels

The channels between HNA and DM are mutually authenticated and encrypted with TLS [RFC8446] and its associated security considerations apply. To ensure the multiple TLS session are continuously authenticating the same entity, TLS may take advantage of second factor authentication as described in [RFC8672].

At the time of writing TLS 1.2 or TLS 1.3 can be used and TLS 1.3 (or newer) SHOULD be supported.

The DNS protocol is subject to reflection attacks, however, these attacks are largely applicable when DNS is carried over UDP. The interfaces between the HNA and DM are using TLS over TCP, which prevents such reflection attacks. Note that Public Authoritative servers hosted by the DOI are subject to such attacks, but that is out of scope of our document.

Note that in the case of the Reverse Homenet Zone, the data is less subject to attacks than in the Public Homenet Zone. In addition, the DM and RDM may be provided by the ISP - as described in [I-D.ietf-homenet-naming-architecture-dhc-options], in which case DM and RDM might be less exposed to attacks - as communications within a network.

13.2. Names are less secure than IP addresses

This document describes how an end user can make their services and devices from his home network reachable on the Internet by using names rather than IP addresses. This exposes the home network to attackers, since names are expected to include less entropy than IP addresses. In fact, with IP addresses, the Interface Identifier is 64 bits long leading to up to 2^{64} possibilities for a given

subnetwork. This is not to mention that the subnet prefix is also of 64 bits long, thus providing up to 2^{64} possibilities. On the other hand, names used either for the home network domain or for the devices present less entropy (livebox, router, printer, nicolas, jennifer, ...) and thus potentially exposes the devices to dictionary attacks.

13.3. Names are less volatile than IP addresses

IP addresses may be used to locate a device, a host or a service. However, home networks are not expected to be assigned a time invariant prefix by ISPs. As a result, observing IP addresses only provides some ephemeral information about who is accessing the service. On the other hand, names are not expected to be as volatile as IP addresses. As a result, logging names over time may be more valuable than logging IP addresses, especially to profile an end user's characteristics.

PTR provides a way to bind an IP address to a name. In that sense, responding to PTR DNS queries may affect the end user's privacy. For that reason end users may choose not to respond to PTR DNS queries and MAY instead return a NXDOMAIN response.

14. Information Model for Outsourced information

This section is non-normative for the front-end protocol. It specifies an optional format for the set of parameters required by the HNA to configure the naming architecture of this document.

In cases where a home router has not been provisioned by the manufacturer (when forward zones are provided by the manufacturer), or by the ISP (when the ISP provides this service), then a home user/owner will need to configure these settings via an administrative interface.

By defining a standard format (in JSON) for this configuration information, the user/owner may be able to just copy and paste a configuration blob from the service provider into the administrative interface of the HNA.

This format may also provide the basis for a future OAUTH2 [RFC6749] flow that could do the setup automatically.

The HNA needs to be configured with the following parameters as described by this CDDL [RFC8610]. These are the parameters necessary to establish a secure channel between the HNA and the DM as well as to specify the DNS zone that is in the scope of the communication.

```

hna-configuration = {
  "registered_domain" : tstr,
  "dm"                 : tstr,
  ? "dm_transport"    : "DoT"
  ? "dm_port"         : uint,
  ? "dm_acl"          : hna-acl / [ +hna-acl ]
  ? "hna_auth_method" : hna-auth-method
  ? "hna_certificate" : tstr
}

```

```

hna-acl           = tstr
hna-auth-method   /= "certificate"

```

For example:

```

{
  "registered_domain" : "n8d234f.r.example.net",
  "dm"                 : "2001:db8:1234:111:222::2",
  "dm_transport"      : "DoT",
  "dm_port"           : 4433,
  "dm_acl"             : "2001:db8:1f15:62e:21c::/64"
                        or [ "2001:db8:1f15:62e:21c::/64", ... ]
  "hna_auth_method"   : "certificate",
  "hna_certificate"   : "-----BEGIN CERTIFICATE-----\nMIIDTjCCFGy....",
}

```

14.1. Outsourced Information Model

Registered Homenet Domain (zone) The Domain Name of the zone.
Multiple Registered Homenet Domains may be provided. This will generate the creation of multiple Public Homenet Zones. This parameter is MANDATORY.

Distribution Manager notification address (dm) The associated FQDNs or IP addresses of the DM to which DNS notifies should be sent. This parameter is MANDATORY. IP addresses are optional and the FQDN is sufficient and preferred. If there are concerns about the security of the name to IP translation, then DNSSEC should be employed.

As the session between the HNA and the DM is authenticated with TLS, the use of names is easier.

As certificates are more commonly emitted for FQDN than for IP addresses, it is preferred to use names and authenticate the name of the DM during the TLS session establishment.

Supported Transport (dm_transport) The transport that carries the DNS exchanges between the HNA and the DM. Typical value is "DoT" but it may be extended in the future with "DoH", "DoQ" for example. This parameter is OPTIONAL and by default the HNA uses DoT.

Distribution Manager Port (dm_port) Indicates the port used by the DM. This parameter is OPTIONAL and the default value is provided by the Supported Transport. In the future, additional transport may not have default port, in which case either a default port needs to be defined or this parameter become MANDATORY.

Note that HNA does not defines ports for the Synchronization Channel. In any case, this is not expected to part of the configuration, but instead negotiated through the Configuration Channel. Currently the Configuration Channel does not provide this, and limits its agility to a dedicated IP address. If such agility is needed in the future, additional exchanges will need to be defined.

Authentication Method ("hna_auth_method"): How the HNA authenticates itself to the DM within the TLS connection(s). The authentication meth of can typically be "certificate", "psk" or "none". This Parameter is OPTIONAL and by default the Authentication Method is "certificate".

Authentication data ("hna_certificate", "hna_key"): : The certificate chain used to authenticate the HNA. This parameter is OPTIONAL and when not specified, a self-signed certificate is used.

Distribution Manager AXFR permission netmask (dm_acl): The subnet from which the CPE should accept SOA queries and AXFR requests. A subnet is used in the case where the DNS Outsourced Infrastructure consists of a number of different systems. An array of addresses is permitted. This parameter is OPTIONAL and if unspecified, the CPE the IP addresses specified in the dm_notify parameters or the IP addresses that result from the DNS(SEC) resolution when dm_notify specifies a FQDN.

For forward zones, the relationship between the HNA and the forward zone provider may be the result of a number of transactions:

1. The forward zone outsourcing may be provided by the maker of the Homenet router. In this case, the identity and authorization could be built in the device at manufacturer provisioning time. The device would need to be provisioned with a device-unique credential, and it is likely that the Registered Homenet Domain would be derived from a public attribute of the device, such as a

serial number (see Appendix B or [I-D.richardson-homerouter-provisioning] for more details).

2. The forward zone outsourcing may be provided by the Internet Service Provider. In this case, the use of [I-D.ietf-homenet-naming-architecture-dhc-options] to provide the credentials is appropriate.
3. The forward zone may be outsourced to a third party, such as a domain registrar. In this case, the use of the JSON-serialized YANG data model described in this section is appropriate, as it can easily be copy and pasted by the user, or downloaded as part of a web transaction.

For reverse zones, the relationship is always with the upstream ISP (although there may be more than one), and so [I-D.ietf-homenet-naming-architecture-dhc-options] is always the appropriate interface.

The following is an abridged example of a set of data that represents the needed configuration parameters for outsourcing.

15. IANA Considerations

This document has no actions for IANA.

16. Acknowledgment

The authors wish to thank Philippe Lemordant for its contributions on the early versions of the draft; Ole Troan for pointing out issues with the IPv6 routed home concept and placing the scope of this document in a wider picture; Mark Townsley for encouragement and injecting a healthy debate on the merits of the idea; Ulrik de Bie for providing alternative solutions; Paul Mockapetris, Christian Jacquenet, Francis Dupont and Ludovic Eschard for their remarks on HNA and low power devices; Olafur Gudmundsson for clarifying DNSSEC capabilities of small devices; Simon Kelley for its feedback as dnsmasq implementer; Andrew Sullivan, Mark Andrew, Ted Lemon, Mikael Abrahamson, and Ray Bellis for their feedback on handling different views as well as clarifying the impact of outsourcing the zone signing operation outside the HNA; Mark Andrew and Peter Koch for clarifying the renumbering.

17. Contributors

The co-authors would like to thank Chris Griffiths and Wouter Cloetens that provided a significant contribution in the early versions of the document.

18. References

18.1. Normative References

- [I-D.ietf-dprive-xfr-over-tls]
Toorop, W., Dickinson, S., Sahib, S., Aras, P., and A. Mankin, "DNS Zone Transfer-over-TLS", draft-ietf-dprive-xfr-over-tls-11 (work in progress), April 2021.
- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [RFC1995] Ohta, M., "Incremental Zone Transfer in DNS", RFC 1995, DOI 10.17487/RFC1995, August 1996, <<https://www.rfc-editor.org/info/rfc1995>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, DOI 10.17487/RFC2136, April 1997, <<https://www.rfc-editor.org/info/rfc2136>>.
- [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <<https://www.rfc-editor.org/info/rfc4034>>.
- [RFC4192] Baker, F., Lear, E., and R. Droms, "Procedures for Renumbering an IPv6 Network without a Flag Day", RFC 4192, DOI 10.17487/RFC4192, September 2005, <<https://www.rfc-editor.org/info/rfc4192>>.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <<https://www.rfc-editor.org/info/rfc4301>>.
- [RFC4555] Eronen, P., "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", RFC 4555, DOI 10.17487/RFC4555, June 2006, <<https://www.rfc-editor.org/info/rfc4555>>.
- [RFC4960] Stewart, R., Ed., "Stream Control Transmission Protocol", RFC 4960, DOI 10.17487/RFC4960, September 2007, <<https://www.rfc-editor.org/info/rfc4960>>.

- [RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS Security (DNSSEC) Hashed Authenticated Denial of Existence", RFC 5155, DOI 10.17487/RFC5155, March 2008, <<https://www.rfc-editor.org/info/rfc5155>>.
- [RFC5936] Lewis, E. and A. Hoenes, Ed., "DNS Zone Transfer Protocol (AXFR)", RFC 5936, DOI 10.17487/RFC5936, June 2010, <<https://www.rfc-editor.org/info/rfc5936>>.
- [RFC6092] Woodyatt, J., Ed., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, DOI 10.17487/RFC6092, January 2011, <<https://www.rfc-editor.org/info/rfc6092>>.
- [RFC6644] Evans, D., Droms, R., and S. Jiang, "Rebind Capability in DHCPv6 Reconfigure Messages", RFC 6644, DOI 10.17487/RFC6644, July 2012, <<https://www.rfc-editor.org/info/rfc6644>>.
- [RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <<https://www.rfc-editor.org/info/rfc6698>>.
- [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>>.
- [RFC6887] Wing, D., Ed., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, DOI 10.17487/RFC6887, April 2013, <<https://www.rfc-editor.org/info/rfc6887>>.
- [RFC7010] Liu, B., Jiang, S., Carpenter, B., Venaas, S., and W. George, "IPv6 Site Renumbering Gap Analysis", RFC 7010, DOI 10.17487/RFC7010, September 2013, <<https://www.rfc-editor.org/info/rfc7010>>.
- [RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, DOI 10.17487/RFC7084, November 2013, <<https://www.rfc-editor.org/info/rfc7084>>.

- [RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <<https://www.rfc-editor.org/info/rfc7296>>.
- [RFC7344] Kumari, W., Gudmundsson, O., and G. Barwood, "Automating DNSSEC Delegation Trust Maintenance", RFC 7344, DOI 10.17487/RFC7344, September 2014, <<https://www.rfc-editor.org/info/rfc7344>>.
- [RFC7368] Chown, T., Ed., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", RFC 7368, DOI 10.17487/RFC7368, October 2014, <<https://www.rfc-editor.org/info/rfc7368>>.
- [RFC7558] Lynn, K., Cheshire, S., Blanchet, M., and D. Migault, "Requirements for Scalable DNS-Based Service Discovery (DNS-SD) / Multicast DNS (mDNS) Extensions", RFC 7558, DOI 10.17487/RFC7558, July 2015, <<https://www.rfc-editor.org/info/rfc7558>>.
- [RFC7707] Gont, F. and T. Chown, "Network Reconnaissance in IPv6 Networks", RFC 7707, DOI 10.17487/RFC7707, March 2016, <<https://www.rfc-editor.org/info/rfc7707>>.
- [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>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8375] Pfister, P. and T. Lemon, "Special-Use Domain 'home.arpa.'", RFC 8375, DOI 10.17487/RFC8375, May 2018, <<https://www.rfc-editor.org/info/rfc8375>>.
- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.

- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [RFC8555] Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", RFC 8555, DOI 10.17487/RFC8555, March 2019, <<https://www.rfc-editor.org/info/rfc8555>>.

18.2. Informative References

- [I-D.ietf-dprive-dnsoquic]
Huitema, C., Mankin, A., and S. Dickinson, "Specification of DNS over Dedicated QUIC Connections", draft-ietf-dprive-dnsoquic-02 (work in progress), February 2021.
- [I-D.ietf-homenet-naming-architecture-dhc-options]
Migault, D., Weber, R., and T. Mrugalski, "DHCPv6 Options for Home Network Naming Authority", draft-ietf-homenet-naming-architecture-dhc-options-12 (work in progress), April 2021.
- [I-D.ietf-homenet-simple-naming]
Lemon, T., Migault, D., and S. Cheshire, "Homenet Naming and Service Discovery Architecture", draft-ietf-homenet-simple-naming-03 (work in progress), October 2018.
- [I-D.richardson-homerouter-provisioning]
Richardson, M., "Provisioning Initial Device Identifiers into Home Routers", draft-richardson-homerouter-provisioning-00 (work in progress), November 2020.
- [RFC2845] Vixie, P., Gudmundsson, O., Eastlake 3rd, D., and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", RFC 2845, DOI 10.17487/RFC2845, May 2000, <<https://www.rfc-editor.org/info/rfc2845>>.
- [RFC2931] Eastlake 3rd, D., "DNS Request and Transaction Signatures (SIG(0)s)", RFC 2931, DOI 10.17487/RFC2931, September 2000, <<https://www.rfc-editor.org/info/rfc2931>>.
- [RFC6749] Hardt, D., Ed., "The OAuth 2.0 Authorization Framework", RFC 6749, DOI 10.17487/RFC6749, October 2012, <<https://www.rfc-editor.org/info/rfc6749>>.

- [RFC8094] Reddy, T., Wing, D., and P. Patil, "DNS over Datagram Transport Layer Security (DTLS)", RFC 8094, DOI 10.17487/RFC8094, February 2017, <<https://www.rfc-editor.org/info/rfc8094>>.
- [RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <<https://www.rfc-editor.org/info/rfc8484>>.
- [RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <<https://www.rfc-editor.org/info/rfc8499>>.
- [RFC8501] Howard, L., "Reverse DNS in IPv6 for Internet Service Providers", RFC 8501, DOI 10.17487/RFC8501, November 2018, <<https://www.rfc-editor.org/info/rfc8501>>.
- [RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <<https://www.rfc-editor.org/info/rfc8610>>.
- [RFC8672] Sheffer, Y. and D. Migault, "TLS Server Identity Pinning with Tickets", RFC 8672, DOI 10.17487/RFC8672, October 2019, <<https://www.rfc-editor.org/info/rfc8672>>.

Appendix A. Envisioned deployment scenarios

A number of deployment have been envisioned, this section aims at providing a brief description. The use cases are not limitations and this section is not normative.

A.1. CPE Vendor

A specific vendor with specific relations with a registrar or a registry may sell a CPE that is provisioned with provisioned domain name. Such domain name does not need to be necessary human readable.

One possible way is that the vendor also provisions the HNA with a private and public keys as well as a certificate. Note that these keys are not expected to be used for DNSSEC signing. Instead these keys are solely used by the HNA to proceed to the authentication. Normally the keys should be necessary and sufficient to proceed to the authentication. The reason to combine the domain name and the key is that DOI are likely handle names better than keys and that domain names might be used as a login which enables the key to be regenerated.

When the home network owner plugs the CPE at home, the relation between HNA and DM is expected to work out-of-the-box.

A.2. Agnostic CPE

An CPE that is not preconfigured may also take advantage to the protocol defined in this document but some configuration steps will be needed.

1. The owner of the home network buys a domain name to a registrar, and as such creates an account on that registrar
2. Either the registrar is also providing the outsourcing infrastructure or the home network needs to create a specific account on the outsourcing infrastructure. * If the DOI is the registrar, it has by design a proof of ownership of the domain name by the homenet owner. In this case, it is expected the DOI provides the necessary parameters to the home network owner to configure the HNA. A good way to provide the parameters would be the home network be able to copy/paste a JSON object - see Section 14. What matters at that point is the DOI being able to generate authentication credentials for the HNA to authenticate itself to the DOI. This obviously requires the home network to provide the public key generated by the HNA in a CSR.

- o If the DOI is not the registrar, then the proof of ownership needs to be established using protocols like ACME [RFC8555] for example that will end in the generation of a certificate. ACME is used here to the purpose of automating the generation of the certificate, the CA may be a specific CA or the DOI. With that being done, the DOI has a roof of ownership and can proceed as above.

Appendix B. Example: A manufacturer provisioned HNA product flow

This scenario is one where a homenet router device manufacturer decides to offer DNS hosting as a value add.

[I-D.richardson-homerouter-provisioning] describes a process for a home router credential provisioning system. The outline of it is that near the end of the manufacturing process, as part of the firmware loading, the manufacturer provisions a private key and certificate into the device.

In addition to having a assymmetric credential known to the manufacturer, the device also has been provisioned with an agreed upon name. In the example in the above document, the name "n8d234f.r.example.net" has already been allocated and confirmed with the manufacturer.

The HNA can use the above domain for itself. It is not very pretty or personal, but if the owner wishes a better name, they can arrange for it.

The configuration would look like:

```
{
  "dm_notify" : "2001:db8:1234:111:222::2",
  "dm_acl"    : "2001:db8:1234:111:222::/64",
  "dm_ctrl"   : "manufacturer.example.net",
  "dm_port"   : "4433",
  "ns_list"   : [ "ns1.publicdns.example", "ns2.publicdns.example"],
  "zone"      : "n8d234f.r.example.net",
  "auth_method" : "certificate",
  "hna_certificate": "-----BEGIN CERTIFICATE-----\nMIIDTjCCFGy....",
}
```

The dm_ctrl and dm_port values would be built into the firmware.

Authors' Addresses

Daniel Migault
Ericsson
8275 Trans Canada Route
Saint Laurent, QC 4S 0B6
Canada

EMail: daniel.migault@ericsson.com

Ralf Weber
Nominum
2000 Seaport Blvd
Redwood City 94063
US

EMail: ralf.weber@nominum.com

Michael Richardson
Sandelman Software Works
470 Dawson Avenue
Ottawa, ON K1Z 5V7
Canada

EMail: mcr+iETF@sandelman.ca

Ray Hunter
Globis Consulting BV
Weegschaalstraat 3
Eindhoven 5632CW
NL

EMail: v6ops@globis.net

Homenet
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2021

D. Migault
Ericsson
R. Weber
Akamai
T. Mrugalski
Internet Systems Consortium, Inc.
C. Griffiths

W. Cloetens
Deutsche Telekom
October 22, 2020

DHCPv6 Options for Home Network Naming Authority
draft-ietf-homenet-naming-architecture-dhc-options-08

Abstract

This document defines DHCPv6 options so any agnostic Homnet Naming Authority (HNA) can automatically proceed to the appropriate configuration and outsource the authoritative naming service for the home network. In most cases, the outsourcing mechanism is transparent for the end user.

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 <https://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 April 25, 2021.

Copyright Notice

Copyright (c) 2020 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

(<https://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. Terminology	2
2. Introduction	3
3. Protocol Overview	4
4. Payload Description	5
4.1. Client Public Key Option	5
4.2. Registered Homenet Domain Option	5
4.3. Distribution Master Option	6
4.3.1. Supported Transport	6
4.4. Reverse Distribution Master Server Option	7
5. DHCP Behavior	8
5.1. DHCPv6 Server Behavior	8
5.2. DHCPv6 Client Behavior	8
5.3. DHCPv6 Relay Agent Behavior	8
6. IANA Considerations	8
7. Security Considerations"	9
8. Acknowledgments	9
9. References	9
9.1. Normative References	9
9.2. Informative References	10
Appendix A. Scenarios and impact on the End User	11
Appendix B. Base Scenario	11
B.1. Third Party Registered Homenet Domain	11
B.2. Third Party DNS Infrastructure	12
B.3. Multiple ISPs	12
Authors' Addresses	13

1. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The reader is expected to be familiar with [I-D.ietf-homenet-front-end-naming-delegation] and its terminology section. This section defines terms that have not been defined in [I-D.ietf-homenet-front-end-naming-delegation]:

- o Client Public Key: designates a public key generated by the HNA and used as an authentication credential for the HNA.

2. Introduction

[I-D.ietf-homenet-front-end-naming-delegation] describes how Homenet Naming Authority (HNA) outsources the Public Homenet Zone to an Outsourcing Infrastructure.

This document shows how an ISP can provision automatically the HNA with an DNS Outsourcing Infrastructure (DOI). Most likely the DOI will be - at least partly be - managed or provided by its ISP, but other cases may envision the ISP storing some configuration so the homenet becomes resilient to HNA replacement.

The ISP delegates the home network an IP prefix it owns as well as the associated reverse zone.

The ISP is well aware of the owner of that prefix, and as such becomes a natural candidate for hosting the Homenet Reverse Zone - that is the Reverse Distribution Master (RDM) and potentially the Reverse Public Authoritative Servers.

In addition, the ISP often identifies the home network with a name. In most cases, the name is used by the ISP for its internal network management operations and is not a name the home network owner has registered to. The ISP may thus leverage such infrastructure and provide the homenet a specific domain name designated as per [I-D.ietf-homenet-front-end-naming-delegation] a Homenet Registered Domain. Similarly to the reverse zone, the ISP is well aware of who owns that domain name and may become a natural candidate for hosting the Homenet Zone - that is the Distribution Master (DM) and the Public Authoritative Servers.

This document describes DHCPv6 options so the HNA can advertise the ISP its identity via a Client Public Key. The ISP internally associates the Registered Homenet Domains with that key - that is to the DM and RDM. The ISP provides the Registered Homenet Domain, necessary information on the DM and the RDM so the HNA can manage and upload the Public Homenet Zone and the Reverse Public Homenet Zone as described in [I-D.ietf-homenet-front-end-naming-delegation].

The use of DHCPv6 options makes the configuration completely transparent to the end user and provides a similar level of trust as the one used to provide the IP prefix. The link between the HNA and the DHCPv6 server may benefit from additional security for example by using [I-D.ietf-dhc-sedhcpv6].

3. Protocol Overview

This section illustrates how a HNA receives the necessary information via DHCPv6 options to outsource its authoritative naming service to the DOI. For the sake of simplicity, and similarly to [I-D.ietf-homenet-front-end-naming-delegation], this section assumes that the HNA and the home network DHCPv6 client are collocated on the CPE. Note also that this is not mandatory and specific communications between the HNA and the DHCPv6 client only are needed. In addition, this section assumes the responsible entity for the DHCPv6 server is able to configure the DM and RDM. In our case, this means a Registered Homenet Domain can be associated to a Client Public Key.

This scenario has been chosen as it is believed to be the most popular scenario. This document does not ignore scenarios where the DHCP Server does not have privileged relations with the DM or RDM. These cases are discussed latter in Appendix A. Such scenarios do not necessarily require configuration for the end user and can also be zero-config.

The scenario considered in this section is as follows:

1. The HNA is willing to outsource the Public Homenet Zone or Homenet Reverse Zone and configures its DHCP Client to provide its Client Public Key to the DHCP Server using a Client Public Key Option (OPTION_PUBLIC_KEY).
In addition, it also requests the DHCP Client to include in its Option Request Option (ORO) the Registered Homnet Domain Option (OPTION_REGISTERED_DOMAIN), the Distribution Master Option (OPTION_DIST_MASTER) and the Reverse Distribution Master Option (OPTION_REVERSE_DIST_MASTER) option codes.
2. The DHCP Server updates as indicated by the ORO, the DM and RDM, and associate the Registered Domain with the Client Public Key.
3. The DHCP Server responds to the HNA with the requested DHCPv6 options, i.e. OPTION_DIST_MASTER) and OPTION_REVERSE_DIST_MASTER. The DHCP Client transmits the information to the HNA.
4. The HNA is able to get authenticated by the DM and the RDM. The HNA builds the Homenet Zone (resp. the Homenet Reverse Zone) and proceed as described in [I-D.ietf-homenet-front-end-naming-delegation].

4. Payload Description

This section details the payload of the DHCPv6 options.

4.1. Client Public Key Option

The Client Public Key Option (OPTION_PUBLIC_KEY) indicates the Client Public Key that is used to authenticate the HNA. This option is also defined in [I-D.ietf-dhc-sedhcpv6].

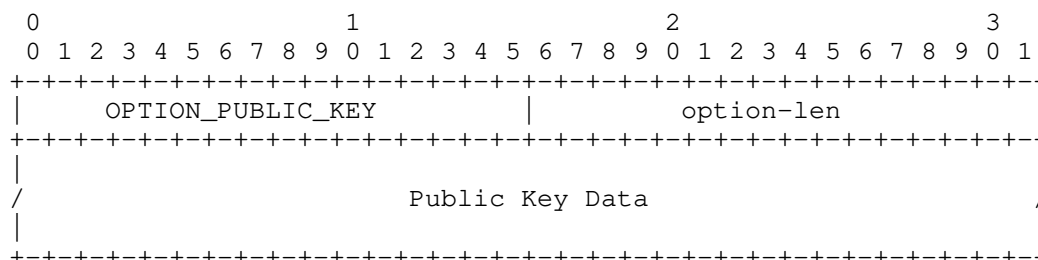


Figure 1: Client Public Key Option

- o option-code (16 bits): `OPTION_PUBLIC_KEY`, the option code for the Client Public Key Option (TBD1).
- o option-len (16 bits): length in octets of the option-data field as described in [RFC3315].
- o Client Public Key Data: contains the Client Public Key. The format is the DNSKEY RDATA format as defined in [RFC4034].

4.2. Registered Homenet Domain Option

The Registered Domain Option (OPTION_REGISTERED_DOMAIN) indicates the FQDN associated to the home network.

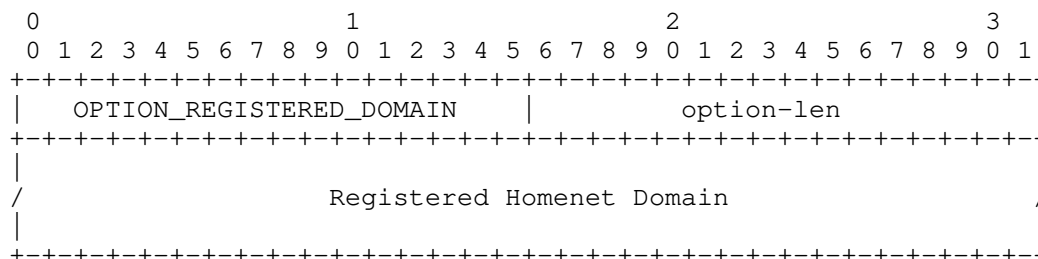


Figure 2: Client Public Key Option

- o option-code (16 bits): OPTION_REGISTERED_DOMAIN, the option code for the Registered Homenet Domain (TBD2).
- o option-len (16 bits): length in octets of the option-data field as described in [RFC3315].
- o Registered Homenet Domain (variable): the FQDN registered for the homenet

4.3. Distribution Master Option

The Distributed Master Option (OPTION_DIST_MASTER) provides the HNA to FQDN of the DM as well as the transport protocol for the transaction between the HNA and the DM.

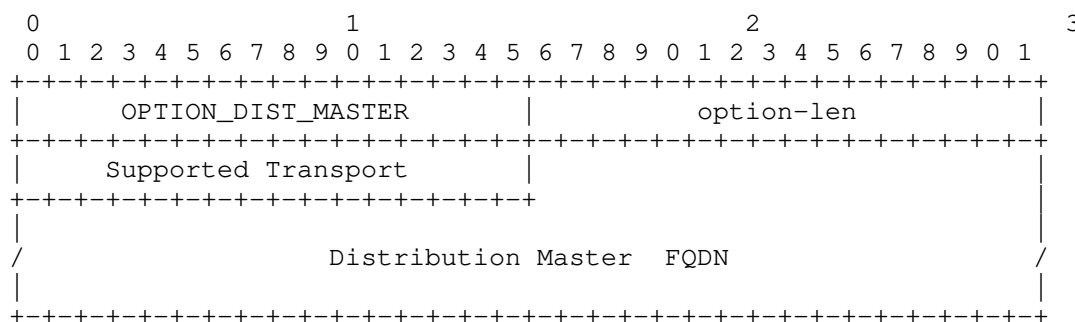


Figure 3: Distribution Master Option

- o option-code (16 bits): OPTION_DIST_MASTER, the option code for the DM Option (TBD3).
- o option-len (16 bits): length in octets of the option-data field as described in [RFC3315].
- o Supported Transport (16 bits): defines the supported transport by the DM. Each bit represents a supported transport, and a DM MAY indicate the support of multiple modes. The bit for DoT MUST be set.
- o Distribution Master FQDN (variable): the FQDN of the DM.

4.3.1. Supported Transport

The Supported Transport field of the DHCPv6 option indicates the supported transport protocol. Each bit represents a specific transport mechanism. The bit sets to 1 indicates the associated

transport protocol is supported. The corresponding bits are assigned as described in Figure 4.

Bit	Transport Protocol	Reference
0	DNS over TLS	
1	DNS over HTTPS	
2-7	unallocated	

Figure 4: Supported Protocols

4.4. Reverse Distribution Master Server Option

The Reverse Distribution Master Server Option (OPTION_REVERSE_DIST_MASTER) provides the HNA to FQDN of the DM as well as the transport protocol for the transaction between the HNA and the DM.

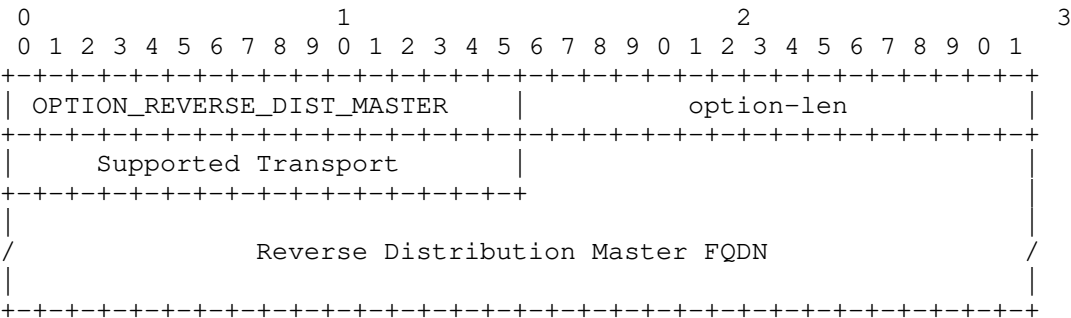


Figure 5: Reverse Distribution Master Option

- o option-code (16 bits): OPTION_REVERSE_DIST_MASTER, the option code for the Reverse Distribution Master Option (TBD4).
- o option-len (16 bits): length in octets of the option-data field as described in [RFC3315].
- o Supported Transport (16 bits): defines the supported transport by the DM. Each bit represents a supported transport, and a DM MAY indicate the support of multiple modes. The bit for DoT MUST be set.
- o Reverse Distribution Master FQDN (variable): the FQDN of the RDM.

5. DHCP Behavior

5.1. DHCPv6 Server Behavior

Sections 17.2.2 and 18.2 of [RFC3315] govern server operation in regards to option assignment. As a convenience to the reader, we mention here that the server will send option foo only if configured with specific values for foo and if the client requested it. In particular, when configured the DHCP Server sends the Registered Homenet Domain Option, Distribution Master Option, the Reverse Distribution Master Option when requested by the DHCPv6 client by including necessary option codes in its ORO.

The DHCP Server may receive a Client Public Key Option (OPTION_PUBLIC_KEY) from the HNA. Upon receipt of this DHCPv6 option, the DHCP Server SHOULD acknowledge the reception of the Client Public Key Option as described and communicate this credential to the available DM and RDM unless not configured to do so.

A HNA may update its Client Public Key by sending a new value in the Client Public Key Option (OPTION_PUBLIC_KEY) as this document assumes the link between the HNA and the DHCP Server is considered authenticated and trusted. The server SHOULD process received Client Public Key Option sent by the client unless not configured to do so.

5.2. DHCPv6 Client Behavior

The DHCPv6 client SHOULD send a Client Public Key Option (OPTION_PUBLIC_KEY) to the DHCP Server. This Client Public Key authenticates the HNA.

The DHCPv6 client sends a ORO with the necessary option codes: Registered Homenet Domain Option, Distribution Master Option, the Reverse Distribution Master Option.

Upon receiving a DHCP option described in this document in the Reply message, the HNA SHOULD proceed as described in [I-D.ietf-homenet-front-end-naming-delegation].

5.3. DHCPv6 Relay Agent Behavior

There are no additional requirements for the DHCP Relay agents.

6. IANA Considerations

The DHCP options detailed in this document are: * OPTION_CLIENT_KEY: TBD1 * OPTION_REGISTERED_DOMAIN: TBD2 * OPTION_DIST_MASTER: TBD3 * OPTION_REVERSE_DIST_MASTER: TBD4

The document also requests a Supported Transport Registry:

Bit	Transport Protocol	Reference
0	DNS over TLS	
1	DNS over HTTPS	
2-7	unallocated	

7. Security Considerations"

8. Acknowledgments

We would like to thank Marcin Siodelski and Bernie Volz for their comments on the design of the DHCPv6 options. We would also like to thank Mark Andrews, Andrew Sullivan and Lorenzo Colliti for their remarks on the architecture design. The designed solution has been largely been inspired by Mark Andrews's document [I-D.andrews-dnsop-pd-reverse] as well as discussions with Mark. We also thank Ray Hunter for its reviews, its comments and for suggesting an appropriated terminology.

9. References

9.1. Normative References

- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, DOI 10.17487/RFC2181, July 1997, <<https://www.rfc-editor.org/info/rfc2181>>.
- [RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <<https://www.rfc-editor.org/info/rfc3315>>.
- [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <<https://www.rfc-editor.org/info/rfc4034>>.

- [RFC6672] Rose, S. and W. Wijngaards, "DNAME Redirection in the DNS", RFC 6672, DOI 10.17487/RFC6672, June 2012, <<https://www.rfc-editor.org/info/rfc6672>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

9.2. Informative References

- [I-D.andrews-dnsop-pd-reverse]
Andrews, M., "Automated Delegation of IP6.ARPA reverse zones with Prefix Delegation", draft-andrews-dnsop-pd-reverse-02 (work in progress), November 2013.
- [I-D.ietf-dhc-sedhcpv6]
Li, L., Jiang, S., Cui, Y., Jinmei, T., Lemon, T., and D. Zhang, "Secure DHCPv6", draft-ietf-dhc-sedhcpv6-21 (work in progress), February 2017.
- [I-D.ietf-homenet-front-end-naming-delegation]
Migault, D., Weber, R., Richardson, M., Hunter, R., Griffiths, C., and W. Cloetens, "Outsourcing Home Network Authoritative Naming Service", draft-ietf-homenet-front-end-naming-delegation-11 (work in progress), April 2020.
- [I-D.sury-dnsexst-cname-dname]
Sury, O., "CNAME+DNAME Name Redirection", draft-sury-dnsexst-cname-dname-00 (work in progress), April 2010.

Appendix A. Scenarios and impact on the End User

This section details various scenarios and discuss their impact on the end user. This section is not normative and limits the description of a limited scope of scenarios that are assumed to be representative. Many other scenarios may be derived from these.

Appendix B. Base Scenario

The base scenario is the one described in Section 3 in which an ISP manages the DHCP Server, the DM and RDM.

The end user subscribes to the ISP (foo), and at subscription time registers for example.foo as its Registered Homenet Domain example.foo.

When the HNA is plugged (at least the first time), it provides its Client Public Key to the DHCP Server. In this scenario, the DHCP Server, DM and RDM are managed by the ISP so the DHCP Server and as such can provide authentication credentials of the HNA to enable secure authenticated transaction with the DM and the Reverse DM.

The main advantage of this scenario is that the naming architecture is configured automatically and transparently for the end user. The drawbacks are that the end user uses a Registered Homenet Domain managed by the ISP and that it relies on the ISP naming infrastructure.

B.1. Third Party Registered Homenet Domain

This section considers the case when the end user wants its home network to use example.com not managed by her ISP (foo) as a Registered Homenet Domain.

This section still consider the ISP manages the home network and still provides example.foo as a Registered Homenet Domain.

When the end user buys the domain name example.com, it may request to redirect the name example.com to example.foo using static redirection with CNAME [RFC2181], [RFC1034], DNAME [RFC6672] or CNAME+DNAME [I-D.sury-dnsextn-cname-dname].

This configuration is performed once when the domain name example.com is registered. The only information the end user needs to know is the domain name assigned by the ISP. Once this configuration is done no additional configuration is needed anymore. More specifically, the HNA may be changed, the zone can be updated as in Appendix B without any additional configuration from the end user.

The main advantage of this scenario is that the end user benefits from the Zero Configuration of the Base Scenario Appendix B. Then, the end user is able to register for its home network an unlimited number of domain names provided by an unlimited number of different third party providers.

The drawback of this scenario may be that the end user still rely on the ISP naming infrastructure. Note that the only case this may be inconvenient is when the DNS Servers provided by the ISPs results in high latency.

B.2. Third Party DNS Infrastructure

This scenario considers that the end user uses example.com as a Registered Homenet Domain, and does not want to rely on the authoritative servers provided by the ISP.

In this section we limit the outsourcing to the DM and Public Authoritative Server(s) to a third party. The Reverse Public Authoritative Server(s) and the RDM remain managed by the ISP as the IP prefix is managed by the ISP.

Outsourcing to a third party DM can be performed in the following ways:

1. Updating the DHCP Server Information. One can imagine a GUI interface that enables the end user to modify its profile parameters. Again, this configuration update is done once-for-ever.
2. Upload the configuration of the DM to the HNA. In some cases, the provider of the CPE hosting the HNA may be the registrar and provide the CPE already configured. In other cases, the CPE may request the end user to log into the registrar to validate the ownership of the Registered Homenet Domain and agree on the necessary credentials to secure the communication between the HNA and the DM. As described in [I-D.ietf-homenet-front-end-naming-delegation], such settings could be performed in an almost automatic way as to limit the necessary interactions with the end user.

B.3. Multiple ISPs

This scenario considers a HNA connected to multiple ISPs.

Suppose the HNA has been configured each of its interfaces independently with each ISPS as described in Appendix B. Each ISP provides a different Registered Homenet Domain. The HNA Client Public Key may be shared between the HNA and the multiple ISPs.

The protocol and DHCPv6 options described in this document are fully compatible with a HNA connected to multiple ISPs with multiple Registered Homenet Domains. However, the HNA should be able to handle different Registered Homenet Domains. This is an implementation issue which is outside the scope of the current document.

If a HNA is not able to handle multiple Registered Homenet Domains, the HNA may remain connected to multiple ISP with a single Registered Homenet Domain. In this case, one entity is chosen to host the Registered Homenet Domain. This entity may be one of the ISP or a third party. Note that having multiple ISPs can be motivated for bandwidth aggregation, or connectivity fail-over. In the case of connectivity fail-over, the fail-over concerns the access network and a failure of the access network may not impact the core network where the DM Server and Public Authoritative Primaries are hosted. In that sense, choosing one of the ISP even in a scenario of multiple ISPs may make sense. However, for sake of simplicity, this scenario assumes that a third party has been chosen to host the Registered Homenet Domain. Configuration is performed as described in Appendix B.1 and Appendix B.2.

With the configuration described in Appendix B.1, the HNA is expect to be able to handle multiple Homenet Registered Domain, as the third party redirect to one of the ISPs Servers. With the configuration described in Appendix B.2, DNS zone are hosted and maintained by the third party. A single DNS(SEC) Homenet Zone is built and maintained by the HNA. This latter configuration is likely to match most HNA implementations.

The protocol and DHCPv6 options described in this document are fully compatible with a HNA connected to multiple ISPs. To configure or not and how to configure the HNA depends on the HNA facilities. Appendix B and Appendix B.1 require the HNA to handle multiple Registered Homenet Domain, whereas Appendix B.2 does not have such requirement.

Authors' Addresses

Daniel Migault
Ericsson
8275 Trans Canada Route
Saint Laurent, QC 4S 0B6
Canada

EMail: daniel.migault@ericsson.com

Ralf Weber
Akamai

EMail: ralf.weber@akamai.com

Tomek Mrugalski
Internet Systems Consortium, Inc.
950 Charter Street
Redwood City 94063
US

EMail: tomasz.mrugalski@gmail.com

Chris Griffiths

EMail: cgriffiths@gmail.com

Wouter Cloetens
Deutsche Telekom

EMail: wouter.cloetens@external.telekom.de

Homenet
Internet-Draft
Intended status: Standards Track
Expires: November 14, 2021

D. Migault
Ericsson
R. Weber
Akamai
T. Mrugalski
Internet Systems Consortium, Inc.
May 13, 2021

DHCPv6 Options for Home Network Naming Authority
draft-ietf-homenet-naming-architecture-dhc-options-14

Abstract

This document defines DHCPv6 options so an agnostic Homenet Naming Authority (HNA) can automatically proceed to the appropriate configuration and outsource the authoritative naming service for the home network. In most cases, the outsourcing mechanism is transparent for the end user.

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 <https://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 November 14, 2021.

Copyright Notice

Copyright (c) 2021 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 (<https://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. Terminology	2
2. Introduction	2
3. Procedure Overview	3
4. DHCPv6 Option	4
4.1. Registered Homenet Domain Option	4
4.2. Distribution Manager Option	5
4.2.1. Supported Transport	6
4.3. Reverse Distribution Manager Server Option	6
5. DHCPv6 Behavior	7
5.1. DHCPv6 Server Behavior	7
5.2. DHCPv6 Client Behavior	7
5.3. DHCPv6 Relay Agent Behavior	7
6. IANA Considerations	7
7. Security Considerations	8
8. Acknowledgments	8
9. Contributors	8
10. References	8
10.1. Normative References	8
10.2. Informative References	9
Appendix A. Scenarios and impact on the End User	11
Appendix B. Base Scenario	11
B.1. Third Party Registered Homenet Domain	11
B.2. Third Party DNS Infrastructure	12
B.3. Multiple ISPs	12
Authors' Addresses	13

1. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The reader should be familiar with
[I-D.ietf-homenet-front-end-naming-delegation].

2. Introduction

[I-D.ietf-homenet-front-end-naming-delegation] specifies how an entity designated as the Homenet Naming Authority (HNA) outsources a Public Homenet Zone to an Outsourcing DNS Infrastructure (DOI).

This document describes how a network can provision the HNA with a specific DOI. This could be particularly useful for a DOI partly managed by an ISP, or to make home networks resilient to HNA replacement. The ISP delegates an IP prefix to the home network as well as the associated reverse zone. The ISP is thus aware of the owner of that IP prefix, and as such becomes a natural candidate for hosting the Homenet Reverse Zone - that is the Reverse Distribution Manager (RDM) and potentially the Reverse Public Authoritative Servers.

In addition, ISPs often identify the line of the home network with a name. Such name is used for their internal network management operations and is not a name the home network owner has registered to. ISPs may leverage such infrastructure and provide the homenet with a specific domain name designated as per [I-D.ietf-homenet-front-end-naming-delegation] a Homenet Registered Domain. Similarly to the reverse zone, ISPs are aware of who owns that domain name and may become a natural candidate for hosting the Homenet Zone - that is the Distribution Manager (DM) and the Public Authoritative Servers.

This document describes DHCPv6 options that enable an ISP to provide the necessary parameters to the HNA, to proceed. More specifically, the ISP provides the Registered Homenet Domain, necessary information on the DM and the RDM so the HNA can manage and upload the Public Homenet Zone and the Reverse Public Homenet Zone as described in [I-D.ietf-homenet-front-end-naming-delegation].

The use of DHCPv6 options may make the configuration completely transparent to the end user and provides a similar level of trust as the one used to provide the IP prefix - when provisioned via DHCP.

3. Procedure Overview

This section illustrates how a HNA receives the necessary information via DHCPv6 options to outsource its authoritative naming service to the DOI. For the sake of simplicity, and similarly to [I-D.ietf-homenet-front-end-naming-delegation], this section assumes that the HNA and the home network DHCPv6 client are collocated on the Customer Edge (CE) router [RFC7368]. Note also that this is not mandatory and the DHCPv6 client may instruct remotely the HNA and the DHCPv6 either with a proprietary protocol or a protocol that will be defined in the future. In addition, this section assumes the responsible entity for the DHCPv6 server is configured with the DM and RDM. This means a Registered Homenet Domain can be associated to the DHCPv6 client.

This scenario is believed to be the most popular scenario. This document does not ignore scenarios where the DHCPv6 server does not have privileged relations with the DM or RDM. These cases are discussed in Appendix A. Such scenarios do not necessarily require configuration for the end user and can also be zero-config.

The scenario considered in this section is as follows:

1. The HNA is willing to outsource the Public Homenet Zone or Homenet Reverse Zone. The DHCPv6 client is configured to include in its Option Request Option (ORO) the Registered Homenet Domain Option (OPTION_REGISTERED_DOMAIN), the Distribution Manager Option (OPTION_DIST_MANAGER) and the Reverse Distribution Manager Option (OPTION_REVERSE_DIST_MANAGER) option codes.
 2. The DHCPv6 server responds to the HNA with the requested DHCPv6 options based on the identified homenet. The DHCPv6 client passes the information to the HNA.
 3. The HNA is authenticated (eventually by a self signed certificate) by the DM and the RDM. The HNA builds the Homenet Zone (or the Homenet Reverse Zone) and proceed as described in [I-D.ietf-homenet-front-end-naming-delegation]. The DHCPv6 options provide the necessary non optional parameters described in Section 14 of [I-D.ietf-homenet-front-end-naming-delegation]. The HNA may complement the configurations with additional parameters via means not yet defined. Section 14 of [I-D.ietf-homenet-front-end-naming-delegation] describes such parameters that MAY take some specific non default value.
4. DHCPv6 Option

This section details the payload of the DHCPv6 options.

4.1. Registered Homenet Domain Option

The Registered Domain Option (OPTION_REGISTERED_DOMAIN) indicates the FQDN associated to the home network.

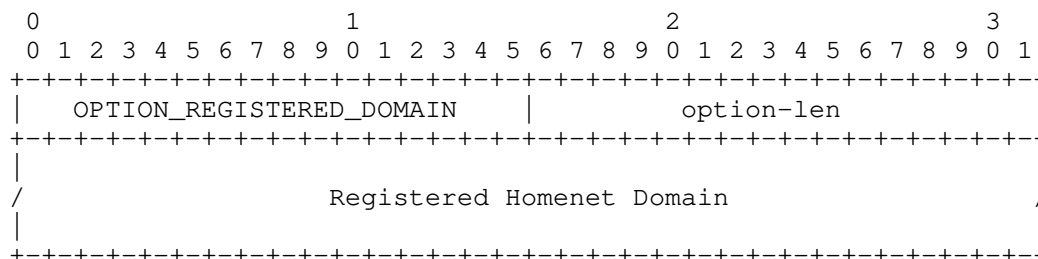


Figure 1: Registered Domain Option

- o option-code (16 bits): `OPTION_REGISTERED_DOMAIN`, the option code for the Registered Homenet Domain (TBD1).
- o option-len (16 bits): length in octets of the Registered Homenet Domain field as described in [RFC8415].
- o Registered Homenet Domain (variable): the FQDN registered for the homenet encoded as described in Section 10 of [RFC8415].

4.2. Distribution Manager Option

The Distributed Manager Option (`OPTION_DIST_MANAGER`) provides the HNA with the FQDN of the DM as well as the transport protocols for the communication between the HNA and the DM.

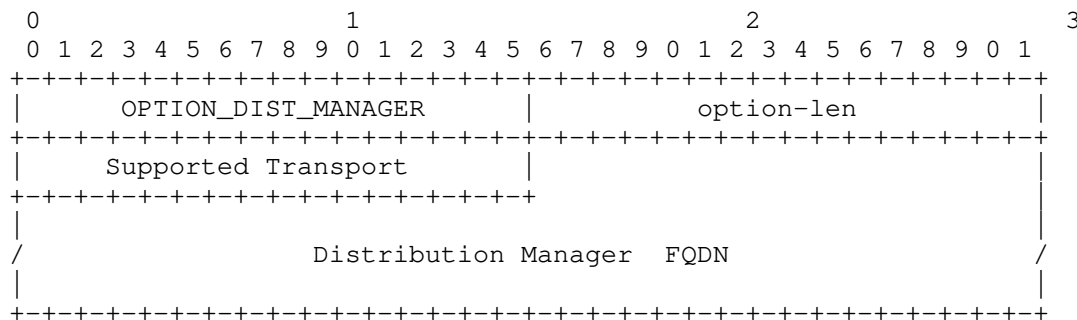


Figure 2: Distribution Manager Option

- o option-code (16 bits): `OPTION_DIST_MANAGER`, the option code for the Distribution Manager Option (TBD2).
- o option-len (16 bits): length in octets of the enclosed data as described in [RFC8415].

- o Supported Transport (16 bits): defines the supported transport by the DM (see Section 4.2.1). Each bit represents a supported transport, and a DM MAY indicate the support of multiple modes. The bit for DNS over TLS [RFC7858] MUST be set.
- o Distribution Manager FQDN (variable): the FQDN of the DM encoded as described in Section 10 of [RFC8415].

4.2.1. Supported Transport

The Supported Transport field of the DHCPv6 option indicates the supported transport protocols. Each bit represents a specific transport mechanism. A bit sets to 1 indicates the associated transport protocol is supported. The corresponding bits are assigned as described in Figure 3 and Section 6.

Bit Position	Transport Protocol	Reference
0	DNS over TLS	This-RFC
1-15	unallocated	

Figure 3: Supported Transport

DNS over TLS: indicates the support of DNS over TLS as described in [RFC7858].

4.3. Reverse Distribution Manager Server Option

The Reverse Distribution Manager Option (OPTION_REVERSE_DIST_MANAGER) provides the HNA with the FQDN of the DM as well as the transport protocols for the communication between the HNA and the DM.

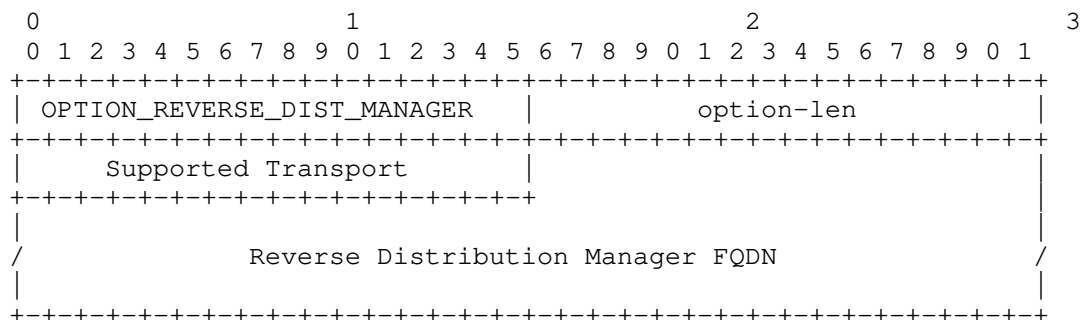


Figure 4: Reverse Distribution Manager Option

- o option-code (16 bits): OPTION_REVERSE_DIST_MANAGER, the option code for the Reverse Distribution Manager Option (TBD3).
- o option-len (16 bits): length in octets of the option-data field as described in [RFC8415].
- o Supported Transport (16 bits): defines the supported transport by the RDM (see Section 4.2.1). Each bit represents a supported transport, and a RDM MAY indicate the support of multiple modes. The bit for DNS over TLS [RFC7858] MUST be set.
- o Reverse Distribution Manager FQDN (variable): the FQDN of the RDM encoded as described in section 10 of [RFC8415].

5. DHCPv6 Behavior

5.1. DHCPv6 Server Behavior

Sections 17.2.2 and 18.2 of [RFC8415] govern server operation in regards to option assignment. As a convenience to the reader, we mention here that the server will send option foo only if configured with specific values for foo and if the client requested it. In particular, when configured the DHCPv6 server sends the Registered Homenet Domain Option, Distribution Manager Option, the Reverse Distribution Manager Option when requested by the DHCPv6 client by including necessary option codes in its ORO.

5.2. DHCPv6 Client Behavior

The DHCPv6 client includes Registered Homenet Domain Option, Distribution Manager Option, the Reverse Distribution Manager Option in an ORO as specified in Sections 18.2.1, 18.2.2, 18.2.4, 18.2.5, 18.2.6, and 21.7 of [RFC8415].

Upon receiving a DHCPv6 option described in this document in the Reply message, the HNA SHOULD proceed as described in [I-D.ietf-homenet-front-end-naming-delegation].

5.3. DHCPv6 Relay Agent Behavior

There are no additional requirements for the DHCPv6 Relay agents.

6. IANA Considerations

IANA is requested to assign the following new DHCPv6 Option Codes in the registry maintained in: <https://www.iana.org/assignments/dhcpv6-parameters/dhcpv6-parameters.xhtml#dhcpv6-parameters-2>.

Value	Description	Client ORO	Singleton Option
TBD1	OPTION_REGISTERED_DOMAIN	Yes	No
TBD2	OPTION_DIST_MANAGER	Yes	Yes
TBD3	OPTION_REVERSE_DIST_MANAGER	Yes	Yes

IANA is requested to maintain a new number space of Supported Transport parameter in the Distributed Manager Option (OPTION_DIST_MANAGER) or the Reverse Distribution Manager Option (OPTION_REVERSE_DIST_MANAGER). The different parameters are defined in Figure 3 in Section 4.2.1. Future code points are assigned under Specification Required as per [RFC8126].

7. Security Considerations

The security considerations in [RFC8415] are to be considered. The use of DHCPv6 options provides a similar level of trust as the one used to provide the IP prefix. The link between the HNA and the DHCPv6 server may benefit from additional security for example by using [I-D.ietf-dhc-sedhcpv6].

8. Acknowledgments

We would like to thank Marcin Siodelski, Bernie Volz and Ted Lemon for their comments on the design of the DHCPv6 options. We would also like to thank Mark Andrews, Andrew Sullivan and Lorenzo Colliti for their remarks on the architecture design. The designed solution has been largely been inspired by Mark Andrews's document [I-D.andrews-dnsop-pd-reverse] as well as discussions with Mark. We also thank Ray Hunter for its reviews, its comments and for suggesting an appropriated terminology.

9. Contributors

The co-authors would like to thank Chris Griffiths and Wouter Cloetens that provided a significant contribution in the early versions of the document.

10. References

10.1. Normative References

[I-D.ietf-homenet-front-end-naming-delegation]
Migault, D., Weber, R., Richardson, M., and R. Hunter,
"Simple Provisioning of Public Names for Residential
Networks", draft-ietf-homenet-front-end-naming-
delegation-14 (work in progress), April 2021.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.

10.2. Informative References

- [I-D.andrews-dnsop-pd-reverse]
Andrews, M., "Automated Delegation of IP6.ARPA reverse zones with Prefix Delegation", draft-andrews-dnsop-pd-reverse-02 (work in progress), November 2013.
- [I-D.ietf-dhc-sedhcpv6]
Li, L., Jiang, S., Cui, Y., Jinmei, T., Lemon, T., and D. Zhang, "Secure DHCPv6", draft-ietf-dhc-sedhcpv6-21 (work in progress), February 2017.
- [I-D.sury-dnsextn-cname-dname]
Sury, O., "CNAME+DNAME Name Redirection", draft-sury-dnsextn-cname-dname-00 (work in progress), April 2010.
- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, DOI 10.17487/RFC2181, July 1997, <<https://www.rfc-editor.org/info/rfc2181>>.

- [RFC6672] Rose, S. and W. Wijngaards, "DNAME Redirection in the DNS", RFC 6672, DOI 10.17487/RFC6672, June 2012, <<https://www.rfc-editor.org/info/rfc6672>>.
- [RFC7368] Chown, T., Ed., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", RFC 7368, DOI 10.17487/RFC7368, October 2014, <<https://www.rfc-editor.org/info/rfc7368>>.

Appendix A. Scenarios and impact on the End User

This section details various scenarios and discuss their impact on the end user. This section is not normative and limits the description of a limited scope of scenarios that are assumed to be representative. Many other scenarios may be derived from these.

Appendix B. Base Scenario

The base scenario is the one described in Section 3 in which an ISP manages the DHCPv6 server, the DM and RDM.

The end user subscribes to the ISP (foo), and at subscription time registers for example.foo as its Registered Homenet Domain example.foo.

In this scenario, the DHCPv6 server, DM and RDM are managed by the ISP so the DHCPv6 server and as such can provide authentication credentials of the HNA to enable secure authenticated transaction with the DM and the Reverse DM.

The main advantage of this scenario is that the naming architecture is configured automatically and transparently for the end user. The drawbacks are that the end user uses a Registered Homenet Domain managed by the ISP and that it relies on the ISP naming infrastructure.

B.1. Third Party Registered Homenet Domain

This section considers the case when the end user wants its home network to use example.com not managed by her ISP (foo) as a Registered Homenet Domain. This section still consider the ISP manages the home network and still provides example.foo as a Registered Homenet Domain.

When the end user buys the domain name example.com, it may request to redirect the name example.com to example.foo using static redirection with CNAME [RFC2181], [RFC1034], DNAME [RFC6672] or CNAME+DNAME [I-D.sury-dnsextn-cname-dname].

This configuration is performed once when the domain name example.com is registered. The only information the end user needs to know is the domain name assigned by the ISP. Once this configuration is done no additional configuration is needed anymore. More specifically, the HNA may be changed, the zone can be updated as in Appendix B without any additional configuration from the end user.

The main advantage of this scenario is that the end user benefits from the Zero Configuration of the Base Scenario Appendix B. Then, the end user is able to register for its home network an unlimited number of domain names provided by an unlimited number of different third party providers. The drawback of this scenario may be that the end user still rely on the ISP naming infrastructure. Note that the only case this may be inconvenient is when the DNS servers provided by the ISPs results in high latency.

B.2. Third Party DNS Infrastructure

This scenario considers that the end user uses example.com as a Registered Homenet Domain, and does not want to rely on the authoritative servers provided by the ISP.

In this section we limit the outsourcing to the DM and Public Authoritative Server(s) to a third party. The Reverse Public Authoritative Server(s) and the RDM remain managed by the ISP as the IP prefix is managed by the ISP.

Outsourcing to a third party DM can be performed in the following ways:

1. Updating the DHCPv6 server Information. One can imagine a GUI interface that enables the end user to modify its profile parameters. Again, this configuration update is done once-for-ever.
2. Upload the configuration of the DM to the HNA. In some cases, the provider of the CE router hosting the HNA may be the registrar and provide the CE router already configured. In other cases, the CE router may request the end user to log into the registrar to validate the ownership of the Registered Homenet Domain and agree on the necessary credentials to secure the communication between the HNA and the DM. As described in [I-D.ietf-homenet-front-end-naming-delegation], such settings could be performed in an almost automatic way as to limit the necessary interactions with the end user.

B.3. Multiple ISPs

This scenario considers a HNA connected to multiple ISPs.

Suppose the HNA has been configured each of its interfaces independently with each ISPS as described in Appendix B. Each ISP provides a different Registered Homenet Domain.

The protocol and DHCPv6 options described in this document are fully compatible with a HNA connected to multiple ISPs with multiple Registered Homenet Domains. However, the HNA should be able to handle different Registered Homenet Domains. This is an implementation issue which is outside the scope of the current document.

If a HNA is not able to handle multiple Registered Homenet Domains, the HNA may remain connected to multiple ISP with a single Registered Homenet Domain. In this case, one entity is chosen to host the Registered Homenet Domain. This entity may be one of the ISP or a third party. Note that having multiple ISPs can be motivated for bandwidth aggregation, or connectivity fail-over. In the case of connectivity fail-over, the fail-over concerns the access network and a failure of the access network may not impact the core network where the DM and Public Authoritative Primaries are hosted. In that sense, choosing one of the ISP even in a scenario of multiple ISPs may make sense. However, for sake of simplicity, this scenario assumes that a third party has been chosen to host the Registered Homenet Domain. Configuration is performed as described in Appendix B.1 and Appendix B.2.

With the configuration described in Appendix B.1, the HNA is expect to be able to handle multiple Homenet Registered Domain, as the third party redirect to one of the ISPs servers. With the configuration described in Appendix B.2, DNS zone are hosted and maintained by the third party. A single DNS(SEC) Homenet Zone is built and maintained by the HNA. This latter configuration is likely to match most HNA implementations.

The protocol and DHCPv6 options described in this document are fully compatible with a HNA connected to multiple ISPs. To configure or not and how to configure the HNA depends on the HNA facilities. Appendix B and Appendix B.1 require the HNA to handle multiple Registered Homenet Domain, whereas Appendix B.2 does not have such requirement.

Authors' Addresses

Daniel Migault
Ericsson
8275 Trans Canada Route
Saint Laurent, QC 4S 0B6
Canada

EMail: daniel.migault@ericsson.com

Ralf Weber
Akamai

EMail: ralf.weber@akamai.com

Tomek Mrugalski
Internet Systems Consortium, Inc.
950 Charter Street
Redwood City 94063
US

EMail: tomasz.mrugalski@gmail.com

Internet Engineering Task Force
Internet-Draft
Intended status: Best Current Practice
Expires: 27 October 2022

T. Lemon
Apple Inc.
25 April 2022

Connecting Stub Networks to Existing Infrastructure
draft-lemon-stub-networks-03

Abstract

This document describes a set of practices for connecting stub networks to adjacent infrastructure networks, as well as to larger network fabrics. This is applicable in cases such as constrained (Internet of Things) networks where there is a need to provide functional parity of service discovery and reachability between devices on the stub network and devices on an adjacent infrastructure link (for example, a home network).

The stub networks use case is intended to fully address the need to attach a single network link to an infrastructure network, where the attached link provides no through routing and in cases where integration to the infrastructure routing fabric (if any) is not available.

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 <https://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 27 October 2022.

Copyright Notice

Copyright (c) 2022 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	3
2. Glossary	3
3. Support for adjacent infrastructure links	4
3.1. Managing addressability on the adjacent infrastructure link	4
3.1.1. IP addressability already present on adjacent infrastructure link	4
3.1.2. IP addressability not present on adjacent infrastructure link	6
3.1.3. Resolving contention over which prefix to deprecate	6
3.1.4. Handling the presence of multiple stub routers	7
3.2. Managing addressability on the stub network	7
3.2.1. Maintenance across stub router restarts	8
3.2.2. Generating a ULA prefix to provide addressability	9
3.3. Managing reachability on the adjacent infrastructure link	9
3.4. Managing reachability on the stub network	9
3.5. Providing discoverability of stub network hosts on the adjacent infrastructure link	10
3.6. Providing discoverability of adjacent infrastructure hosts on the stub network	12
4. Providing reachability to IPv4 services to the stub network	12
4.1. NAT64 provided by infrastructure	12
4.2. NAT64 provided by stub router(s)	13
5. Handling partitioning events on a stub network	14
6. Support for non-adjacent links	14
6.1. Acquiring an off-stub-network-routable prefix for the stub network	15
6.2. Arranging for routing to a stub network's off-stub-network routable prefix	16
6.3. Making service advertisements available on non-adjacent infrastructure	16
6.4. Making service advertisements available on the internet	16

6.5. Distinction between non-adjacent infrastructure and global internet connectivity	17
7. Normative References	17
Author's Address	18

1. Introduction

This document describes a set of practices for connecting stub networks to adjacent infrastructure networks, as well as to larger network fabrics. This is applicable in cases such as constrained (Internet of Things) networks where there is a need to provide functional parity of service discovery and reachability between devices on the stub network and devices on an adjacent infrastructure link (for example, a home network).

The stub networks use case is intended to fully address the need to attach a single network link to an infrastructure network, where the attached link provides no through routing and in cases where integration to the infrastructure routing fabric (if any) is not available.

2. Glossary

Addressability The ability to associate each node on a link with its own IPv6 address.

Reachability Given an IPv6 destination address that is not on-link for any link to which a node is attached, the information required that allows the node to send packets to a router that can forward those packets towards a link where the destination address is on-link.

Infrastructure network the network infrastructure to which a stub router connects. This network can be a single link, or a network of links. The network may also provide some services, such as a DNS resolver, a DHCPv4 server, and a DHCPv6 prefix delegation server, for example.

Infrastructure link any link in a network infrastructure that is managed by a single entity.

Adjacent infrastructure link (AIL) an infrastructure link to which a stub router is directly connected.

Non-adjacent infrastructure link (NAIL) an infrastructure link to which a stub router is not directly connected.

Non-adjacent link (NAL) any link to which the stub router is not

directly connected, whether within an infrastructure or elsewhere on the Internet.

Off-Stub-Network-Routable (OSNR) Prefix a prefix advertised on the stub network that can be used for communication with hosts not on the stub network.

3. Support for adjacent infrastructure links

We assume that adjacent infrastructure link supports Router and Prefix Discovery using router advertisements. Adjacent infrastructure links on networks where this is not supported are out of scope for this document.

3.1. Managing addressability on the adjacent infrastructure link

In order to provide IPv6 routing to the stub network, IPv6 addressing must be available on the adjacent infrastructure link. In the ideal case, such addressing is already present on the link, and need not be provided. In this case, the stub router SHOULD NOT provide addressability on the adjacent infrastructure link.

3.1.1. IP addressability already present on adjacent infrastructure link

IPv6 addressing is considered to be present on the link if a usable on-link prefix is advertised on the adjacent infrastructure link. A usable on-link prefix could be a prefix advertised on the link that is on-link and allows autonomous configuration. A prefix is also a usable on-link prefix if it is advertised on the link as on-link, and if the 'm' bit is set in the Router Advertisement message header ([RFC4861], Section 4.2) that contains the Prefix option. This indicates that node addressability is being managed using DHCPv6.

A prefix is advertised on the link if, when a Router Solicit message ([RFC4861], Section 4.1) is sent, a Router Advertisement message is received in response which contains a prefix information option ([RFC4861], Section 4.6.2) for that prefix.

After such an RA message has been received, it can be assumed for some period of time thereafter that the prefix is still valid on the link. However, prefix lifetimes and router lifetimes are often quite long. The mere fact that a prefix that has been advertised is still within its valid lifetime does not mean that that prefix is still being advertised on the link.

This is important because when a new host appears on the adjacent infrastructure link and sends an initial router solicit, if it does not receive a usable on-link prefix, it will not be able to communicate. Consequently, the stub router MUST monitor router solicits and advertisements on the link in order to determine whether a prefix that has been advertised on the link is still being advertised.

There are several methods that can be used to accomplish this:

The stub router MUST listen for router advertisements on the adjacent infrastructure link, and record the time at which each router advertisement was received. A router advertisement that is more than STALE_RA_TIME seconds old MUST be assumed to no longer be advertised on the link. When the last non-stale router advertisement containing a usable prefixes on the link is marked stale, the stub router should begin Router Discovery ([RFC4861], Section 6.3).

The stub router MUST listen for router solicits on the adjacent infrastructure link. When a router solicit is received, the router SHOULD set a timer for VICARIOUS_SOLICIT_TIME seconds. If, after that amount of time, no router advertisements are received that contain a usable on-link prefix, the stub router MUST begin router discovery. This is necessary in case the response to the router solicit was unicast, since in this case the stub router would not see that response. When the stub router first connects to the adjacent infrastructure link, it MUST begin router discovery.

When router discovery completes, the stub router evaluates whether or not a usable on-link prefix has been seen in a non-stale router advertisement during router discovery. If no usable on-link prefix has been seen, then the stub router MUST begin to provide a usable on-link prefix.

As an alternative to the vicarious router discovery process described here, the stub router could monitor the presence of the router advertising the on-link prefix in the neighbor cache. If the neighbor cache entry becomes stale, this could be an indication that the prefix is also stale. If the neighbor cache entry goes stale, the router would need to try to refresh it, and if that fails, then the stub router must begin advertising its own on-link prefix on the stub network.

3.1.2. IP addressability not present on adjacent infrastructure link

When there is no usable on-link prefix on the adjacent infrastructure network, the stub router provides its own on-link prefix. This prefix has a valid and preferred lifetime of `STUB_PROVIDED_PREFIX_LIFETIME` seconds. This prefix **MUST** allow for autonomous configuration (SLAAC).

The stub router must advertise this prefix every `BEACON_INTERVAL` seconds. When the stub router is advertising reachability to the stub network, the on-link prefix advertisement and the route information advertisement must be contained in the same router advertisement.

When the stub router is advertising an on-link prefix on the AIL, it may receive a router advertisement containing a usable on-link prefix for the AIL with a non-zero preferred lifetime. In this case, the stub router should begin to deprecate the on-link prefix it is advertising on the AIL. The preferred lifetime for this prefix should be set to zero in subsequent advertisements.

The valid lifetime (`VALID`) is computed based on three values: the current time when a router advertisement is being generated (`NOW`), the time at which the new usable on-link prefix advertisement was received (`DEPRECATE_TIME`), and `STUB_PROVIDED_PREFIX_LIFETIME`. All of these values are in seconds. `VALID` is computed as follows:

$$\text{VALID} = \text{STUB_PROVIDED_PREFIX_LIFETIME} - (\text{NOW} - \text{DEPRECATE_TIME})$$

If `VALID` is less than `BEACON_INTERVAL`, the stub router does not include the deprecated prefix in the router advertisement. Note that `VALID` could be less than zero. Otherwise, the prefix is provided in the advertisement, but with a valid lifetime of `VALID`.

3.1.3. Resolving contention over which prefix to deprecate

It is also possible that all routers on the link that are capable of advertising prefixes might be following the same protocol of deprecating their own prefix when a valid prefix shows up. To prevent a situation where all routers deprecate their prefix and wait until there are no valid prefixes being advertised before advertising a prefix, each stub router must detect the situation where, having deprecated its own prefix, all of the other prefixes being advertised on the link have also been deprecated.

When this situation occurs, each router on the link MUST compare the valid lifetimes of all the prefixes that have been seen. If the router's own prefix expires last, then that router should immediately resume publishing its prefix as a preferred prefix.

If a router observes this situation and its prefix is not the one that expires last, it MUST set a timer for UNDEPRECATE_WAIT seconds, while continuing to observe prefix advertisements on the link. If, when the timer expires, the prefix that expires last has not been re-published as a preferred prefix, then that prefix is marked as 'really deprecated', and no longer considered a candidate for deprecation.

Using the remaining list of prefixes, the router should then apply the same algorithm. It should continue to apply this algorithm until either its prefix becomes the one to re-publish as preferred, or some other router has re-published its prefix as preferred.

3.1.4. Handling the presence of multiple stub routers

When multiple stub routers are connected to the same AIL, and no usable on-link prefix is being provided on that link by the infrastructure, there will be a competition between routers to provide a usable on-link prefix. In order to avoid duplication, stub routers MUST include a random offset in the time interval across which router discovery is performed. This ensures that after a power failure, not all stub routers will exit router discovery at the exact same time, and so one stub router should advertise a usable on-link prefix before the others. This should prevent the other stub routers from advertising additional on-link prefixes.

There is no particular harm caused by advertising multiple on-link prefixes, but it is preferable to minimize this, because each on-link prefix consumes space in every on-link host's routing table, and consumes time when making source address selection and routing decisions.

3.2. Managing addressability on the stub network

How addressability is managed on stub networks depends on the nature of the stub network. For some stub networks, the stub router can be sure that it is the only router. For example, a stub router that is providing a Wi-Fi network for tethering will advertise its own SSID and use its own joining credentials; in this case, it can assume that it is the only router for that network, and advertise a default route and on-link prefix just like any other router.

However, some stub networks are more cooperative in nature, for example IP mesh networks. On such networks, multiple stub routers may be present and be providing addressability and reachability.

In either case, some stub router connected to the stub network MUST provide a usable on-link prefix (the OSNR prefix) for the stub network. If the stub network is a multicast-capable medium where Router Advertisements are used for router discovery, the same mechanism described in section [Support for adjacent infrastructure links] is used.

Stub networks that do not support the use of Router Advertisements for router discovery must use some similar mechanism that is compatible with that type of network. Describing the process of establishing a common OSNR prefix on such networks is out of scope for this document.

3.2.1. Maintenance across stub router restarts

Stub routers may restart from time to time; when a restart occurs, the stub router may have been advertising state to the network which, following the restart, is no longer required.

For example, suppose there are two stub routers connected to the same infrastructure link. When the first stub router is restarted, the second takes over providing an on-link prefix. Now the first router rejoins the link. It sees that the second stub router's prefix is advertised on the infrastructure link, and therefore does not advertise its own.

This behavior can cause problems because the first stub router no longer sees the on-link prefix it had been advertising on infrastructure as on-link. Consequently, if it receives a packet to forward to such an address, it will forward that packet directly to a default router, if one is present; otherwise, it will have no route to the destination, and will drop the packet.

To address this problem, stub routers SHOULD remember the last time a prefix was advertised across restarts. On restart, the router can immediately begin deprecating the prefix, and can stop after the prefix valid lifetime goes to zero, based on the recorded time that the last advertisement was sent.

When a stub router has only flash memory with limited write lifetime, it may be inappropriate to do a write to flash every time a prefix beacon happens. In this case, the router SHOULD record the set of prefixes that have been advertised on infrastructure and the maximum valid lifetime that was advertised. On restart, the router should

assume that hosts on the infrastructure link have received advertisements for any such prefixes, and should immediately deprecate them, and continue to do so until the maximum valid lifetime has elapsed after restart.

3.2.2. Generating a ULA prefix to provide addressability

In order to be able to provide addressability either on the stub network or on an adjacent infrastructure network, a stub router must allocate its own ULA prefix. ULA prefixes, described in Unique Local IPv6 Unicast Addresses ([RFC4193]) are randomly allocated prefixes. A stub router **MUST** allocate a single ULA prefix for use in providing on-link prefixes to the stub network and the infrastructure network, as needed.

The ULA prefix allocated by a stub router **SHOULD** be maintained across reboots, and **SHOULD** remain stable over time. For privacy reasons, a stub router that roams from network to network may wish to allocate a different ULA prefix each time it connects to a different infrastructure network.

If IPv6 prefix delegation is available, which implies that IPv6 service is also available on the infrastructure link, then the stub router **MAY** use IPv6 prefix delegation to acquire a prefix to advertise on the stub network, rather than allocating one out of its ULA prefix.

3.3. Managing reachability on the adjacent infrastructure link

Stub routers **MUST** advertise reachability to stub network OSNR prefixes on any AIL to which they are connected.

Each stub network will have some set of prefixes that are advertised as on-link for that network. A stub router connected to that network **SHOULD** advertise reachability to all such prefixes on any AIL to which it is attached using router advertisements

3.4. Managing reachability on the stub network

The stub router **MAY** advertise itself as a default router on the stub network, if it itself has a default route on the AIL. In some cases it may not be desirable to advertise reachability to the Internet as a whole; in this case the stub router need not advertise itself as a default router.

If the stub router is not advertising itself as a default on the stub network, it MUST advertise reachability to any prefixes that are being advertised as on-link on AILs to which it is attached. This is true for prefixes it is advertising, and for other prefixes being advertised on that link.

Note that in some stub network configurations, it is possible for more than one stub router to be connected to the stub network, and each stub router may be connected to a different AIL. In this case, a stub router advertising a default route may receive a packet destined for a link that is not an AIL for that router, but is an AIL for a different router. In such a case, if the infrastructure is not capable of routing between these two AILs, a packet which could have been delivered by another stub router will be lost by the stub router that received it.

Consequently, stub routers SHOULD be configurable to not advertise themselves as default routers on the stub network. Stub routers SHOULD be configurable to explicitly advertise AIL prefixes on the stub network even if they are advertising as a default router. Stub routers SHOULD be configurable to advertise NAIL prefixes on the stub network; such configuration would include a list of NAIL prefixes to advertise. This list may be configured in a management interface or as a result of these routes being delivered in a routing protocol or through router discovery. The mechanisms by which such configuration can be accomplished are out of scope for this document.

3.5. Providing discoverability of stub network hosts on the adjacent infrastructure link

In some cases it will be necessary for hosts on the adjacent infrastructure link to be able to discover devices on the stub network. In other cases, this will be unnecessary or even undesirable. For example, it may be undesirable for devices on an adjacent infrastructure link to be able to discover devices on a Wi-Fi tether, for example provided by a mobile phone.

One example of a use case for stub networks where such discovery is desirable is the constrained network use case. In this case a low-power, low-cost stub network provides connectivity for devices that provide services to the infrastructure. For such networks, it is necessary that devices on the infrastructure be able to discover devices on the stub network.

The most basic use case for this is to provide feature parity with existing solutions like multicast DNS (mDNS). For example, a light bulb with built-in Wi-Fi connectivity might be discoverable on the infrastructure link to which it is connected, using mDNS, but likely

is not discoverable on other links. To provide equivalent functionality for an equivalent device on a constrained network that is a stub network, the stub network device must be discoverable on the infrastructure link (which is an AIL from the perspective of the stub network).

If services are to be advertised using DNS Service Discovery [RFC6763], there are in principle two ways to accomplish this. One is to present services on the stub network as a DNS zone which can then be configured as a browsing domain in the DNS ([RFC6763], Section 11). The second is to advertise stub network services on the AIL using multicast DNS (mDNS) [RFC6762].

Stub network routers cannot be assumed to be able to integrate into the DNS naming hierarchy of the infrastructure network. Therefore, stub networks must be able to rely on ad-hoc service advertisement protocols. Since mDNS is in wide use, this is a suitable protocol for this use case. This is not to say that mDNS is the only such protocol that could be used, but it is the one that we suggest implementing.

In order to provide mDNS discovery for devices on the stub network, one of two solutions is likely to be applicable, depending on the operational practicalities of the stub network. For a constrained stub network, on which battery operated devices may be attached, mass multicast traffic for service discovery is impractical, since every device needs to wake up for every service discovery, even if they don't offer that service, and since many such devices may be operating on battery power. For such a network, multicast DNS is not a good choice.

For such networks, a unicast service registration protocol such as DNS-SD Service Registration Protocol (SRP) [I-D.ietf-dnssd-srp] is a good solution. The stub router can act as an SRP server on the stub network, accepting service advertisements from stub network devices. On the adjacent infrastructure network, it can advertise those services as multicast DNS Advertising Proxy [I-D.sctl-advertising-proxy].

For other stub networks, for example a Wi-Fi-based Personal Area Network provided as part of a tethering function on a mobile device, multicast DNS may be the only option. For Wi-Fi stub networks, there is such a large installed base of devices supporting mDNS that requiring some other service advertisement solution would be problematic simply because it would require new software for that entire installed base. For other networks, particularly constrained networks, where devices do not currently support mDNS, no such obstacle exists.

Because the primary use case for discovery of devices on a stub network is the use case where the stub network is joining a constrained network to an existing infrastructure link, we currently only describe a solution (DNS-SD SRP) for that use case. A solution for the use case where the stub router must provide discoverability for a stub network where mDNS advertising is preferred is out of scope for this document.

3.6. Providing discoverability of adjacent infrastructure hosts on the stub network

Hosts on the stub network may need to discover hosts on the adjacent infrastructure network. In the IoT network example we've been using, there might be a light switch on the stub network which needs to be able to actuate a light bulb connected to the adjacent infrastructure network. In order to know where to send the actuation messages, the light switch will need to be able to discover the light bulb's address somehow.

In the case of a Wi-Fi stub network, devices on the stub network will need to be able to access the Internet, and may also need to be able to access local services on the adjacent infrastructure link.

In order to address these use cases, the stub network router SHOULD provide a DNS-SD Discovery Proxy [RFC8766] and a DNS resolver. Since these two functions are combined, if the stub router provides them, it MUST offer both services on the standard DNS UDP and TCP ports.

4. Providing reachability to IPv4 services to the stub network

4.1. NAT64 provided by infrastructure

Stub networks are defined to be IPv6-only because it would be difficult to implement a stub network using IPv4 technology. However, stub network devices may need to be able to communicate with IPv4-only services either on the adjacent infrastructure, or on the global internet. Ideally, the infrastructure network fully supports IPv6, and all services on the infrastructure network are IPv6-capable. In this case, perhaps the infrastructure network provides NAT64 service to IPv4-only hosts on the internet. In this ideal setting, the stub router need do nothing-the infrastructure network is doing it all.

In this situation, if there are multiple stub routers, each connected to the same adjacent infrastructure link, there is no need for special behavior-each stub router can advertise a default route, and any stub router will do to route NAT64 traffic. If some stub routers are connected to different adjacent infrastructure links than others,

some of which support NAT64 and some of which do not, then the default route may not carry traffic to the correct link for NAT64 service. In this case, a more specific address to the infrastructure NAT64 prefix(es) MUST be advertised by those stub routers that are able to discover it.

4.2. NAT64 provided by stub router(s)

Most infrastructure networks at present do not provide NAT64 service. It is therefore necessary for stub routers to be able to provide NAT64 service if IPv4 hosts are to be reachable from the stub network.

To provide NAT64 service, a stub router must allocate a NAT64 prefix. For convenience, the stub network allocates a single prefix out of the /48 ULA prefix that it maintains. Out of the 2^{16} possible subnets of the /48, the stub router SHOULD use the numerically highest /64 prefix.

If there are multiple stub routers providing connectivity between the stub network and infrastructure, each stub network uses its own NAT64 prefix--there is no common NAT64 prefix. The reason for this is that NAT64 translation is not stateless, and is tied to the stub router's IPv4 address. Therefore each NAT64 egress is not equivalent.

A stub network that services a Wi-Fi stub network SHOULD provide DNS64 translation: hosts on the stub network cannot be assumed to be able to do DNS64 synthesis in the stub resolver. In this case the DNS resolver on the stub router MUST honor the CD and DO bits if received in a request, since this indicates that the stub resolver on the requestor intends to do DNSSEC validation. In this case, the resolver on the stub router MUST NOT perform DNS64 synthesis.

On specific stub networks it may be desirable to require the stub network device to perform DNS64 synthesis. Stub network routers for such networks do not need to provide DNS64 synthesis. Instead, they MUST provide an `ipv4only.arpa` answer that advertises the NAT64 prefix for that stub router, and MUST provide an explicit route to that NAT64 prefix on the stub network using RA or whatever technology is specific to that stub network type.

In constrained networks it can be very useful if stub network resolvers provide the information required to do DNS64 translation in the answer to the AAAA query. If the answer to an AAAA query comes back with "no data" (not NXDOMAIN), this suggests that there may be an A record. In this case, the stub network's resolver SHOULD attempt to look up an A record on the same name. If such a record exists, the resolver SHOULD return no data in the Answer section of

the DNS response, and SHOULD provide any CNAME records that were involved in returning the "no data" answer to the AAAA query, and SHOULD provide any A records that were ultimately returned, in the Additional section. The resolver should also include an ipv4only.arpa record in the Additional section.

5. Handling partitioning events on a stub network

If a stub network is constructed using mesh technology, it may become partitioned. In such a situation, it may be one stub router is connected to one partition, and another stub router is connected to the other partition. In this situation, in order for all nodes to be reachable, it is necessary that each partition of the stub network have its own prefix. When such a partition occurs, the stub routers must detect that it has occurred. If a stub router is currently providing a prefix on the stub network, it need take no action. If a stub router had not been providing a prefix on the stub network, and now discovers that there is no stub router providing a prefix on the network, it MUST begin to provide its own prefix on the stub network. It MUST also advertise reachability to that new prefix on its adjacent infrastructure link(s).

When partitions of this type occur, they may also heal. When a partition heals in a situation where two stub routers have both been advertising a prefix, it will now appear that there are two prefixes on the stub network. Since partition events may represent a recurring situation, stub routers SHOULD wait for at least PARTITION_HEAL_WAIT_TIME before deprecating one of these prefixes.

When the time comes to deprecate one or more prefixes as a result of a network partition healing, only one prefix should remain. If there are any GUA prefixes, and if there is no specific configuration contradicting this, the GUA prefix that is numerically lowest should be kept, and all others deprecated. If there are no GUA prefixes, then the ULA prefix that is numerically lowest should be kept, and the others deprecated. By using this approach, it is not necessary for the routers to coordinate in advance.

6. Support for non-adjacent links

There are two ways that connectivity to non-adjacent links can be established. The first is that if the infrastructure network as a whole has a working IPv4 routing fabric, NAT64 can be used to enable hosts on the stub network to establish communications with hosts on non-adjacent links, including the Internet. In some cases, this is all that is needed.

However, if it will be necessary for nodes on non-adjacent networks to establish communications with nodes on the stub network, this will require a working IPv6 routing fabric connecting the stub network to any non-adjacent links from which communications will need to be established.

In order for such routing to work, the stub network will also need to acquire a prefix that the infrastructure network is aware of and can route to. The ULA prefix that can work for communicating to adjacent infrastructure links will not work for communicating to non-adjacent links.

6.1. Acquiring an off-stub-network-routable prefix for the stub network

A prefix may be acquired by using DHCPv6 Prefix Delegation ([RFC8415], Section 6.3). The stub router then advertises this prefix as the on-link prefix for the stub network, as before. It also advertises reachability to this prefix using router advertisements, as before.

In the case where there is more than one stub router, it would be best if only one stub router requested a delegated prefix. This can be managed through the mechanism described earlier: the stub router only acquires a prefix to advertise when it has decided that it needs to advertise a prefix, and so in most cases only one stub router at a time will request a delegated prefix.

In order to avoid excessive consumption of delegated prefixes, stub routers connected to stub networks that support multiple stub routers SHOULD request short lifetimes for delegated prefixes and renew frequently. Stub routers SHOULD request a lifetime of PREFIX_DELEGATION_INTERVAL. Stub routers SHOULD record the time that a prefix was acquired in stable storage, and SHOULD release the prefix using a "DHCP Release" transaction when shutting down, or when it determines that a prefix is no longer needed (See "graceful shutdown" in Figure 9 of [RFC8415] for details). Stub routers SHOULD release any remembered still-valid prefix after reboot, if after rebooting it is discovered that another prefix is being advertised on the stub network.

6.2. Arranging for routing to a stub network's off-stub-network routable prefix

We can assume that a side effect of the prefix delegation process will be to establish routing to the stub router that requested the prefix. This should mean that any node that wishes to establish communication with a node on the stub network will be able to do so through the delegating router that provides the prefix or, if it is attached to an infrastructure link that is adjacent to the stub router, through the stub router itself by means of the router advertisement it is providing.

The case of multiple stub routers is more complicated however. Any routing that comes as a side-effect of DHCPv6 Prefix Delegation will only route through the stub router that acquired the prefix. Other stub routers can provide reachability on their respective adjacent infrastructure links, but reachability across the full routing fabric of the infrastructure network will only be possible if there is some routing protocol present on the infrastructure network. Addressing this problem is out of scope for this document.

6.3. Making service advertisements available on non-adjacent infrastructure

In order for service advertisements to be available on non-adjacent infrastructure, the infrastructure must provide SRP service for constrained stub networks, and must advertise the availability of such service so that stub routers can forward SRP updates to that SRP service, rather than providing SRP as a local service. This SRP service can be discovered using DNS-SD, using the `_dnssd-srp-tls` service type. If the stub network requires UDP-based SRP rather than tls-based SRP, the stub router MUST act as a proxy to deliver SRP updates over the tcp+tls transport.

For stub networks that use multicast DNS, stub routers must provide a discovery proxy service, and must advertise that service to the infrastructure. In turn, the infrastructure must configure that service to be discoverable by devices on the infrastructure, as described in [RFC8766], Section 6.

6.4. Making service advertisements available on the internet

The mechanism described previously for making service advertisements available to non-adjacent infrastructure also scales to the internet, since it uses DNS. Indeed, the question an operator should ask before enabling such discovery is, do they want their stub network devices to be discoverable on the internet. If it becomes possible to configure service advertising automatically, behavior similar to

that specified in [RFC6092], Section 3.2 and 3.3, would be advised: do not automatically advertise stub network devices on the Internet.

6.5. Distinction between non-adjacent infrastructure and global internet connectivity

Stub routers may be mobile, or fixed. That is, they may move from location to location along with some or all of their connected devices, attaching to whatever infrastructure is available. Or they may be fixed devices that are only ever expected to exist in one particular location.

For devices that are intended to be in a fixed location, the distinction between infrastructure links and the internet as a whole is meaningful; for mobile nodes it most likely is not, unless such a node is only going to ever attach to trusted infrastructure as it moves from location to location-not a common scenario.

For fixed links, the infrastructure may be trusted, in which case the distinction between infrastructure and internet can be expected to be managed by the infrastructure, and therefore only visible to the stub router in the sense that some non-adjacent destinations may be reachable (infrastructure destinations, for example) while others are not.

The reason for mentioning this here is to point out that the stub router can't be expected to manage this interface: it is up to the infrastructure network to do so, either implicitly or explicitly. [RFC7084] provides a set of default behaviors for home routers that may be adequate for automatically managing this interface, but further work in this area may be warranted.

7. Normative References

- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <<https://www.rfc-editor.org/info/rfc4861>>.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, DOI 10.17487/RFC4193, October 2005, <<https://www.rfc-editor.org/info/rfc4193>>.
- [RFC6092] Woodyatt, J., Ed., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, DOI 10.17487/RFC6092, January 2011, <<https://www.rfc-editor.org/info/rfc6092>>.

- [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>>.
- [RFC7084] Singh, H., Beebe, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, DOI 10.17487/RFC7084, November 2013, <<https://www.rfc-editor.org/info/rfc7084>>.
- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.
- [RFC8766] Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", RFC 8766, DOI 10.17487/RFC8766, June 2020, <<https://www.rfc-editor.org/info/rfc8766>>.
- [I-D.ietf-dnssd-srp]
Lemon, T. and S. Cheshire, "Service Registration Protocol for DNS-Based Service Discovery", Work in Progress, Internet-Draft, draft-ietf-dnssd-srp-12, 24 October 2021, <<https://datatracker.ietf.org/doc/html/draft-ietf-dnssd-srp-12>>.
- [I-D.sctl-advertising-proxy]
Cheshire, S. and T. Lemon, "Advertising Proxy for DNS-SD Service Registration Protocol", Work in Progress, Internet-Draft, draft-sctl-advertising-proxy-02, 12 July 2021, <<https://datatracker.ietf.org/doc/html/draft-sctl-advertising-proxy-02>>.

Author's Address

Ted Lemon
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America
Email: mellon@fugue.com