

DHC Work Group
Internet-Draft
Intended status: Standards Track
Expires: June 19, 2020

I. Farrer
Deutsche Telekom AG
Naveen. Kottapalli
Benu Networks
M. Hunek
Technical University of Liberec
Richard. Patterson
December 17, 2019

DHCPv6 Prefix Delegating relay
draft-fkhp-dhc-dhcpv6-pd-relay-requirements-03

Abstract

Operational experience with DHCPv6 prefix delegation has shown that when the DHCPv6 relay function is not co-located with the DHCPv6 server function, issues such as timer synchronization between the DHCP functional elements, rejection of client's messages by the relay, and other problems have been observed. These problems can result in prefix delegation failing or traffic to/from clients addressed from the delegated prefix being unrouteable. Although [RFC8415] mentions this deployment scenario, it does not provide necessary detail on how the relay element should behave when used with PD.

This document describes functional requirements for a DHCPv6 PD relay when used for relaying prefixes delegated by a separate DHCPv6 server.

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 June 19, 2020.

Internet-Draft

DHCPv6 PD relay

December 2019

Copyright Notice

Copyright (c) 2019 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	2
2.	Terminology	3
2.1.	General	3
2.2.	Topology	4
2.3.	Requirements Language	5
3.	Problems Observed with Existing Delegating Relays Implementations	5
3.1.	DHCP Messages not being Forwarded by the Delegating relay	5
3.2.	Delegating Relay Loss of State on Reboot	5
3.3.	Multiple Simultaneous Delegated Prefixes for a Single DUID on a Single Client	6
3.4.	Dropping Messages from Devices with Duplicate MAC addresses and DUIDs	6
4.	Requirements for Delegating Relays	6
4.1.	General Requirements	6
4.2.	Routing Requirements	7
4.3.	Service Continuity Requirements	8
4.4.	Operational Requirements	8
5.	Acknowledgements	9
6.	IANA Considerations	9
7.	Security Considerations	9
8.	References	9
8.1.	Normative References	9
8.2.	Informative References	10
	Authors' Addresses	10

1. Introduction

For internet service providers that offer native IPv6 access with prefix delegation to their customers, a common deployment architecture is to have a DHCPv6 relay agent function located in the

ISP's Layer-3 customer edge device and separate, centralized DHCPv6 server infrastructure. [[RFC8415](#)] describes the functionality of a DHCPv6 relay and [Section 19.1.3](#) mentions the deployment scenario, but does not provide detail for all of the functional requirements that the relay needs to fulfill to operate deterministically in this deployment scenario.

A DHCPv6 relay agent for prefix delegation is a function commonly implemented in routing devices, but implementations vary in their functionality and client/server inter-working. This can result in operational problems such as messages not being forwarded by the relay or unreachability of the delegated prefixes. This document provides a set of requirements for devices implementing a relay function for use with prefix delegation.

The mechanisms for the redistribution of remote routes learnt via DHCP PD is out of scope of the document. Multi-hop relaying is also not considered as the functionality is solely required by a DHCP relay agent that is co-located with the first-hop router that the DHCPv6 client requesting the prefix is connected to.

The behavior defined in [[RFC7283](#)] is also applicable for DHCPv6-PD-relay deployments.

2. Terminology

2.1. General

This document uses the terminology defined in [[RFC8415](#)], however when defining the functional elements for prefix delegation [[RFC8415](#)], [Section 4.2](#) defines the term 'delegating router' as:

"The router that acts as a DHCP server and responds to requests for delegated prefixes."

This document is concerned with deployment scenarios in which the

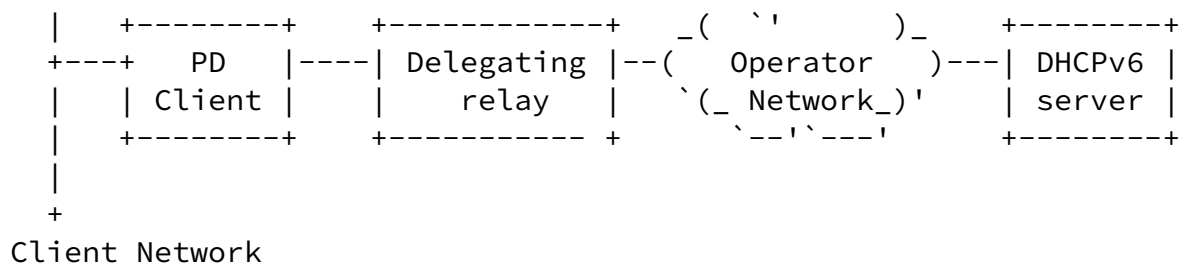


Figure 1

The client request prefixes via the client facing interface of the delegating relay. The resulting prefixes will be used for addressing the client network. The delegating relay is responsible for forwarding DHCP messages, including prefix delegation requests and responses between the client and server. Messages are forwarded from the delegating relay to the server using multicast or unicast via the operator network facing interface.

The delegating relay provides the operator's Layer-3 edge towards the client and is responsible for routing traffic to and from clients connected to the client network using addresses from the delegated prefixes.

2.3. Requirements Language

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. This document uses these keywords not strictly for the purpose of interoperability, but rather for the purpose of establishing industry-common baseline functionality. As such, the document points to several other specifications (preferably in RFC or stable form) to provide additional guidance to implementers regarding any protocol implementation required to produce a DHCP relaying router that functions successfully with prefix delegation.

3. Problems Observed with Existing Delegating Relays Implementations

The following sections of the document describe problems that have

been observed with delegating relay implementations in commercially available devices.

[3.1.](#) DHCP Messages not being Forwarded by the Delegating relay

Delegating relay implementations have been observed not to forward messages between the client and server. This generally occurs if a client sends a message which is unexpected by the delegating relay. For example, the delegating router already has an active PD lease entry for an existing client on a port. A new client is connected to this port and sends a solicit message. The delegating relay then drops the solicit messages until it receives either a DHCP release message from the original client, or the existing lease times out. This causes a particular problem when a client device needs to be replaced due to a failure.

In addition to dropping messages, in some cases the delegating relay will generate error messages and send them to the client, e.g. 'NoBinding' messages being sent in the event that the delegating relay does not have an active delegated prefix lease.

[3.2.](#) Delegating Relay Loss of State on Reboot

For proper routing of client's traffic, the delegating relay requires a corresponding routing table entry for each active prefix delegated to a connected client. A delegating router which does not store this

state persistently across reboots will not be able to forward traffic to client's delegated leases until the state is re-established through new DHCP messages.

[3.3.](#) Multiple Simultaneous Delegated Prefixes for a Single DUID on a Single Client

[RFC8415] allows for a client to include more than one instance of OPTION_IA_PD in messages in order to request multiple prefix delegations by the server. If configured for this, the server supplies one instance of OPTION_IAPREFIX for each received instance of OPTION_IA_PD, each containing information for a different delegated prefix.

In some delegating relay implementations, only a single delegated

prefix per-DUID is supported. In those cases only one IPv6 route for only one of the delegated router is installed; meaning that other prefixes delegated to a client are unreachable.

[3.4.](#) Dropping Messages from Devices with Duplicate MAC addresses and DUIDs

It is an unfortunate operational reality that client devices with duplicate MAC addresses and/or DUIDs exist and have been deployed. In this situation, the operational costs of locating and swapping out such devices are prohibitive.

Delegating relays have been observed to restrict forwarding client messages originating from one client DUID to a single interface. In this case if the same client DUID appears from a second client on another interface while there is already an active lease, messages originating from the second client are dropped causing the second client to be unable to obtain a prefix delegation.

[4.](#) Requirements for Delegating Relays

To resolve the problems described in [Section 3](#) the following section of the document describes a set of functional requirements for the delegating relay.

[4.1.](#) General Requirements

G-1: The delegating router MUST forward messages bidirectionally between the client and server without changing the contents of the message.

G-2: As described in [Section 16 of \[RFC8415\]](#), in the event that a received message contains a DHCPv6 option which the relay does not implement, the message MUST be forwarded.

G-3: The relay MUST allow for multiple prefixes to be delegated for the same client IA_PD. These delegations may have different lifetimes.

- G-4: The relay MUST allow for multiple prefixes with separate IA_PDs to be delegated to a single client connected to a single interface, identified by its DHCPv6 Client Identifier (DUID).
- G-5: The relay MUST allow the same client identifier (DUID) to have active delegated prefix leases on more than one interface simultaneously. This is to allow client devices with duplicate DUIDs to function on separate broadcast domains.
- G-6: The maximum number of simultaneous prefixes delegated to a single client MUST be configurable.
- G-7: The relay MUST implement a mechanism to limit the maximum number of active prefix delegations on a single port for all client identifiers and IA_PDs. This value SHOULD be configurable.
- G-8: The delegating relay MUST synchronize the lifetimes of active prefix delegation leases with server.

4.2. Routing Requirements

- R-1: The relay MUST maintain a local routing table that is dynamically updated with prefixes and the associated next-hops as they are delegated to clients. When a delegated prefix is released or expires, the associated route MUST be removed from the relay's routing table.
- R-2: The relay MUST provide a mechanism to dynamically update access control lists permitting ingress traffic sourced from clients' delegated prefixes. This is to implement anti-spoofing as described in [[BCP38](#)].
- R-3: The relay MAY provide a mechanism to dynamically advertise delegated prefixes into an routing protocol as they are learnt. When a delegated prefix is released or expires, the delegated route MUST be withdrawn from the routing protocol.

is out of the scope of this document.

4.3. Service Continuity Requirements

- S-1: In the event that the relay is restarted, active client prefix delegations will be lost. This may result in clients becoming unreachable. In order to mitigate this problem, it is RECOMMENDED that the relay implements either of the following:

The relay MAY implement DHCPv6 bulk lease query as defined in [[RFC5460](#)].

The relay SHOULD store active prefix delegations in persistent storage so they can be re-read after the reboot.

- S-2: If a client's next-hop link-local address becomes unreachable (e.g., due to a link-down event on the relevant physical interface), routes for the client's delegated prefixes MUST be retained by the delegating relay unless they are released or removed due to expiring DHCP timers. This is to re-establish routing for the delegated prefix if the client next-hop becomes reachable without the relay needing to be re-learned.
- S-3: The relay MAY implement DHCPv6 active lease query as defined in [[RFC7653](#)] to keep the local lease database in sync with the DHCPv6 server.

4.4. Operational Requirements

- O-1: The relay SHOULD implement an interface allowing the operator to view the active delegated prefixes. This SHOULD provide information about the delegated lease and client details such as client identifier, next-hop address, connected interface, and remaining lifetimes.
- O-2: The relay SHOULD provide a method for the operator to clear active bindings for an individual lease, client or all bindings on a port.
- O-3: To facilitate troubleshooting of operational problems between the delegating relay and other elements, it is RECOMMENDED

that the delegating relay's system time is synchronised with the network.

5. Acknowledgements

The authors of this document would like to thank Bernie Volz for his valuable comments.

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

If the delegating relay implements [BCP38] filtering, then the filtering rules will need to be dynamically updated as delegated prefixes are leased.

[RFC8213] describes a method for securing traffic between the relay agent and server by sending DHCP messages over an IPsec tunnel. In this case the IPsec tunnel is functionally the server-facing interface and DHCPv6 message snooping can be carried out as described. It is RECOMMENDED that this is implemented by the delegating relay.

8. References

8.1. Normative References

- [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>>.
- [RFC5460] Stapp, M., "DHCPv6 Bulk Leasequery", [RFC 5460](#), DOI 10.17487/RFC5460, February 2009, <<https://www.rfc-editor.org/info/rfc5460>>.
- [RFC7653] Raghuvanshi, D., Kinnear, K., and D. Kukrety, "DHCPv6 Active Leasequery", [RFC 7653](#), DOI 10.17487/RFC7653, October 2015, <<https://www.rfc-editor.org/info/rfc7653>>.
- [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>>.

Internet-Draft

DHCPv6 PD relay

December 2019

- [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>>.

8.2. Informative References

- [BCP38] IETF, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing" <https://tools.ietf.org/html/bcp38>", [RFC 2827](#), [BCP 38](#).
- [RFC7283] Cui, Y., Sun, Q., and T. Lemon, "Handling Unknown DHCPv6 Messages", [RFC 7283](#), DOI 10.17487/RFC7283, July 2014, <<https://www.rfc-editor.org/info/rfc7283>>.
- [RFC8213] Volz, B. and Y. Pal, "Security of Messages Exchanged between Servers and Relay Agents", [RFC 8213](#), DOI 10.17487/RFC8213, August 2017, <<https://www.rfc-editor.org/info/rfc8213>>.

Authors' Addresses

Ian Farrer
Deutsche Telekom AG
Landgrabenweg 151
Bonn, NRW 53227
DE

Email: ian.farrer@telekom.de

Naveen Kottapalli
Benu Networks
300 Concord Road
Billerica, MA 01821
US

Email: naveen.sarma@gmail.com

Martin Hunek
Technical University of Liberec
Studentska 1402/2
Liberec, L 46017
CZ

Email: martin.hunek@tul.cz

Farrer, et al.

Expires June 19, 2020

[Page 10]

Internet-Draft

DHCPv6 PD relay

December 2019

Richard Patterson

Email: richard@helix.net.nz

