Network Working Group
Internet-Draft
Updates: 3315,3633 (if approved)
Intended status: Standards Track
Expires: January 1, 2015

O. Troan
B. Volz
Cisco Systems, Inc.
M. Siodelski
ISC

June 30, 2014

Issues with multiple stateful DHCPv6 options
draft-ietf-dhc-dhcpv6-stateful-issues-06.txt

Abstract

Dynamic Host Configuration Protocol for IPv6 (DHCPv6) was not written with the expectation that additional stateful DHCPv6 options would be developed. IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6 shoe-horned the new options for Prefix Delegation into DHCPv6. Implementation experience of the CPE model described in RFC 7084 has shown multiple issues with the DHCPv6 protocol in supporting multiple stateful options. This document updates RFC 3315 and RFC 3633.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 1, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

DHCPv6 [RFC3315] was not written with the expectation that additional stateful DHCPv6 options would be developed. DHCPv6 Prefix Delegation [RFC3633] shoe-horned the new options for Prefix Delegation into DHCPv6. Implementation experience of the CPE model described in [RFC7084] has shown multiple issues with the DHCPv6 protocol in supporting multiple stateful option types, in particular IA_NA and IA_PD.

This document describes a number of problems encountered with multiple IA option types and recommended changes to the DHCPv6 protocol specifications.

The intention of this work is to modify the DHCP protocol specification to support multiple IA option types within a single DHCP session. This problem can also be solved by implementing a separate DHCP session (separate client state machine) per IA option type. This latter approach has a number of issues: additional DHCP protocol traffic, 'collisions' between stateless options also included with the IA options, divergence in that each IA option type specification specifies its 'own' version of the DHCP protocol.

Note that while IA_TA options may be included with other IA option type requests, these generally are not renewed (there are no T1/T2 times) and have a separate life cycle from IA_NA and IA_PD option types. IA_TA also has limited value when DHCPv6 is used for address assignment, as the privacy issues identified for IPv6 stateless address assignment ([RFC4941]) do not apply to DHCPv6 assignments.

The changes described in this document will be incorporated in a new revision of the DHCPv6 protocol specification [RFC3315].

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Terminology

In addition to the terminology defined in [RFC3315], [RFC3633], and [RFC7227], the following terminology is used in this document:

Resource (allocable resource): A value (or a collection of values) dynamically assigned to the client by the server and being carried in the stateful options. An example of the resources are: IPv6 address or an IPv6 prefix. Information about the resources is transported in stateful options such as IA_NA, for addresses, and IA_PD, for prefix delegations. In the future, other types of resources and stateful options may be defined.

Identity association (IA): A collection of stateful options assigned to a client. Each IA has an associated IAID. A client may have more than one IA assigned to it; for example, one for each of its interfaces. Each IA holds one type of IA option; for example, an identity association for temporary addresses (IA_TA) holds temporary addresses (see "identity association for temporary addresses"). Throughout this document, "IA" is used to refer to an identity association without identifying the type of stateful option in the IA.

IA option types: This is used to generally mean an IA_NA and/or IA_PD and may also include IA_TA, as well as any future IA options.

Stateful options: Options that require dynamic binding state per client on the server.

4. Handling of multiple IA options types

DHCPv6 was written with the assumption that the only stateful options were for assigning addresses. DHCPv6 PD describes how to extend the DHCPv6 protocol to handle prefix delegation, but [RFC3633] did not consider how DHCP address assignment and prefix delegation could co-exist.

If a client requests multiple IA option types, but the server is configured to only offer a subset of them, the client could react in several ways. Reset the state machine and continue to send Solicit messages, create separate DHCP sessions for each IA option type and continue to Solicit for the unfulfilled IA options, or it could continue with the single session, and include the unfulfilled IA options on subsequent messages to the server.

Proposed solution: the client should keep a single session with the server and include the missing options on subsequent messages (Request, Renew, and Rebind) to the server.

4.1. Placement of Status Codes

In Reply messages IA specific status codes (i.e., NoAddrsAvail, NotOnlink, NoBinding, NoPrefixAvail) are encapsulated in the IA option. In Advertise messages the Status Code option with the NoAddrsAvail code is in the top-level. That makes sense when the failure case is fatal. With the introduction of multiple IA option types, there might be a case where a server is not willing to offer addresses, but might be willing to offer other stateful option types.

While a Status Code option is implicitly bound to a specific type of IA, e.g. NoPrefixAvail is only applicable to IA_PD and NoAddrsAvail is only applicable to IA_NA/IA_TA, it may be problematic to make this assumption for all status codes. Ideally the Status Code option should be encapsulated in the IA option for all DHCP messages. This makes Status Code option placement for Advertise messages identical to Reply messages.

However, how a server formats the Advertise message when addresses are not available has been a point of some confusion and implementations seem to vary.

Therefore, the Proposed solution is:

Clients MUST be prepared to handle each of the following Advertise messages formats when there are no addresses available (even when no IA_PD was in the Solicit):

1. Advertise containing just a top-level Status Code option (of NoAddrsAvail) and no IA_NAs/IA_TAs.

2. Advertise containing the IA_NAs and/or IA_TAs with encapsulated Status Code option (of NoAddrsAvail) and no top-level Status Code option.

3. Advertise containing a top-level Status Code option (of NoAddrsAvail) and IA_NAs and/or IA_TAs with a Status Code option (of NoAddrsAvail).

Servers MUST use the Status Code option (of NoAddrsAvail) encapsulated in an IA_NA/IA_TA options and not a top-level Status Code option (of NoAddrsAvail) when no addresses will be assigned (2 in the above list). This means that the Advertise response matches the Reply response with respect to the handling of the NoAddrsAvail status.

Replace the following paragraph in RFC 3315, section 17.2.2:

> If the server will not assign any addresses to any IAs in a subsequent Request from the client, the server MUST send an Advertise message to the client that includes only a Status Code option with code NoAddrsAvail and a status message for the user, a Server Identifier option with the server's DUID, and a Client Identifier option with the client's DUID.

With:

> If the server will not assign any addresses to an IA in a subsequent Request from the client, the server MUST include the IA in the Advertise message with no addresses in the IA and a Status Code option encapsulated in the IA containing status code NoAddrsAvail.

4.2. Advertise Message

[RFC3315] specifies that a client must ignore an Advertise message if a server will not assign any addresses to a client. A client requesting both IA_NA and IA_PD, with a server that only offers one of them, is not supported in the current protocol specification.

Proposed solution: a client SHOULD accept Advertise messages, even when not all IA option types are being offered. And, in this case, the client SHOULD include the not offered IA option types in its Request. A client SHOULD only ignore an Advertise message when no IA option includes any offered addresses or delegated prefixes (or any future allocable resource). Note that ignored messages MUST still be processed for SOL_MAX_RT and INF_MAX_RT options as specified in [RFC7083].

Replace Section 17.1.3 of RFC 3315: (existing errata)

> The client MUST ignore any Advertise message that includes a Status Code option containing the value NoAddrsAvail, with the exception that the client MAY display the associated status message(s) to the user.

With (this includes the changes made by [RFC7083]):

> The client MUST ignore any Advertise message that contains no addresses (IAADDR options encapsulated in IA_NA or IA_TA options) and no delegated prefixes (IAPREFIX options encapsulated in IA_PD options, see RFC 3633) with the exception that the client MUST process an included SOL_MAX_RT option (RFC 7083), MUST process an included INF_MAX_RT option (RFC 7083), and MAY display any associated status message(s) to the user.

And, replace:

- The client MAY choose a less-preferred server if that server has a better set of advertised parameters, such as the available addresses advertised in IAs.

With:

- The client MAY choose a less-preferred server if that server has a better set of advertised parameters, such as the available options advertised in IAs.

It is important to note that the receipt of an Advertise message without any addresses and delegated prefixes does not imply that the client should restart the Solicit retransmissions timers. Doing so would lead to a Solicit/Advertise storm.

4.3. T1/T2 Timers

The T1 and T2 timers determine when the client will contact the server to extend lifetimes of information received in an IA. How should a client handle the case where multiple IA options have different T1 and T2 timers?

In a multiple IA option type model, the T1/T2 timers are protocol timers, that should be independent of the IA options themselves. If we were to redo the DHCP protocol from scratch the T1/T2 timers should be carried in a separate DHCP option.

Proposed solution: The server SHOULD set the T1/T2 timers in all IA options in Reply and Advertise messages to the same value. To deal with the case where servers have not yet been updated to do that, clients MUST use the shortest (explicit or implicit) T1/T2 timer (larger than 0) in any IA options in the Reply. Longer T1/T2 timers are ignored.

4.4. Renew Message

The Renew message, as described in [RFC3315], allows a client to only renew bindings assigned via a Request message.

In a multiple IA option type model, the Renew does not support the ability for the client to renew one IA option type while requesting bindings for other IA option types that were not available when the client sent the Request.

Proposed solution: The client should continue with the IA options received, while continuing to include the other IA options in

subsequent messages to the server. The client and server processing need to be modified. Note that this change makes the server's IA processing of Renew similar to the Request processing.

The first two subsections contain the required updates to [RFC3315] and [RFC3633] to accommodate this behavior on the client and the server. The remaining two subsections propose a "unified" text to be included in the [I-D.dhcwg-dhc-rfc3315bis], describing the client's and server's behavior for renewing both addresses and prefixes.

4.4.1. Updates to RFC 3315

Replace Section 18.1.3 of RFC 3315:

> At time T1 for an IA, the client initiates a Renew/Reply message exchange to extend the lifetimes on any addresses in the IA. The client includes an IA option with all addresses currently assigned to the IA in its Renew message.

With:

> At time T1 for an IA, the client initiates a Renew/Reply message exchange to extend the lifetimes on any addresses in the IA. The client includes an IA option with all addresses currently assigned to the IA in its Renew message. The client also includes an IA option for each binding it desires but has been unable to obtain.

Replace Section 18.2.3 of RFC 3315:

> If the server cannot find a client entry for the IA the server returns the IA containing no addresses with a Status Code option set to NoBinding in the Reply message.

With:

> If the server cannot find a client entry for the IA the server creates the bindings for that client according to the server's policy and configuration information and returns the IAs and other information requested by the client.

Note that clients that communicate with servers that do not support this updated Renew processing will receive the NoBinding status for the IA which had no bindings. The client MUST continue to process the other IAs in the Reply. The client MAY attempt a Solicit/Advertise/Request/Reply sequence periodically to obtain bindings for these IAs. However, it MUST limit the frequency at which it does this to no more often than the renewal frequency.

4.4.2. Updates to RFC 3633

Replace Section 12.2 of RFC 3633:

```
   Renew message:   If the delegating router cannot find a binding
                    for the requesting router's IA_PD the delegating
                    router returns the IA_PD containing no prefixes
                    with a Status Code option set to NoBinding in the
                    Reply message.
```

With:

```
   Renew message:   If the delegating router cannot find a binding
                    for the requesting router's IA_PD, the delegating
                    router creates the bindings for that client
                    according to the server's policy and
                    configuration information and returns the IAs and
                    other information requested by the client.
```

4.4.3. Creation and Transmission of Renew Messages (unified text)

To extend the valid and preferred lifetimes for the resources associated with IAs, the client sends a Renew message to the server from which the client obtained the resources.

The server controls the time at which the client contacts the server to extend the lifetimes on client's bindings through the T1 and T2 parameters assigned to IAs. The server SHOULD assign the same T1 and T2 value to each binding assigned to the client. In this case the client uses the common T1 or T2 value returned in the IAs to determine the time when it should send Renew or Rebind message to the server. If the server sends different T1/T2 values for different IAs, the client uses the shortest T1/T2 value (larger than 0). T1/T2 values in other IA options are ignored.

If T1 or T2 is set to 0 by the server for all IA_NA and IA_PD options, or there are no T1 or T2 times (for an IA_TA), the client may send Renew or Rebind message, respectively, at the client's discretion.

At time T1, the client that initiates a Renew/Reply message exchange, includes IA options for all bindings for which it desires to extend lifetimes in its Renew message. For each IA being included, the client includes all resources currently associated with the IA.

The client also includes IA option for each binding it desires but has been unable to obtain. For example: a client which has non-temporary addresses assigned but hasn't been delegated a prefix, may include an IA_PD option (in addition to the IA_NA option) in the Renew message to request prefix delegation.

The client constructing a Renew message SHOULD NOT include resources in IA options that the client does not have. If the client included a resource it does not have and the server does not allocate this resource for the client, the server will return the resource to the client in the IA with the lifetimes set to 0. This is a signal to the client to not use this resource. The server MAY allocate a different resource to the client and send it in the same IA.

The client sets "msg-type" field to RENEW. The client generates a transaction ID and inserts this value in the "transaction-id" field.

The client places the identifier of the destination server in a Server Identifier option.

The client MUST include a Client Identifier option to identify itself to the server. The client adds any appropriate options, including one or more IA options.

The client MUST include an Option Request option to indicate the options the client is interested in receiving. The client MAY include options with data values as hints to the server about parameter values the client would like to have returned.

The client transmits the message according to section 14, using the following parameters:

IRT	REN_TIMEOUT
MRT	REN_MAX_RT
MRC	0
MRD	Remaining time until T2

If the server finds that any resource sent by the client is not appropriate, according to the server's configuration information, the server sends back the IA with the corresponding IA Address (for inappropriate address), IA Prefix option (for inappropriate prefix) or other option appropriate for the type of the resource, with lifetimes set to 0. The client which receives the option with lifetimes set to 0 MUST NOT use the corresponding resource.

The message exchange is terminated when time T2 is reached, at which time the client begins a Rebind message exchange.

4.4.4. Receipt of Renew Messages (unified text)

When the server receives a Renew message via unicast from a client to which the server has not sent a unicast option, the server discards the Renew message and responds with a Reply message containing a Status Code option with the value UseMulticast, a Server Identifier option containing the server's DUID, the Client Identifier option from the client message, and no other options.

When the server receives a Renew message that contains an IA option from a client, it locates the client's binding and verifies that the information in the IA from the client matches the information stored for the client.

If the server cannot find a client entry for the IA, the server creates the bindings for that client according to the server's policy and configuration information and returns the IAs and other information requested by the client. For each IA for which the server will not create a binding the server returns an IA option containing a Status Code option set to NoBinding in the Reply message.

If the server finds that any resource sent by the client is not appropriate, according to the server's configuration information, the server sends back the IA with the corresponding IA Address (for invalid address), IA Prefix option (for invalid prefix) or any other option appropriate for the type of the resource, with lifetimes set to 0.

The server constructs a Reply message by setting the "msg-type" field to REPLY, and copying the transaction ID from the Renew message into the transaction-id field.

The server MUST include a Server Identifier option containing the server's DUID and the Client Identifier option from the Renew message in the Reply message.

The server includes other options containing configuration information to be returned to the client as described in [RFC3315].

4.5. Rebind Message

In the Section 4.4 it has been proposed that the client includes IA options in a Renew message for the bindings it desires but has been

unable to obtain by sending a Request message, apart from the IA options for the existing bindings.

At time T2 (being a shortest, greater than 0, time across all client's IAs), the client stops sending Renew messages to the server and initiates the Rebind/Reply message exchange with any available server. In this case, it should be possible to continue trying to obtain new bindings using the Rebind message if the client failed to get the response from the server to the Renew message.

The Rebind message, as described in [RFC3315] does not explicitly specify what a server should do when an IA option which contains no addresses is present.

Proposed solution: The client should continue with the IA options received and it MAY include additional IA options to request creation of additional bindings.

The first two subsections contain the required updates to [RFC3315] and [RFC3633] to accommodate this behavior on the client and the server. The remaining two subsections propose a "unified" text to be included in the [I-D.dhcwg-dhc-rfc3315bis], describing the client's and server's behavior with respect to processing different IA option types in a single Rebind message.

4.5.1. Updates to RFC 3315

Replace Section 18.1.4 of RFC 3315:

> At time T2 for an IA (which will only be reached if the server to which the Renew message was sent at time T1 has not responded), the client initiates a Rebind/Reply message exchange with any available server. The client includes an IA option with all addresses currently assigned to the IA in its Rebind message.

With:

> At time T2 for an IA (which will only be reached if the server to which the Renew message was sent at time T1 has not responded), the client initiates a Rebind/Reply message exchange with any available server. The client includes an IA option with all addresses currently assigned to the IA in its Rebind message. The client also includes an IA option (without the IA Address option) for each binding it desires but has been unable to obtain.
>
> The client constructing a Rebind message SHOULD NOT include addresses in IA options that the client does not have. If the client included an address it does not have and the server does

not allocate this address for the client, the server will return the IA Address option containing address included by the client in the IA with lifetimes set to 0. This is an indication to the client to not use this address. The server MAY allocate a different address to the client and send it in the same IA.

Replace Section 18.2.4 of RFC 3315 with the following text to clarify how the server should handle all of the possible conditions:

When the server receives a Rebind message that contains an IA option from a client, it locates the client's binding and verifies that the information in the IA from the client matches the information stored for that client.

If the server finds the addresses in the IA for the client and the server determines that the addresses in the IA are appropriate for the link to which the client's interface is attached according to the server's explicit configuration information, the server SHOULD send back the IA to the client with new lifetimes and T1/T2 times.

If the server cannot find a client entry for the IA and the server determines that the addresses in the IA are not appropriate for the link to which the client's interface is attached according to the server's explicit configuration information, the server MAY send a Reply message to the client containing the client's IA, with the lifetimes for the addresses in the IA set to zero. This Reply constitutes an explicit notification to the client that the addresses in the IA are no longer valid. In this situation, if the server does not send a Reply message it silently discards the Rebind message.

If the server cannot find a client entry for the IA and the IA is empty (i.e. contains no addresses), the server MAY assign the addresses to this IA and send a Reply message to the client with this IA containing allocated addresses with lifetimes and T1/T2 times. In the case when the client included addresses in the IA, included addresses are appropriate for the link to which the client's interface is attached according to the server's explicit configuration information and they are not in use, the server MAY allocate these addresses to the client. If the server does not allocate addresses to the client, the server returns the IA containing only Status Code option set to NoBinding in the Reply message.

When the server creates new bindings for the IA it is possible that other servers also create bindings as a result of receiving the same Rebind message. This is the same issue as in the Discussion under the Rapid Commit option, see section 22.14. Therefore, the

server SHOULD only create new bindings during processing of a Rebind message if the server is configured to respond with a Reply message to a Solicit message containing the Rapid Commit option.

The server constructs a Reply message by setting the "msg-type" field to REPLY, and copying the transaction ID from the Rebind message into the transaction-id field.

The server MUST include a Server Identifier option containing the server's DUID and the Client Identifier option from the Rebind message in the Reply message.

The server includes other options containing configuration information to be returned to the client as described in section 18.2.

4.5.2. Updates to RFC 3633

Replace Section 12.2 of RFC 3633:

```
   Rebind message:  If the delegating router cannot find a binding
                    for the requesting router's IA_PD and the
                    delegating router determines that the prefixes in
                    the IA_PD are not appropriate for the link to
                    which the requesting router's interface is
                    attached according to the delegating routers
                    explicit configuration, the delegating router MAY
                    send a Reply message to the requesting router
                    containing the IA_PD with the lifetimes of the
                    prefixes in the IA_PD set to zero.  This Reply
                    constitutes an explicit notification to the
                    requesting router that the prefixes in the IA_PD
                    are no longer valid.  If the delegating router is
                    unable to determine if the prefix is not
                    appropriate for the link, the Rebind message is
                    discarded.
```

with:

```
   Rebind message:  If the delegating router cannot find a binding
                    for the requesting router's IA_PD and the
                    delegating router determines that the prefixes in
                    the IA_PD are not appropriate for the link to
                    which the requesting router's interface is
```

attached according to the delegating routers explicit configuration, the delegating router MAY send a Reply message to the requesting router containing the IA_PD with the lifetimes of the prefixes in the IA_PD set to zero. This Reply constitutes an explicit notification to the requesting router that the prefixes in the IA_PD are no longer valid. If the delegating router is unable to determine if the prefix is not appropriate for the link, the Rebind message is discarded. If the IA_PD contains no prefixes or the prefixes are appropriate for the link to which the requesting router's interface is attached according to the delegating router's explicit configuration information and if prefixes are not in use, the delegating router MAY assign prefixes to this IA_PD.

4.5.3. Creation and Transmission of Rebind Messages (unified text)

At time T2 for an IA (which will only be reached if the server to which the Renew message was sent at time T1 has not responded), the client initiates a Rebind/Reply message exchange with any available server. For each IA being included, the client stores all resources currently associated with the IA.

The client also includes IA option for each binding it desires but has been unable to obtain. For example: a client which has non-temporary addresses assigned but has not been delegated a prefix, may include an IA_PD option (in addition to the IA_NA option) in the Rebind message to request the prefix delegation.

The client constructing a Rebind message SHOULD NOT include resources in IA options that the client does not have. If the client included a resource it does not have and the server does not allocate this resource for the client, the server will return the appropriate option containing the resource (e.g. IA Address, IA Prefix) with the lifetimes set to 0. This is an indication to the client to not use this resource. The server MAY allocate a different resource to the client and send it in the same IA.

The client sets the "msg-type" field to REBIND. The client generates a transaction ID and inserts this value in the "transaction-id" field.

The client MUST include a Client Identifier option to identify itself to the server. The client adds any appropriate options, including one or more IA options. The client MUST include the list of

resources the client currently has associated with the IAs in the Rebind message.

The client MUST include an Option Request option to indicate the options that client is interested in receiving. The client MAY include options with data values as hints to the server about parameter values the client would like to have returned.

The client transmits the message according to Section 14, using the following parameters:

IRT	REB_TIMEOUT
MRT	REB_MAX_RT
MRC	0
MRD	Remaining time until valid lifetimes of all addresses or prefixes in the IA have expired

The message exchange is terminated when the valid lifetimes of all the resources assigned to the IA expire, at which time the client has several alternative actions to choose from; for example:

- The client may choose to use a Solicit message to locate a new DHCP server and send a Request for the expired IA to the new server.

- The client may have other resources in other IAs, so the client may choose to discard the expired IA and use the other IAs.

4.5.4. Receipt of Rebind Messages (unified text)

When the server receives a Rebind message that contains an IA option from a client, it locates the client's binding and verifies that the information in the IA from the client matches the information stored for that client.

If the server finds the resource in the IA for the client and the server determines that the resources in the IA are appropriate for the link to which the client's interface is attached according to the server's explicit configuration information, the server SHOULD send back the IA to the client with new lifetimes and T1/T2 times.

If the server cannot find a client entry for the IA and the server determines that the resources in the IA are not appropriate for the link to which the client's interface is attached according to the server's explicit configuration information, the server MAY send a

Reply message to the client containing the client's IA, with the lifetimes for the resources in the IA set to zero. This Reply constitutes an explicit notification to the client that the resources in the IA are no longer valid. In this situation, if the server does not send a Reply message it silently discards the Rebind message.

If the server cannot find a client entry for the IA and the IA is empty (i.e. contains no resources), the server MAY assign the resources to this IA and send a Reply message to the client with this IA containing allocated resources with lifetimes and T1/T2 times. In the case when the client included resources in the IA, included resources are appropriate for the link to which the client's interface is attached according to the server's explicit configuration information and they are not in use, the server MAY allocate these resources to the client. If the server does not allocate resources to the client, the server returns the IA containing only Status Code option set to NoBinding in the Reply message.

When the server creates new bindings for the IA it is possible that other servers also create bindings as a result of receiving the same Rebind message. This is the same issue as in the Discussion under the Rapid Commit option, see section 22.14 of [RFC3315]. Therefore, the server SHOULD only create new bindings during processing of a Rebind message if the server is configured to respond with a Reply message to a Solicit message containing the Rapid Commit option.

The server constructs a Reply message by setting the "msg-type" field to REPLY, and copying the transaction ID from the Rebind message into the transaction-id field.

The server MUST include a Server Identifier option containing the server's DUID and the Client Identifier option from the Rebind message in the Reply message.

The server includes other options containing configuration information to be returned to the client as described in section 18.2.

4.6. Confirm Message

The Confirm message, as described in [RFC3315], is specific to address assignment. It allows a server without a binding to reply to the message, under the assumption that the server only needs knowledge about the prefix(es) on the link, to inform the client that the address is likely valid or not. This message is sent when e.g. the client has moved and needs to validate its addresses. Not all bindings can be validated by servers and the Confirm message provides

for this by specifying that a server that is unable to determine the on-link status MUST NOT send a Reply.

Note: Confirm has a specific meaning and does not overload Renew/ Rebind. It also is lower processing cost as the server does NOT need to extend lease times or otherwise send back other configuration options.

The Confirm message is used by the client to verify that it has not moved to a different link. For IAs with addresses, the mechanism used to verify if a client has moved or not, is by matching the link's on-link prefix(es) (typically a /64) against the prefix-length first bits of the addresses provided by the client in the IA_NA or IA_TA IA-types. As a consequence Confirm can only be used when the client has an IA with address(es) (IA_NA or IA_TA).

A client MUST have a binding including an IA with addresses to use the Confirm message. A client with IAs with addresses as well as other IA-types MAY, depending on the IA-type, use the Confirm message to detect if the client has moved to a different link. A client that does not have a binding with an IA with addresses MUST use the Rebind message instead.

IA_PD requires verification that the server has the binding for the IAs. In that case a client MUST use the Rebind message in place of the Confirm message and it MUST include all of its bindings, even address IAs.

4.7. Decline Should Not Necessarily Trigger a Release

Some clients will send a Release message for other bindings they may have received after they determine a conflict and have correctly sent a Decline message for the conflicting address(es).

It is recommended that a client SHOULD NOT send a Release message for other bindings it may have received just because it sent a Decline message. The client should retain the non-conflicting bindings.

4.8. Multiple Provisioning Domains

This document has assumed that all DHCP servers on a network are in a single provisioning domain and thus should be "equal" in the service that they offer. This was also assumed by [RFC3315] and [RFC3633].

One could envision a network where the DHCP servers are in multiple provisioning domains, and it may be desirable to have the DHCP client obtain different IA types from different provisioning domains. How a client detects the multiple provisioning domains and how it would

interact with the multiple servers in these different domains is outside the scope of this document.

5. IANA Considerations

This specification does not require any IANA actions.

6. Security Considerations

There are no new security considerations pertaining to this document.

7. Acknowledgements

Thanks to many people that contributed to identify the issues in this document, including Ralph Droms, John, Brzozowski, Ted Lemon, Hemant Singh, Wes Beebee, Gaurau Halwasia, Bud Millword, Tim Winters, and Rob Shakir.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[RFC7083] Droms, R., "Modification to Default Values of SOL_MAX_RT and INF_MAX_RT", RFC 7083, November 2013.

8.2. Informative References

[I-D.dhcwg-dhc-rfc3315bis] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., and T. Lemon, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) bis", draft-dhcwg-dhc-rfc3315bis-01 (work in progress), May 2014.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, September 2007.

[RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, November 2013.

[RFC7227] Hankins, D., Mrugalski, T., Siodelski, M., Jiang, S., and S. Krishnan, "Guidelines for Creating New DHCPv6 Options", BCP 187, RFC 7227, May 2014.

Authors' Addresses

Ole Troan
Cisco Systems, Inc.
Philip Pedersens vei 20
N-1324 Lysaker
Norway

Email: ot@cisco.com

Bernie Volz
Cisco Systems, Inc.
1414 Massachusetts Ave
Boxborough, MA 01719
USA

Email: volz@cisco.com

Marcin Siodelski
ISC
950 Charter Street
Redwood City, CA 94063
USA

Email: msiodelski@gmail.com