

DHC Working Group  
Internet Draft  
Intended status: Standards Track  
Expires: July 2013

Rajiv Asati  
Cisco Systems

Dan Wing  
Cisco Systems

December 31, 2012

Tracking of Static/Autoconfigured IPv6 addresses  
draft-asati-dhc-ipv6-autoconfig-address-tracking-00.txt

## Abstract

Network operators commonly use Dynamic Host Configuration Protocol (DHCP) to assign IP addresses to the hosts, and track them. However, the tracking capability is lost when stateless autoconfiguration or manual methods are used to assign IPv6 addresses.

This document proposes a Dynamic Host Configuration Protocol for IPv6 (DHCPv6) based mechanism that the last hop router can use to convey the hosts' IPv6 addresses for the tracking and logging purposes, without requiring any changes to the hosts.

## Status of this Memo

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

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

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

This Internet-Draft will expire on July 2, 2013.

## Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

## Table of Contents

1. Introduction.....	2
2. Terminology.....	3
3. Scope.....	3
4. Protocol Operation.....	4
4.1. Host Behavior.....	4
4.2. Router Behavior.....	4
4.3. DHCP Server Behavior.....	5
5. DHCP Messages and Options.....	5
5.1. DHCPv6 Messages.....	5
5.1.1. RECORD-CLIENTBINDING message.....	6
5.1.2. RECORD-CLIENTBINDING-ACK message.....	6
5.2. DHCPv6 Options.....	7
5.2.1. CLIENT_BINDING Option.....	7
6. Security Considerations.....	9
7. IANA Considerations.....	9
8. References.....	9
8.1. Normative References.....	9
8.2. Informative References.....	9
9. Acknowledgments.....	10

## 1. Introduction

As network operators leverage SLAAC (Stateless Address auto configuration) [RFC4862] or static methods to assign the IPv6 address to the hosts (e.g. subscribers/users), they can no longer track or figure out which IPv6 address is used by which host. This potentially impacts the operator's operational aspects (e.g.

subscriber management) and forces the operators to postpone the IPv6 roll-out until all the hosts can support DHCPv6 [RFC3315].

Note that the operators rely on DHCP server's lease assignments to keep track of IP address assigned to hosts by creating the mapping of MAC address and IP address(es) pertaining to each host. This assumes DHCP support on the hosts, of course.

This document describes a Dynamic Host Configuration Protocol version 6 (DHCPv6) mechanism that the first hop router (or switch) can use to convey the IPv6 addresses obtained by the hosts using stateless address autoconfiguration (SLAAC) [RFC4862] or static methods, to the DHCPv6 server for tracking and/or logging purposes.

This document does NOT propose any changes to the hosts.

## 2. Terminology

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

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 RFC-2119 [RFC2119].

## 3. Scope

This document caters to the following sample network topology that could be mapped to a residential broadband network topology or enterprise network topology or data center.

```
Host11-----Node11-----Network-----Router2-----DHCP Server(s)
Host12-----//              |
Host13=====Node12-----|
```

Figure 1 Sample Network

The above topology could be illustrated a bit differently in figure 2 -

```

Host11----CE11-----PE11---Network----Router2-----DHCP Server(s)
Host12-----//                                     |
Host13=====PE12-----|

```

Figure 2 Network

In case of residential broadband homes, CE11 could be a residential gateway (RGW) or CPE router, whereas PE11 could be a Broadband Network Gateway (BNG) or Cable Modem Termination System (CMTS) or 3GPP Packet Data Network (PDN) Gateway aka PGW.

In case of enterprise network, CE11 could be a Customer Edge Router or Switch connecting to the hosts.

CE11 could also be dubbed as the requesting router.

#### 4. Protocol Operation

This section describes the protocol operation in terms of changes needed on Host, router and DHCP server.

##### 4.1. Host Behavior

This document assumes hosts' support for [RFC4861], and does not require any changes to the host behavior.

Per [RFC4861], a host MUST perform Neighbor Discovery and Router Discovery as soon as IPv6 is enabled on any of its interfaces. As part of Neighbor Discovery, a host must perform DAD for each of the IPv6 addresses (unless interface-identifier is negotiated to be unique, as may be the case with IPv6 over PPP [RFC5072]).

A host can have one or more IPv6 addresses belonging to one or more prefixes that it learned dynamically (i.e. SLAAC) [RFC4862] or statically.

##### 4.2. Router Behavior

The router that is connected to the hosts via zero or more layer2 switches maintains the neighbor cache, which comprises of one or more bindings between host's identifier (e.g. MAC address, interface-identifier etc.) and assigned IPv6 addresses.

This is true even if the host to host communication does not involve the first-hop router (assuming one or more layer2 switches between router and hosts).

The router periodically exports these bindings pertaining to IPv6 GUA [RFC4291] and/or ULA [RFC4193] using the RECORD\_CLIENTBINDING message to the DHCPv6 server. The periodicity is decided by the lift-time of each neighbor cache entry.

When the router deletes one or more neighbor cache entries for whatever reason (e.g. host disconnected), it notifies the DHCP server using the RECORD\_CLIENTBINDING message with liftime of zero for the relevant clients. The server may decide to purge those entries.

The router expects the acknowledgement from the server in form of RECORD-CLIENTBINDING-ACK to ensure that the message was delivered. This is in accord with rest of the DHCPv6 messages.

#### 4.3. DHCP Server Behavior

The DHCP sever, upon receiving the RECORD-CLIENTBINDING message, records the binding between host's identifier (e.g. MAC address) and IPv6 address(es). DHCP server also records the identity of the router that sent the RECORD-CLIENTBINDING message for a given client.

The DHCP server sends RECORD-CLIENTBINDING-ACK message to the router to acknowledge the receipt of the router sent information.

Each binding is maintained with a lifetime and is expected to be refreshed prior to the expiration. If the server does not receive a RECORD-CLIENTBINDING message prior to expiration, then the server deletes that binding.

### 5. DHCP Messages and Options

#### 5.1. DHCPv6 Messages

This section details one or more messages.

## 5.1.1. RECORD-CLIENTBINDING message

This message is used by the router to inform (export, really) the information (e.g. bindings) from its neighbor cache.

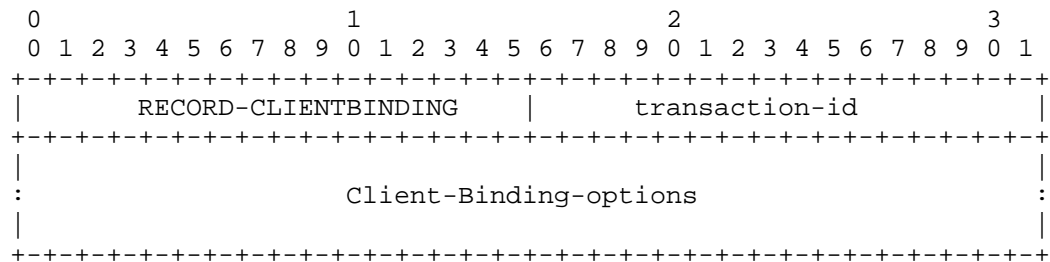


Figure 3 'Record Client Binding' Message

msg-type                      RECORD-CLIENTBINDING message (IANA TBD)

transaction-id                The transaction ID for this message exchange

Client-Binding-options    MUST include at least one 'Client Binding option' (see section 5.2)

## 5.1.2. RECORD-CLIENTBINDING-ACK message

This message is used by the DHCPv6 server to acknowledge the receipt of RECORD-CLIENTBINDING message sent by the router.

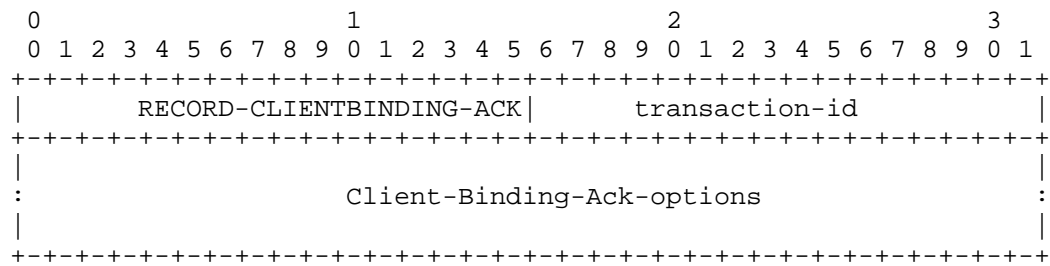


Figure 4 'Record Client Binding Ack' Message

msg-type                RECORD-CLIENTBINDING-ACK message (IANA TBD)

transaction-id        The transaction ID for this message exchange

Client-Binding-Ack-options   None defined by this document.

## 5.2. DHCPv6 Options

This section discusses one or more options used by the messages defined earlier.

### 5.2.1. CLIENT\_BINDING Option

This specification assumes host compliance of [RFC4861], and does not require any changes to the host behavior.

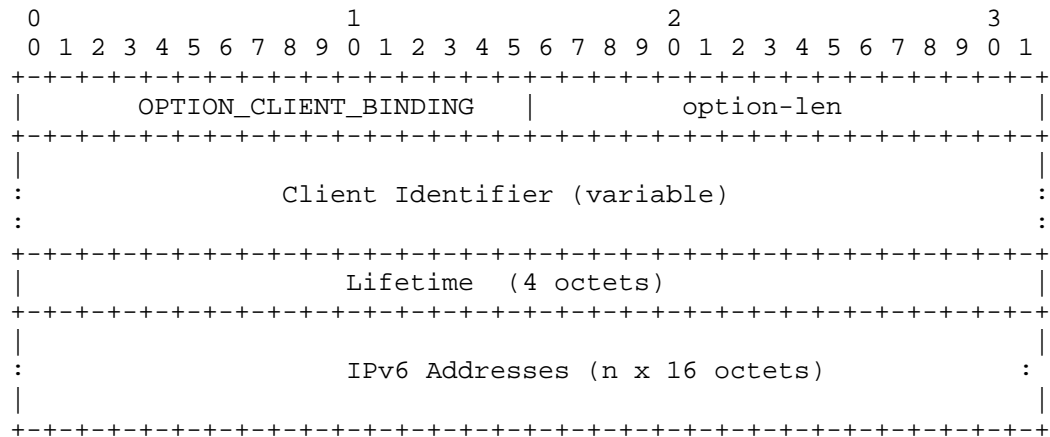


Figure 5 Client Binding Option

option-code      OPTION\_CLIENT\_BINDING (IANA allocation TBD)

option-len      Length, in octets

Client Identifier      Link-layer Address of the host. Encoded as the DUID-LL as defined in section 9.4 of [RFC3315] in case of neighbor cache entry containing host's MAC address.

Lifetime      Expiration time of the binding. This conveys to the DHCP server whether the binding needs to be refreshed (since host is still connected to the router) or deleted (since the host is disconnected to the router).

IPv6 addresses      Zero or more IPv6 addresses assigned to the same Link-layer Address of the host.

Zero IPv6 address is valid only if the lifetime field equals zero. In other words, if the router intends for the server to delete all the addresses pertaining to the device identified by the client identifier, then it could do so by setting the lifetime to zero and not including any IPv6 addresses.



## 6. Security Considerations

None.

## 7. IANA Considerations

This document defines two new DHCPv6 messages and one DHCPv6 option, and requests their IANA assignments.

RECORD-CLIENTBINDING

RECORD-CLIENTBINDING-ACK

OPTION\_CLIENT\_BINDING

## 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, et al., "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

### 8.2. Informative References

- [RFC4291] Hinden, R. and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC 4291, February 2006.
- [RFC4862] Thomson, et al., "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.
- [RFC4861] Narten, et al., "Neighbor Discovery for IP Version 6 (IPv6)", RFC 4861, September 2007.
- [RFC5072] Varada, et al., "IP Version 6 over PPP", RFC 5072, September 2007.

## 9. Acknowledgments

The authors would like to thank Bernie Volz.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Rajiv Asati  
Cisco Systems,  
7025 Kit Creek Rd, RTP, NC, 27709  
Email: rajiva@cisco.com

Dan Wing  
Cisco Systems,  
821 Alder Drive, Milpitas, CA, 95035  
Email: dwing@cisco.com



Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: August 22, 2013

S. Bhandari  
G. Halwasia  
S. Gundavelli  
Cisco Systems  
H. Deng  
China Mobile  
L. Thiebaut  
Alcatel-Lucent  
J. Korhonen  
Renesas Mobile  
February 18, 2013

DHCPv6 class based prefix  
draft-bhandari-dhc-class-based-prefix-04

Abstract

This document introduces options to communicate property and associate metadata with prefixes. It extends DHCPv6 prefix delegation and address allocation using the metadata for selection of prefixes and addresses.

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 August 22, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Motivation . . . . .	3
1.1.1. Mobile networks . . . . .	4
1.1.2. Home networks . . . . .	4
1.2. Terminology . . . . .	5
1.3. Requirements Language . . . . .	5
2. Overview . . . . .	5
2.1. Prefix Property and Class Options . . . . .	5
2.2. Consideration for different DHCPv6 entities . . . . .	7
2.2.1. Requesting Router Behavior for IA_PD allocation . . . . .	7
2.2.2. Delegating Router Behavior for IA_PD allocation . . . . .	8
2.2.3. DHCPv6 Client Behavior for IA_NA allocation . . . . .	9
2.2.4. DHCPv6 Server Behavior for IA_NA allocation . . . . .	9
2.3. Usage . . . . .	10
2.3.1. Class based prefix and IA_NA allocation . . . . .	10
2.3.2. Class based prefix and IA_PD allocation . . . . .	10
2.3.3. Class based prefix and SLAAC . . . . .	10
2.3.4. Class based prefix and applications . . . . .	11
3. Example Application . . . . .	11
3.1. Mobile gateway example . . . . .	11
3.1.1. Class based prefix delegation . . . . .	12
3.1.2. IPv6 address assignment from class based prefix . . . . .	13
3.2. Homenet Example . . . . .	14
3.2.1. Class based prefix delegation to the HGW . . . . .	15
3.2.2. IPv6 Assignment to Homenet hosts using stateful DHCPv6 . . . . .	16
4. Acknowledgements . . . . .	16
5. Contributors . . . . .	17
6. IANA Considerations . . . . .	17
6.1. OPTION_PREFIX_PROPERTY values . . . . .	17
7. Security Considerations . . . . .	18
8. Change History (to be removed prior to publication as an RFC) . . . . .	18
9. References . . . . .	19
9.1. Normative References . . . . .	19
9.2. Informative References . . . . .	20
Authors' Addresses . . . . .	20

## 1. Introduction

In IPv6 a network interface can acquire multiple addresses from the same scope. In such a multi-prefix network each of the multiple prefixes can have a specific property and purpose associated with it. Example: In a mobile network a mobile device can be assigned a prefix from its home network and another from the visiting network that it is attached to. Another example is a prefix may provide free internet access without offering any quality of service guarantees while another prefix may be charged along with providing quality of service guarantees for network service access. A prefix can have well defined properties that is universal and have a metadata associated with it that communicates its local significance. The properties and metadata of prefix will be relevant for prefix delegation, source address selection as elaborated in the subsequent sections.

This document defines `OPTION_PREFIX_PROPERTY` option that communicates property of the prefix that is universally understood. This document defines `OPTION_PREFIX_CLASS` option to communicate metadata of the prefix that communicates the prefix's local significance.

This document discusses usage of `OPTION_PREFIX_CLASS` to request and select prefixes with specific metadata via `IA_PD` and `IA_NA` as defined in [RFC3633] and [RFC3315] respectively. This document defines the behavior of the DHCPv6 server, the DHCPv6 prefix requesting router and the DHCPv6 client to use `OPTION_PREFIX_CLASS` option for requesting and selecting prefixes and addresses.

The network address can be configured via DHCPv6 as defined in [RFC3315] or via Stateless Address Autoconfiguration (SLAAC) as defined in [RFC4862], additional information of a prefix can be provided via DHCPv6 or via IPv6 Router Advertisement (RA). The information provided in the options defined in this document `OPTION_PREFIX_PROPERTY` and `OPTION_PREFIX_CLASS` can be used for source address selection. Communicated property and metadata information about the prefix via IPv6 Router Advertisement (RA) will be dealt with in separate document [I-D.korhonen-dmm-prefix-properties].

### 1.1. Motivation

In this section motivation for class based prefix delegation that qualifies the delegated prefix with additional class information is described in the context of mobile networks and home networks. The property information attached to a delegated prefix helps to distinguish a delegated IPv6 prefix and selection of the prefix by different applications using it.

#### 1.1.1.1. Mobile networks

In the mobile network architecture, there is a mobile router which functions as a IP network gateway and provides IP connectivity to mobile nodes. Mobile router can be the requesting router requesting delegated IPv6 prefix using DHCPv6. Mobile router can assume the role of DHCPv6 server for mobile nodes(DHCPv6 clients) attached to it. A mobile node in mobile network architecture can be associated with multiple IPv6 prefixes belonging to different domains for e.g. home address prefix, care of address prefix as specified in [RFC3775].

The delegated prefixes when seen from the mobile router perspective appear to be like any other prefix, but each prefixes have different metadata referred to as "Prefix Color" in the mobile networks. Some delegated prefixes may be topologically local and some may be remote prefixes anchored on a global anchor, but available to the local anchor by means of tunnel setup in the network between the local and global anchor. Some may be local with low latency characteristics suitable for voice call break-out, some may have global mobility support. So, the prefixes have different properties and it is required for the application using the prefix to learn about this property in order to use it intelligently. There is currently no semantics in DHCPv6 prefix delegation that can carry this information to specify properties of a delegated prefix. In this scenario, the mobile router is unable to further delegate a longer prefix intelligently based on properties of the prefix learnt. Neither is a mobile device able to learn about the property of the prefix assigned to influence source address selection. Example to determine if the prefix is a home address or care of address.

#### 1.1.1.2. Home networks

In a fixed network environment, the homenet CPE may also function as both a DHCPv6 client (requesting the IA\_PD(s)) and a DHCPv6 server allocating prefixes from delegated prefix(es) to downstream home network hosts. Some service providers may wish to delegate multiple prefixes to the CPE for use by different services classes and traffic types.

Motivations for this include:

- o Using source prefix to identify the service class / traffic type that is being transported. The source prefix may then reliably be used as a parameter for differentiated services or other purposes. E.g. [I-D.jiang-v6ops-semantic-prefix]



- o Using the specific source prefix as a host identifier for other services. E.g. as an input parameter to a DHCPv4 over IPv6 server [I-D.ietf-dhc-dhcpv4-over-ipv6]

To meet these requirements, when the CPE (functioning as a DHCPv6 server) receives an IA\_NA or IA\_TA request from a homenet host, a mechanism is required so that the correct prefix for requested service class can be selected for allocation. Likewise for DHCPv6 clients located in the homenet, a mechanism is necessary so that the intended service class for a requested prefix can be signalled to the DHCPv6 server.

### 1.2. Terminology

This document uses the terminology defined in [RFC2460], [RFC3315] and [RFC3633].

### 1.3. Requirements Language

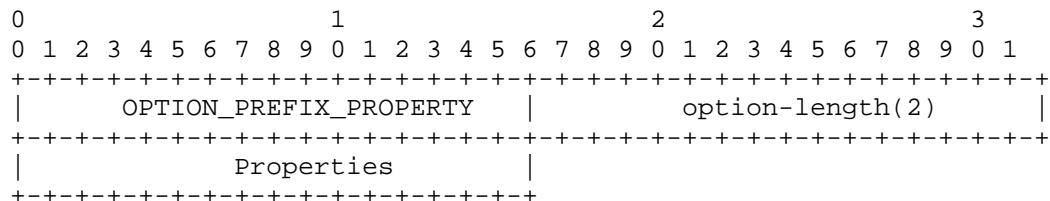
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

## 2. Overview

This section defines prefix property and prefix class options in IA\_PD and IA\_NA. This section defines the behavior of the delegating router, the requesting router and the DHCPv6 client. It discusses these options in the context of a DHCPv6 information request from a DHCPv6 client to a DHCPv6 server.

### 2.1. Prefix Property and Class Options

The format of the DHCPv6 prefix property and prefix class options are shown below.

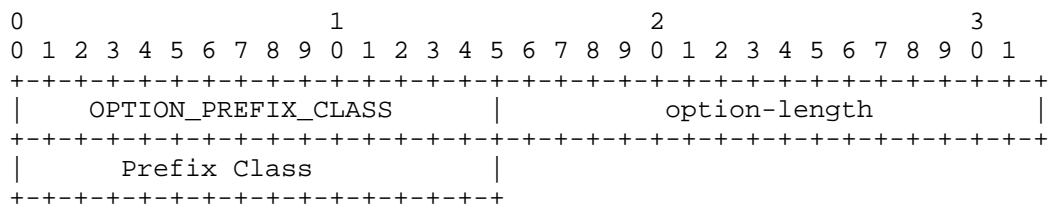


option-code:           OPTION\_PREFIX\_PROPERTY (TBD1)  
option-length:         2  
Properties:            16 bits maintained as  
                      OPTION\_PREFIX\_PROPERTY in  
                      IANA registered namespace.  
Each value in the registry represents a property.  
Multiple properties can be represented by bitwise  
ORing of the individual property values in this  
field.

#### Prefix Property Option

The individual property are maintained in OPTION\_PREFIX\_PROPERTY values enumeration explained in Section Section 6.1.

Along with the OPTION\_PREFIX\_PROPERTY a metadata associated with the prefix that is of local relevance is communicated using OPTION\_PREFIX\_CLASS defined below:



option-code:           OPTION\_PREFIX\_CLASS (TBD2)  
option-length:         2  
Prefix Class:          16 bit integer with the integer value  
                      of local significance.

#### Prefix Class Option

## 2.2. Consideration for different DHCPv6 entities

The model of operation of communicating prefixes to be used by a DHCPv6 server is as follows. A requesting router requests prefix(es) from the delegating router, as described in Section 2.2.1. A delegating router is provided IPv6 prefixes to be delegated to the requesting router. Examples of ways in which the delegating router is provided these prefixes are:

- o Configuration
- o Prefix delegated via a DHCPv6 request to another DHCPv6 server
- o Using a Authentication Authorization Accounting (AAA) protocol like RADIUS [RFC2865]

The delegating router chooses prefix(es) for delegation, and responds with prefix(es) to the requesting router along with additional options in the allocated prefix as described in Section 2.2.2. The requesting router is then responsible for the delegated prefix(es) after the DHCPv6 REQUEST message exchange. For example, the requesting router may create DHCPv6 server configuration pools from the delegated prefix, and function as a DHCPv6 Server. When the requesting router then receives a DHCPv6 IA\_NA requests it can select the address to be allocated based on the OPTION\_PREFIX\_CLASS option received in IA\_NA request or any of the other method as described in Section 2.3.1.

### 2.2.1. Requesting Router Behavior for IA\_PD allocation

DHCPv6 requesting router can request for prefixes in the following ways:

- o In the SOLICIT message within the IA\_PD Prefix option, it MAY include OPTION\_PREFIX\_CLASS requesting prefix delegation for the specific class indicated in the OPTION\_PREFIX\_CLASS option. It can include multiple IA\_PD Prefix options to indicate it's preference for more than one prefix class. The class of prefix it requests is learnt via configuration or any other out of band mechanism not defined in this document.
- o In the SOLICIT message include an OPTION\_ORO option with the OPTION\_PREFIX\_CLASS option code to request prefixes from all the classes that the DHCPv6 server can provide to this requesting Router.

The requesting router parses the OPTION\_PREFIX\_CLASS option in the OPTION\_IAPREFIX option area of the corresponding IA\_PD Prefix option

in the ADVERTISE message. The Requesting router MUST then include all or subset of the received class based prefix(es) in the REQUEST message so that it will be responsible for the prefixes selected.

The requesting router parses and stores OPTION\_PREFIX\_PROPERTY if received with the prefix.

#### 2.2.2. Delegating Router Behavior for IA\_PD allocation

If the Delegating router supports class based prefix allocation by supporting the OPTION\_PREFIX\_CLASS option and it is configured to assign prefixes from different classes, it selects prefixes for class based prefix allocation in the following way:

- o If requesting router includes OPTION\_PREFIX\_CLASS within the IA\_PD Prefix option, it selects prefixes to be offered from that specific class.
- o If requesting router includes OPTION\_PREFIX\_CLASS within OPTION\_ORO, then based on its configuration and policy it MAY offer prefixes from multiple classes available.

The delegating router responds with an ADVERTISE message after populating the IP\_PD option with prefixes from different classes. Along with including the IA\_PD prefix options in the IA\_PD option, it MAY include the OPTION\_PREFIX\_CLASS option in the OPTION\_IAPREFIX option area of the corresponding IA\_PD prefix option with the class information of the prefix.

If neither the OPTION\_ORO nor the IA\_PD option in the SOLICIT message include the OPTION\_PREFIX\_CLASS option, then the delegating router MAY allocate the prefix as specified in [RFC3633] without including the class option in the IA\_PD prefix option in the response.

If OPTION\_ORO option in the Solicit message includes the OPTION\_PREFIX\_CLASS option code but the delegating router does not support the solution described in this specification, then the delegating router acts as specified in [RFC3633]. The requesting router MUST in this case also fall back to the behavior specified in [RFC3633].

If both delegating and requesting routers support class-based prefix allocation, but the delegating router cannot offer prefixes for any other reason, it MUST respond to requesting router with appropriate status code as specified in [RFC3633]. For e.g., if no prefixes are available in the specified class then the delegating router MUST include the status code NoPrefixAvail in the response message.

In addition if the delegating router has additional property associated with the prefix it will be provided in `OPTION_PREFIX_PROPERTY` option.

#### 2.2.3. DHCPv6 Client Behavior for IA\_NA allocation

DHCPv6 client MAY request for an IA\_NA address allocation from a specific prefix class in the following way:

- o In the SOLICIT message within the IA\_NA option, it MAY include the `OPTION_PREFIX_CLASS` requesting address to be allocated from a specific class indicated in that option. The class information to be requested can be learnt via configuration or any other out of band mechanism not described in this document.

If DHCPv6 client receives `OPTION_PREFIX_CLASS`, `OPTION_PREFIX_PROPERTY` options in the IAaddr-options area of the corresponding IA\_NA but does not support one or both of these options, it SHOULD ignore the received option(s).

#### 2.2.4. DHCPv6 Server Behavior for IA\_NA allocation

The DHCPv6 server parses `OPTION_PREFIX_CLASS` option received and when it supports allocation within the requested `OPTION_PREFIX_CLASS` responds with an ADVERTISE message after populating the IA\_NA option with address information from the requested prefix class. Along with including the IA Address options in the IA\_NA option, it also includes the `OPTION_PREFIX_CLASS` option in the corresponding IAaddr-options area.

Even though the IA\_NA option in the SOLICIT message does not include the `OPTION_PREFIX_CLASS` option, based on local policies, the DHCP server MAY select a default `OPTION_PREFIX_CLASS` value for the client and then SHOULD include the `OPTION_PREFIX_CLASS` option in the IAaddr-options area of the corresponding IA\_NA it sends to the client. If both DHCP client and server support class based address allocation, but the DHCP server cannot offer addresses in the specified Usage class then the DHCP server MUST include the status code `NoAddrsAvail` (as defined in [RFC3315]) in the response message. If the DHCP server cannot offer addresses for any other reason, it MUST respond to client with appropriate status code as specified in [RFC3315]. In addition if the server has additional property associated with the prefix by means of configuration or learnt from DHCPv6 prefix delegation or derived via any other means it MUST be sent as `OPTION_PREFIX_PROPERTY` option.

### 2.3. Usage

Class based prefix delegation can be used by the requesting router to configure itself as a DHCPv6 server to serve its DHCPv6 clients. It can allocate longer prefixes from a delegated shorter prefix it received, for serving IA\_NA and IA\_PD requests. Prefix property and class can be used for source address selection by applications using the prefix for communication.

#### 2.3.1. Class based prefix and IA\_NA allocation

The requesting router can use the delegated prefix(es) from different classes (for example "video" (1), "guest"(2), "voice" (3) etc), for assigning the IPv6 addresses to the end hosts through DHCPv6 IA\_NA based on a preconfigured mapping with OPTION\_PREFIX\_CLASS option, the following conditions MAY be observed:

- o It MAY have a pre-configured mapping between the prefix class and OPTION\_USER\_CLASS option received in IA\_NA.
- o It MAY match the OPTION\_PREFIX\_CLASS if the IA\_NA request received contains OPTION\_PREFIX\_CLASS.
- o It MAY have a pre-configured mapping between the prefix class and the client DUID received in DHCPv6 message.
- o It MAY have a pre-configured mapping between the prefix class and its network interface on which the IA\_NA request was received.

The requesting router playing the role of a DHCPv6 server can ADVERTISE IA\_NA from a class of prefix(es) thus selected.

#### 2.3.2. Class based prefix and IA\_PD allocation

If the requesting router, receives prefix(es) for different classes (for example "video"(1), "guest"(2), "voice"(3) etc), it can use these prefix(es) for assigning the longer IPv6 prefixes to requesting routers it serves through DHCPv6 IA\_PD by assuming the role of delegating router, its behavior is explained in Section 2.2.2.

#### 2.3.3. Class based prefix and SLAAC

DHCPv6 IA\_NA and IPv6 Stateless Address Autoconfiguration (SLAAC as defined in [RFC4862]) are two ways by IPv6 addresses can be dynamically assigned to end hosts. Making SLAAC class aware is outside the scope of this document, it is specified in [I-D.korhonen-dmm-prefix-properties].

#### 2.3.4. Class based prefix and applications

Applications within a host can do source address selection based on the class of the prefix learnt in `OPTION_PREFIX_PROPERTY` and `OPTION_PREFIX_CLASS` using rules defined in [RFC6724].

### 3. Example Application

#### 3.1. Mobile gateway example

The following sub-sections provide examples of class based prefix delegation and how it is used in a mobile network. Each of the examples will refer to the below network:

The example network consists of :

**Mobile Gateway** It is network entity anchoring IP traffic in the mobile core network. This entity allocates an IP address which is topologically valid in the mobile network and may act as a mobility anchor if handover between mobile and Wi-Fi is supported.

**Mobile Nodes (MN)** A host or router that changes its point of attachment from one network or subnetwork to another. A mobile node may change its location without changing its IP address; it may continue to communicate with other Internet nodes at any location using its (constant) IP address, assuming link-layer connectivity to a point of attachment is available.

**Access Point (AP)** A wireless access point, identified by a MAC address, providing service to the wired network for wireless nodes.

**Access Router (AR)** An IP router residing in an access network and connected to one or more Access Point(AP)s. An AR offers IP connectivity to MNs.

**WLAN controller (WLC)** The entity that provides the centralized forwarding, routing function for the user traffic.

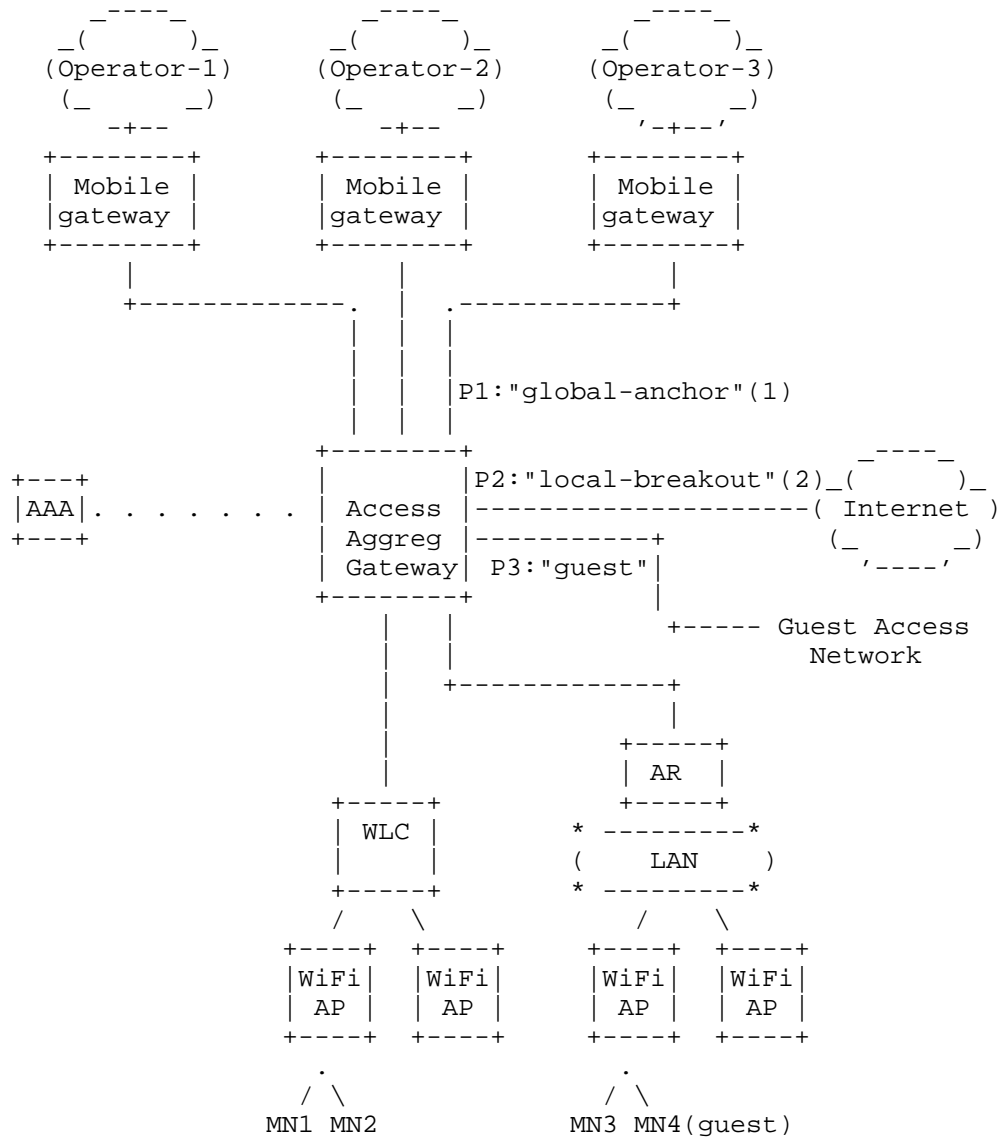


Figure 1: Example mobile network

### 3.1.1. Class based prefix delegation

The Access Aggregation Gateway requests for Prefix delegation from Mobile gateway and associates the prefix received with class "global-anchor"(1). The Access Aggregation Gateway is preconfigured to



provide prefixes from the following classes: "global-anchor" (1), "local-breakout"(2), "guest"(3). It has a preconfigured policy to advertise prefixes to requesting routers and mobile nodes based on the service class supported by the service provider for the requesting device. In the example mobile network, the Access Router(AR) requests class based prefix allocation by sending a DHCPv6 SOLICIT message and include OPTION\_PREFIX\_CLASS in the OPTION\_ORO.

The Access Router (AR) receives an advertise with following prefixes in the IA\_PD option:

1. P1: IA\_PD Prefix option with a prefix 3001:1::/64 containing OPTION\_PREFIX\_CLASS set to "global-anchor"(1)
2. P2: IA\_PD Prefix option with a prefix 3001:2::/64 containing OPTION\_PREFIX\_CLASS set to "local-breakout"(2)
3. P3: IA\_PD Prefix option with a prefix 3001:3::/64 containing OPTION\_PREFIX\_CLASS set to "guest"(3)

It sends a REQUEST message with all of above prefixes and receives a REPLY message with prefixes allocated for each of the requested class.

### 3.1.2. IPv6 address assignment from class based prefix

When the Access Router(AR) receives a DHCPv6 SOLICIT requesting IA\_NA from the mobile node that has mobility service enabled, it offers an IPv6 address from the prefix class "global-anchor"(1). For MN3 it advertises 3001:1::1 as the IPv6 address in OPTION\_IAADDR in response to the IA\_NA request.

The Mobile Node(MN4) Figure 1 sends a DHCPv6 SOLICIT message requesting IA\_NA address assignment with OPTION\_USER\_CLASS option containing the value "guest" towards the CPE. The Access Router(AR) assumes the role of the DHCPv6 server and sends an ADVERTISE to the MN with OPTION\_IA\_NA containing an IPv6 address in OPTION\_IAADDR from the "guest"(3) class. The IPv6 address in the OPTION\_IAADDR is set to 3001:3::1. The "guest" class can also be distinguished based on a preconfigured interface or SSID advertised for MNs connecting to it.

When the Access Aggregation Gateway receives a DHCPv6 SOLICIT requesting IA\_NA from MNs through WLC and it has a preconfigured profile to provide both local-breakout internet access and global-anchor, it offers an IPv6 address from the class "local-breakout" (2) and "global-anchor"(1). For MN1 it advertises 3001:2::1 and 3001:1::2 as the IPv6 address in OPTION\_IAADDR in response to the IA\_NA request. Applications within MN1 can choose to use the

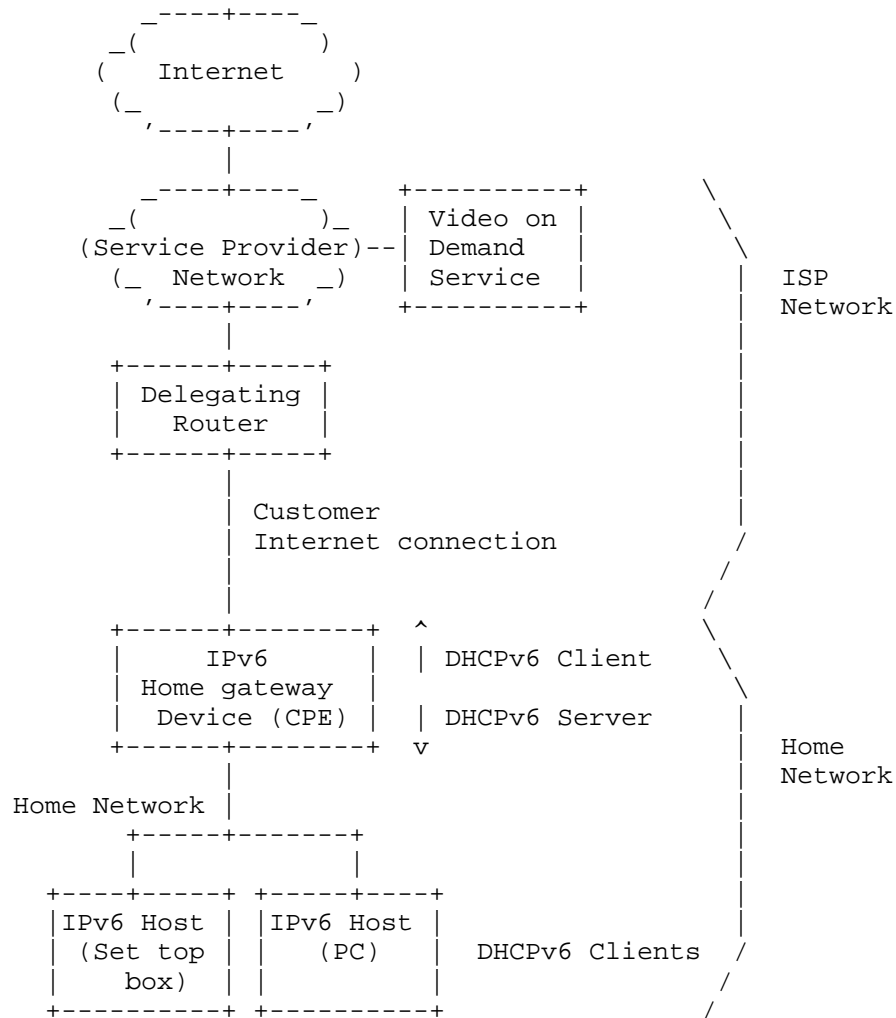
appropriate prefix based on the mobility enabled or local-breakout property attached to the prefix based on source address selection policy.

The prefixes that are globally anchored and hence have mobility can be advertised with OPTION\_PREFIX\_PROPERTY set to 0x0002 to convey that the prefix provides network based mobility as listed in Section 6.1. If the prefix also provides security guarantees OPTION\_PREFIX\_PROPERTY can be set to 0x0009 to indicated mobility and security guarantees by bitwise ORing of both the properties.

### 3.2. Homenet Example

The following sub-section describes an example of class based prefix delegation in a home network environment. The network consists of the following elements:

- o Home Gateway (HGW) device: a routing device located in the customer's premises that provides connectivity between the customer and the service provider. In this example, the HGW is functioning as both a DHCP client towards the service provider's DHCP infrastructure and a DHCP server towards hosts located in the home network.
- o IPv6 Set Top Box (STB): A dedicated, IPv6 attached, video on demand device.
- o IPv6 PC: An IPv6 attached personal computer.
- o Delegating Router: The router in the ISPs network acting as a DHCP server for the IA\_PD request.
- o Video On Demand (VOD) service: A server providing unicast based streaming video content to subscribers.



Simple home network with Data and Video devices

### 3.2.1. Class based prefix delegation to the HGW

In this example, two different services are being run on the same network. The service provider wishes that traffic is sourced from different prefixes by the home network clients [I-D.jiang-v6ops-semantic-prefix]. The HGW (requesting router) has been configured to request prefix delegation from the ISPs delegating router with the usage classes "video" (1) and "internet"(2) the meaning of these being of relevance to the ISP operating this and

application that are configured out of band to utilize it.

The delegating router is preconfigured to advertise prefixes with these service classes. The HGW sends two IA\_PD options within the SOLICIT message, one with OPTION\_PREFIX\_CLASS "video" (1) and the second with "internet" (2). The HGW receives an advertise with the following prefixes in the IA\_PD option:

1. P1: IA\_PD Prefix option with a prefix 3001:5::/56 containing OPTION\_PREFIX\_CLASS set to "video" (1)
2. P2: IP\_PD Prefix option with a prefix 3001:6::/56 containing OPTION\_PREFIX\_CLASS set to "internet" (2)

It sends a REQUEST message with all of the above prefixes and receives a REPLY message with prefixes allocated for each of the requested classes. The HGW then configures a /64 prefix from each of the delegated prefixes on its LAN interface [RFC6204] and sends out router advertisements (RAs) with the "M" and "O" bits set.

#### 3.2.2. IPv6 Assignment to Homenet hosts using stateful DHCPv6

STB sends a DHCPv6 SOLICIT message with the OPTION\_PREFIX\_CLASS option set to "video" (1) within the IA\_NA. The HGW checks the requested prefix class against the available prefixes it has been delegated and advertises 3001:5::1 to the STB. The STB then configures this address on its LAN interface and uses it for sourcing requests to the VOD service.

The PC sends a DHCPv6 SOLICIT message with the OPTION\_PREFIX\_CLASS option set to "internet" within the IA\_NA or without OPTION\_PREFIX\_PROPERTY. The HGW checks the requested prefix class against the available prefixes it has been delegated and advertises 3001:6::1 to the PC or in absence of OPTION\_PREFIX\_CLASS in the solicit HGW is preconfigured to assign from the "internet"(2) class as the default. The PC then configures this address on its LAN interface and uses it for sourcing requests to the Internet.

#### 4. Acknowledgements

The authors would like to acknowledge review and guidance received from Frank Brockners, Wojciech Dec, Richard Johnson, Erik Nordmark, Hemant Singh, Mark Townsley, Ole Troan, Bernie Volz

## 5. Contributors

Authors would like to thank contributions to use cases and text for various sections received from Ian Farrer and Sindhura Bandi.

## 6. IANA Considerations

IANA is requested to assign an option code to `OPTION_PREFIX_PROPERTY` (TBD1) and `OPTION_PREFIX_CLASS` (TBD2) from the "DHCPv6 and DHCPv6 options" registry (<http://www.iana.org/assignments/dhcpv6-parameters/dhcpv6-parameters.xml>).

### 6.1. `OPTION_PREFIX_PROPERTY` values

IANA is requested to reserve and maintain registry of `OPTION_PREFIX_PROPERTY` values and manage allocation of values as per as per policy defined in [RFC5226] with IETF assigned values requiring IETF consensus, RFC Required policy, any other values other than the ones listed below are not valid.

1. 0x0001 The prefix cannot be used to reach the Internet
2. 0x0002 The prefix provides network based mobility
3. 0x0004 The prefix requires authentication
4. 0x0008 The prefix is assigned on an interface that provides security guarantees
5. 0x0010 Usage is charged
6. 0x0020 Unassigned
7. 0x0040 Unassigned
8. 0x0080 Unassigned
9. 0x0100 Unassigned
10. 0x0200 Unassigned
11. 0x0400 Unassigned
12. 0x0800 Unassigned
13. 0x1000 Unassigned

- 14. 0x2000 Unassigned
- 15. 0x4000 Unassigned
- 16. 0x8000 Unassigned

## 7. Security Considerations

Security issues related to DHCPv6 which are described in section 23 of [RFC3315] and [RFC3633] apply for scenarios mentioned in this draft as well.

## 8. Change History (to be removed prior to publication as an RFC)

Changes from -00 to -01

- a. Modified motivation section to focus on mobile networks
- b. Modified example with a mobile network and class based prefix delegation in it

Changes from -01 to -02

- a. Modified option format to be enumerated values
- b. Added IANA section to request managing of registry for the enumerated values
- c. Added initial values for the class
- d. Added section for applications to select address with a specific property

Changes from -02 to -03

- a. Added server behaviour for IA\_NA and IA\_PD allocation
- b. Added Class based Information-Request usage

Changes from -03 to -04

- a. Added homenet use case
- b. Split usage class into a fixed IANA maintained properties registry and a prefix class

## 9. References

### 9.1. Normative References

- [I-D.ietf-dhc-dhcpv4-over-ipv6]  
Cui, Y., Wu, P., Wu, J., and T. Lemon, "DHCPv4 over IPv6 Transport", draft-ietf-dhc-dhcpv4-over-ipv6-05 (work in progress), September 2012.
- [I-D.jiang-v6ops-semantic-prefix]  
Jiang, S., Sun, Q., and I. Farrer, "A Framework for Semantic IPv6 Prefix and Gap Analysis", draft-jiang-v6ops-semantic-prefix-02 (work in progress), January 2013.
- [I-D.korhonen-dmm-prefix-properties]  
Korhonen, J., Patil, B., Gundavelli, S., Seite, P., and D. Liu, "IPv6 Prefix Mobility Management Properties", draft-korhonen-dmm-prefix-properties-03 (work in progress), October 2012.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.
- [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.
- [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.
- [RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

- [RFC6204] Singh, H., Beebee, W., Donley, C., Stark, B., and O. Troan, "Basic Requirements for IPv6 Customer Edge Routers", RFC 6204, April 2011.
- [RFC6724] Thaler, D., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, September 2012.

## 9.2. Informative References

- [RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.

## Authors' Addresses

Shwetha Bhandari  
Cisco Systems  
Cessna Business Park, Sarjapura Marathalli Outer Ring Road  
Bangalore, KARNATAKA 560 087  
India

Phone:  
Email: shwethab@cisco.com

Gaurav Halwasia  
Cisco Systems  
Cessna Business Park, Sarjapura Marathalli Outer Ring Road  
Bangalore, KARNATAKA 560 087  
India

Phone: +91 80 4426 1321  
Email: ghalwasi@cisco.com

Sri Gundavelli  
Cisco Systems  
170 West Tasman Drive  
San Jose, CA 95134  
USA

Email: sgundave@cisco.com



Hui Deng  
China Mobile  
53A, Xibianmennei Ave., Xuanwu District  
Beijing 100053  
China

Email: denghui02@gmail.com

Laurent Thiebaut  
Alcatel-Lucent  
France

Email: laurent.thiebaut@alcatel-lucent.com

Jouni Korhonen  
Renesas Mobile  
Linnoitustie 6  
FIN-02600 Espoo,  
Finland

Phone:  
Email: jouni.nospam@gmail.com



Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: January 16, 2014

S. Bhandari  
G. Halwasia  
S. Gundavelli  
Cisco Systems  
H. Deng  
China Mobile  
L. Thiebaut  
Alcatel-Lucent  
J. Korhonen  
Renesas Mobile  
I. Farrer  
Deutsche Telekom AG  
July 15, 2013

DHCPv6 class based prefix  
draft-bhandari-dhc-class-based-prefix-05

#### Abstract

This document introduces options to communicate property and associate meta data with prefixes. It extends DHCPv6 prefix delegation and address allocation using the meta data for selection of prefixes and addresses.

#### 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 16, 2014.

#### Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

## Table of Contents

1. Introduction . . . . .	3
1.1. Motivation . . . . .	3
1.1.1. Mobile networks . . . . .	4
1.1.2. Home networks . . . . .	4
1.2. Terminology . . . . .	5
1.3. Requirements Language . . . . .	5
2. Overview . . . . .	5
2.1. Prefix Property and Class Options . . . . .	5
2.2. Consideration for different DHCPv6 entities . . . . .	6
2.2.1. Requesting Router Behavior for IA_PD allocation . . . . .	7
2.2.2. Delegating Router Behavior for IA_PD allocation . . . . .	8
2.2.3. DHCPv6 Client Behavior for IA_NA allocation . . . . .	9
2.2.4. DHCPv6 Server Behavior for IA_NA allocation . . . . .	9
2.3. Usage . . . . .	10
2.3.1. Class based prefix and IA_NA allocation . . . . .	10
2.3.2. Class based prefix and IA_PD allocation . . . . .	10
2.3.3. Class based prefix and SLAAC . . . . .	10
2.3.4. Class based prefix and applications . . . . .	11
3. Example Application . . . . .	11
3.1. Mobile gateway example . . . . .	11
3.1.1. Class based prefix delegation . . . . .	13
3.1.2. IPv6 address assignment from class based prefix . . . . .	13
3.2. Homenet Example . . . . .	14
3.2.1. Class based prefix delegation to the HGW . . . . .	15
3.2.2. IPv6 Assignment to Homenet hosts using stateful DHCPv6 . . . . .	16
4. Acknowledgements . . . . .	17
5. Contributors . . . . .	17
6. IANA Considerations . . . . .	17
6.1. OPTION_PREFIX_PROPERTY values . . . . .	17
7. Security Considerations . . . . .	18
8. Change History (to be removed prior to publication as an RFC) . . . . .	18
9. References . . . . .	19
9.1. Normative References . . . . .	19
9.2. Informative References . . . . .	20
Authors' Addresses . . . . .	20

## 1. Introduction

In IPv6 a network interface can acquire multiple addresses from the same scope. In such a multi-prefix network each of the multiple prefixes can have a specific property and purpose associated with it. Example: In a mobile network a mobile device can be assigned a prefix from its home network and another from the visiting network that it is attached to. Another example is a prefix may provide free Internet access without offering any quality of service guarantees while another prefix may be charged along with providing quality of service guarantees for network service access. A prefix can have well defined properties that is universal and have a meta data associated with it that communicates its local significance. The properties and meta data of prefix will be relevant for prefix delegation, source address selection as elaborated in the subsequent sections.

This document defines `OPTION_PREFIX_PROPERTY` option that communicates property of the prefix that is universally understood. This document defines `OPTION_PREFIX_CLASS` option to communicate meta data of the prefix that communicates the prefix's local significance.

This document discusses usage of `OPTION_PREFIX_CLASS` to request and select prefixes with specific meta data via `IA_PD` and `IA_NA` as defined in [RFC3633] and [RFC3315] respectively. This document defines the behavior of the DHCPv6 server, the DHCPv6 prefix requesting router and the DHCPv6 client to use `OPTION_PREFIX_CLASS` option for requesting and selecting prefixes and addresses.

The network address can be configured via DHCPv6 as defined in [RFC3315] or via Stateless Address Autoconfiguration (SLAAC) as defined in [RFC4862], additional information of a prefix can be provided via DHCPv6 or via IPv6 Router Advertisement (RA). The information provided in the options defined in this document `OPTION_PREFIX_PROPERTY` and `OPTION_PREFIX_CLASS` can be used for source address selection. Communicated property and meta data information about the prefix via IPv6 Router Advertisement (RA) will be dealt with in separate document [I-D.korhonen-6man-prefix-properties].

### 1.1. Motivation

In this section motivation for class based prefix delegation that qualifies the delegated prefix with additional class information is described in the context of mobile networks and home networks. The property information attached to a delegated prefix helps to distinguish a delegated IPv6 prefix and selection of the prefix by different applications using it.

#### 1.1.1.1. Mobile networks

In the mobile network architecture, there is a mobile router which functions as a IP network gateway and provides IP connectivity to mobile nodes. Mobile router can be the requesting router requesting delegated IPv6 prefix using DHCPv6. Mobile router can assume the role of DHCPv6 server for mobile nodes(DHCPv6 clients) attached to it. A mobile node in mobile network architecture can be associated with multiple IPv6 prefixes belonging to different domains for e.g. home address prefix, care of address prefix as specified in [RFC3775].

The delegated prefixes when seen from the mobile router perspective appear to be like any other prefix, but each prefixes have different meta data referred to as "Prefix Color" in the mobile networks. Some delegated prefixes may be topologically local and some may be remote prefixes anchored on a global anchor, but available to the local anchor by means of tunnel setup in the network between the local and global anchor. Some may be local with low latency characteristics suitable for voice call break-out, some may have global mobility support. So, the prefixes have different properties and it is required for the application using the prefix to learn about this property in order to use it intelligently. There is currently no semantics in DHCPv6 prefix delegation that can carry this information to specify properties of a delegated prefix. In this scenario, the mobile router is unable to further delegate a longer prefix intelligently based on properties of the prefix learnt. Neither is a mobile device able to learn about the property of the prefix assigned to influence source address selection. Example to determine if the prefix is a home address or care of address.

#### 1.1.1.2. Home networks

In a fixed network environment, the homenet CPE may also function as both a DHCPv6 client (requesting the IA\_PD(s)) and a DHCPv6 server allocating prefixes from delegated prefix(es) to downstream home network hosts. Some service providers may wish to delegate multiple prefixes to the CPE for use by different services classes and traffic types.

Motivations for this include:

- o Using source prefix to identify the service class / traffic type that is being transported. The source prefix may then reliably be used as a parameter for differentiated services or other purposes. E.g. [I-D.jiang-v6ops-semantic-prefix]

- o Using the specific source prefix as a host identifier for other services. E.g. as an input parameter to a DHCPv4 over IPv6 server [I-D.ietf-dhc-dhcpv4-over-ipv6]

To meet these requirements, when the CPE (functioning as a DHCPv6 server) receives an IA\_NA or IA\_TA request from a homenet host, a mechanism is required so that the correct prefix for requested service class can be selected for allocation. Likewise for DHCPv6 clients located in the homenet, a mechanism is necessary so that the intended service class for a requested prefix can be signalled to the DHCPv6 server.

## 1.2. Terminology

This document uses the terminology defined in [RFC2460], [RFC3315] and [RFC3633].

## 1.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

## 2. Overview

This section defines prefix property and prefix class options in IA\_PD and IA\_NA. This section defines the behavior of the delegating router, the requesting router and the DHCPv6 client. It discusses these options in the context of a DHCPv6 information request from a DHCPv6 client to a DHCPv6 server.

### 2.1. Prefix Property and Class Options

The format of the DHCPv6 prefix property and prefix class options are shown below.

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      OPTION_PREFIX_PROPERTY      |      option-length(2)      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      Properties      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

option-code:      OPTION_PREFIX_PROPERTY (TBD1)
option-length:    2
Properties:       16 bits maintained as
                  OPTION_PREFIX_PROPERTY in
                  IANA registered namespace.
                  Each value in the registry represents a property.
                  Multiple properties can be represented by bitwise
                  ORing of the individual property values in this
                  field.

```

#### Prefix Property Option

The individual property are maintained in OPTION\_PREFIX\_PROPERTY values enumeration explained in Section Section 6.1.

Along with the OPTION\_PREFIX\_PROPERTY a meta data associated with the prefix that is of local relevance is communicated using OPTION\_PREFIX\_CLASS defined below:

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      OPTION_PREFIX_CLASS      |      option-length      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      Prefix Class      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

option-code:      OPTION_PREFIX_CLASS (TBD2)
option-length:    2
Prefix Class:     16 bit integer with the integer value
                  of local significance.

```

#### Prefix Class Option

### 2.2. Consideration for different DHCPv6 entities



The model of operation of communicating prefixes to be used by a DHCPv6 server is as follows. A requesting router requests prefix(es) from the delegating router, as described in Section 2.2.1. A delegating router is provided IPv6 prefixes to be delegated to the requesting router. Examples of ways in which the delegating router is provided these prefixes are:

- o Configuration
- o Prefix delegated via a DHCPv6 request to another DHCPv6 server
- o Using a Authentication Authorization Accounting (AAA) protocol like RADIUS [RFC2865]

The delegating router chooses prefix(es) for delegation, and responds with prefix(es) to the requesting router along with additional options in the allocated prefix as described in Section 2.2.2. The requesting router is then responsible for the delegated prefix(es) after the DHCPv6 REQUEST message exchange. For example, the requesting router may create DHCPv6 server configuration pools from the delegated prefix, and function as a DHCPv6 Server. When the requesting router then receives a DHCPv6 IA\_NA requests it can select the address to be allocated based on the OPTION\_PREFIX\_CLASS option received in IA\_NA request or any of the other method as described in Section 2.3.1.

#### 2.2.1. Requesting Router Behavior for IA\_PD allocation

DHCPv6 requesting router can request for prefixes in the following ways:

- o In the SOLICIT message within the IA\_PD Prefix option, it MAY include OPTION\_PREFIX\_CLASS requesting prefix delegation for the specific class indicated in the OPTION\_PREFIX\_CLASS option. It can include multiple IA\_PD Prefix options to indicate it's preference for more than one prefix class. The class of prefix it requests is learnt via configuration or any other out of band mechanism not defined in this document.
- o In the SOLICIT message include an OPTION\_ORO option with the OPTION\_PREFIX\_CLASS option code to request prefixes from all the classes that the DHCPv6 server can provide to this requesting Router.

The requesting router parses the `OPTION_PREFIX_CLASS` option in the `OPTION_IAPREFIX` option area of the corresponding `IA_PD` Prefix option in the `ADVERTISE` message. The Requesting router **MUST** then include all or subset of the received class based prefix(es) in the `REQUEST` message so that it will be responsible for the prefixes selected.

The requesting router parses and stores `OPTION_PREFIX_PROPERTY` if received with the prefix.

#### 2.2.2. Delegating Router Behavior for `IA_PD` allocation

If the Delegating router supports class based prefix allocation by supporting the `OPTION_PREFIX_CLASS` option and it is configured to assign prefixes from different classes, it selects prefixes for class based prefix allocation in the following way:

- o If requesting router includes `OPTION_PREFIX_CLASS` within the `IA_PD` Prefix option, it selects prefixes to be offered from that specific class.
- o If requesting router includes `OPTION_PREFIX_CLASS` within `OPTION_ORO`, then based on its configuration and policy it **MAY** offer prefixes from multiple classes available.

The delegating router responds with an `ADVERTISE` message after populating the `IP_PD` option with prefixes from different classes. Along with including the `IA_PD` prefix options in the `IA_PD` option, it **MAY** include the `OPTION_PREFIX_CLASS` option in the `OPTION_IAPREFIX` option area of the corresponding `IA_PD` prefix option with the class information of the prefix.

If neither the `OPTION_ORO` nor the `IA_PD` option in the `SOLICIT` message include the `OPTION_PREFIX_CLASS` option, then the delegating router **MAY** allocate the prefix as specified in [RFC3633] without including the class option in the `IA_PD` prefix option in the response.

If `OPTION_ORO` option in the `Solicit` message includes the `OPTION_PREFIX_CLASS` option code but the delegating router does not support the solution described in this specification, then the delegating router acts as specified in [RFC3633]. The requesting router **MUST** in this case also fall back to the behavior specified in [RFC3633].

If both delegating and requesting routers support class-based prefix allocation, but the delegating router cannot offer prefixes for any other reason, it MUST respond to requesting router with appropriate status code as specified in [RFC3633]. For e.g., if no prefixes are available in the specified class then the delegating router MUST include the status code NoPrefixAvail in the response message.

In addition if the delegating router has additional property associated with the prefix it will be provided in OPTION\_PREFIX\_PROPERTY option.

#### 2.2.3. DHCPv6 Client Behavior for IA\_NA allocation

DHCPv6 client MAY request for an IA\_NA address allocation from a specific prefix class in the following way:

- o In the SOLICIT message within the IA\_NA option, it MAY include the OPTION\_PREFIX\_CLASS requesting address to be allocated from a specific class indicated in that option. The class information to be requested can be learnt via configuration or any other out of band mechanism not described in this document.

If DHCPv6 client receives OPTION\_PREFIX\_CLASS, OPTION\_PREFIX\_PROPERTY options in the IAaddr-options area of the corresponding IA\_NA but does not support one or both of these options, it SHOULD ignore the received option(s).

#### 2.2.4. DHCPv6 Server Behavior for IA\_NA allocation

The DHCPv6 server parses OPTION\_PREFIX\_CLASS option received and when it supports allocation within the requested OPTION\_PREFIX\_CLASS responds with an ADVERTISE message after populating the IA\_NA option with address information from the requested prefix class. Along with including the IA Address options in the IA\_NA option, it also includes the OPTION\_PREFIX\_CLASS option in the corresponding IAaddr-options area.

Even though the IA\_NA option in the SOLICIT message does not include the OPTION\_PREFIX\_CLASS option, based on local policies, the DHCP server MAY select a default OPTION\_PREFIX\_CLASS value for the client and then SHOULD include the OPTION\_PREFIX\_CLASS option in the IAaddr-options area of the corresponding IA\_NA it sends to the client. If both DHCP client and server support class based address allocation, but the DHCP server cannot offer addresses in the specified Usage class then the DHCP server MUST include the status code NoAddrsAvail (as defined in [RFC3315]) in the response message. If the DHCP server cannot offer addresses for any other reason, it MUST respond to client with appropriate status code as specified in [RFC3315]. In

addition if the server has additional property associated with the prefix by means of configuration or learnt from DHCPv6 prefix delegation or derived via any other means it MUST be sent as OPTION\_PREFIX\_PROPERTY option.

### 2.3. Usage

Class based prefix delegation can be used by the requesting router to configure itself as a DHCPv6 server to serve its DHCPv6 clients. It can allocate longer prefixes from a delegated shorter prefix it received, for serving IA\_NA and IA\_PD requests. Prefix property and class can be used for source address selection by applications using the prefix for communication.

#### 2.3.1. Class based prefix and IA\_NA allocation

The requesting router can use the delegated prefix(es) from different classes (for example "video" (1), "guest"(2), "voice" (3) etc), for assigning the IPv6 addresses to the end hosts through DHCPv6 IA\_NA based on a preconfigured mapping with OPTION\_PREFIX\_CLASS option, the following conditions MAY be observed:

- o It MAY have a pre-configured mapping between the prefix class and OPTION\_USER\_CLASS option received in IA\_NA.
- o It MAY match the OPTION\_PREFIX\_CLASS if the IA\_NA request received contains OPTION\_PREFIX\_CLASS.
- o It MAY have a pre-configured mapping between the prefix class and the client DUID received in DHCPv6 message.
- o It MAY have a pre-configured mapping between the prefix class and its network interface on which the IA\_NA request was received.

The requesting router playing the role of a DHCPv6 server can ADVERTISE IA\_NA from a class of prefix(es) thus selected.

#### 2.3.2. Class based prefix and IA\_PD allocation

If the requesting router, receives prefix(es) for different classes (for example "video"(1), "guest"(2), "voice"(3) etc), it can use these prefix(es) for assigning the longer IPv6 prefixes to requesting routers it serves through DHCPv6 IA\_PD by assuming the role of delegating router, its behavior is explained in Section 2.2.2.

#### 2.3.3. Class based prefix and SLAAC

DHCPv6 IA\_NA and IPv6 Stateless Address Autoconfiguration (SLAAC as defined in [RFC4862]) are two ways by IPv6 addresses can be dynamically assigned to end hosts. Making SLAAC class aware is outside the scope of this document, it is specified in [I-D.korhonen-6man-prefix-properties].

#### 2.3.4. Class based prefix and applications

Applications within a host can do source address selection based on the class of the prefix learnt in OPTION\_PREFIX\_PROPERTY and OPTION\_PREFIX\_CLASS using rules defined in [RFC6724]. The internal data structure and interface for source address selection used by application to choose source prefix with specific property and class in a host is beyond the scope of this document.

### 3. Example Application

#### 3.1. Mobile gateway example

The following sub-sections provide examples of class based prefix delegation and how it is used in a mobile network. Each of the examples will refer to the below network:

The example network consists of :

**Mobile Gateway** It is network entity anchoring IP traffic in the mobile core network. This entity allocates an IP address which is topologically valid in the mobile network and may act as a mobility anchor if handover between mobile and Wi-Fi is supported.

**Mobile Nodes (MN)** A host or router that changes its point of attachment from one network or subnetwork to another. A mobile node may change its location without changing its IP address; it may continue to communicate with other Internet nodes at any location using its (constant) IP address, assuming link-layer connectivity to a point of attachment is available.

**Access Point (AP)** A wireless access point, identified by a MAC address, providing service to the wired network for wireless nodes.

**Access Router (AR)** An IP router residing in an access network and connected to one or more Access Point(AP)s. An AR offers IP connectivity to MNs.

**WLAN controller (WLC)** The entity that provides the centralized forwarding, routing function for the user traffic.

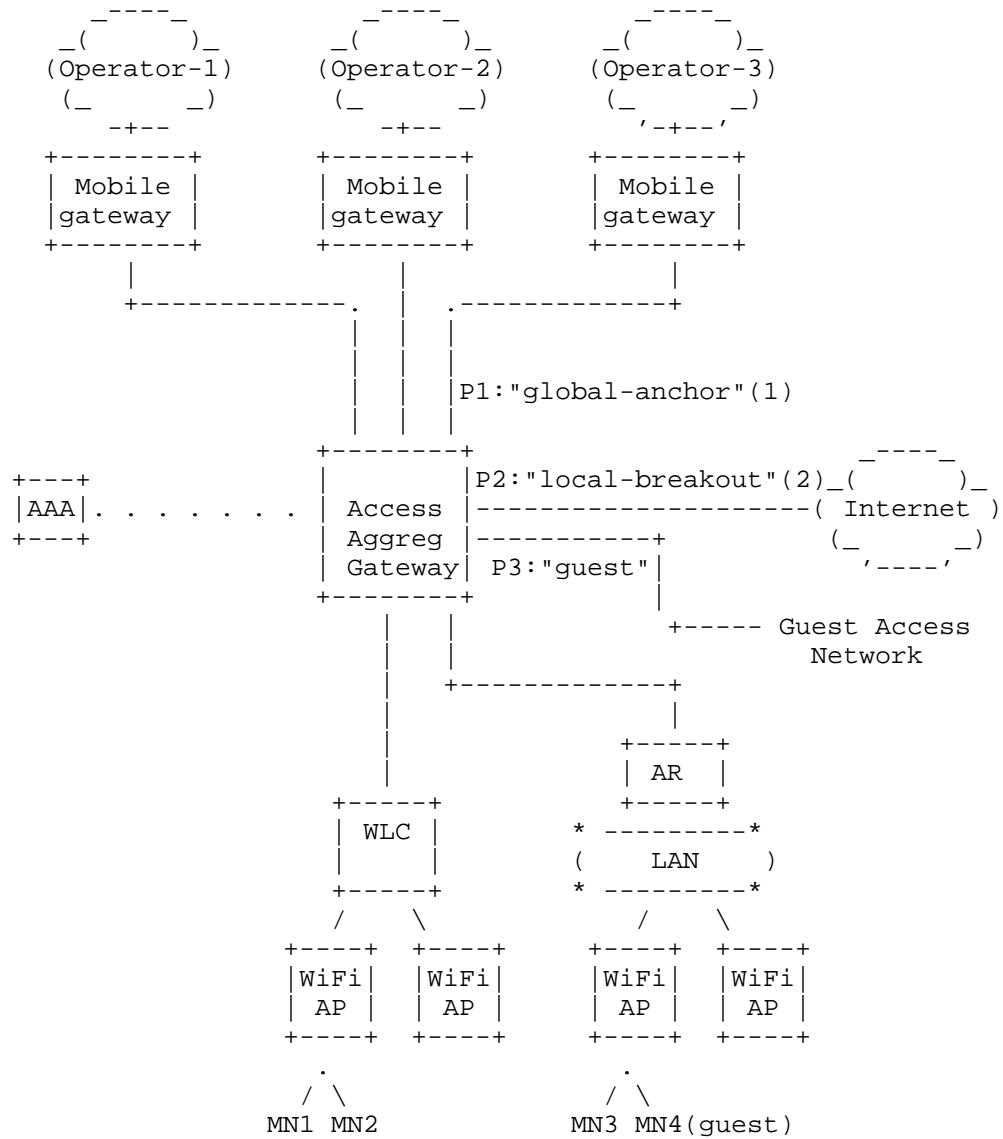


Figure 1: Example mobile network

### 3.1.1. Class based prefix delegation

The Access Aggregation Gateway requests for Prefix delegation from Mobile gateway and associates the prefix received with class "global-anchor"(1). The Access Aggregation Gateway is preconfigured to provide prefixes from the following classes: "global-anchor" (1), "local-breakout"(2), "guest"(3). It has a preconfigured policy to advertise prefixes to requesting routers and mobile nodes based on the service class supported by the service provider for the requesting device. In the example mobile network, the Access Router(AR) requests class based prefix allocation by sending a DHCPv6 SOLICIT message and include OPTION\_PREFIX\_CLASS in the OPTION\_ORO.

The Access Router (AR) receives an advertise with following prefixes in the IA\_PD option:

1. P1: IA\_PD Prefix option with a prefix 3001:1::/64 containing OPTION\_PREFIX\_CLASS set to "global-anchor"(1)
2. P2: IA\_PD Prefix option with a prefix 3001:2::/64 containing OPTION\_PREFIX\_CLASS set to "local-breakout"(2)
3. P3: IA\_PD Prefix option with a prefix 3001:3::/64 containing OPTION\_PREFIX\_CLASS set to "guest"(3)

It sends a REQUEST message with all of above prefixes and receives a REPLY message with prefixes allocated for each of the requested class.

### 3.1.2. IPv6 address assignment from class based prefix

When the Access Router(AR) receives a DHCPv6 SOLICIT requesting IA\_NA from the mobile node that has mobility service enabled, it offers an IPv6 address from the prefix class "global-anchor"(1). For MN3 it advertises 3001:1::1 as the IPv6 address in OPTION\_IAADDR in response to the IA\_NA request.

The Mobile Node(MN4) Figure 1 sends a DHCPv6 SOLICIT message requesting IA\_NA address assignment with OPTION\_USER\_CLASS option containing the value "guest" towards the CPE. The Access Router(AR) assumes the role of the DHCPv6 server and sends an ADVERTISE to the MN with OPTION\_IA\_NA containing an IPv6 address in OPTION\_IAADDR from the "guest"(3) class. The IPv6 address in the OPTION\_IAADDR is set to 3001:3::1. The "guest" class can also be distinguished based on a preconfigured interface or SSID advertised for MNs connecting to it.

When the Access Aggregation Gateway receives a DHCPv6 SOLICIT requesting IA\_NA from MNs through WLC and it has a preconfigured

profile to provide both local-breakout Internet access and global-anchor, it offers an IPv6 address from the class "local-breakout" (2) and "global-anchor"(1). For MN1 it advertises 3001:2::1 and 3001:1::2 as the IPv6 address in OPTION\_IAADDR in response to the IA\_NA request. Applications within MN1 can choose to use the appropriate prefix based on the mobility enabled or local-breakout property attached to the prefix based on source address selection policy.

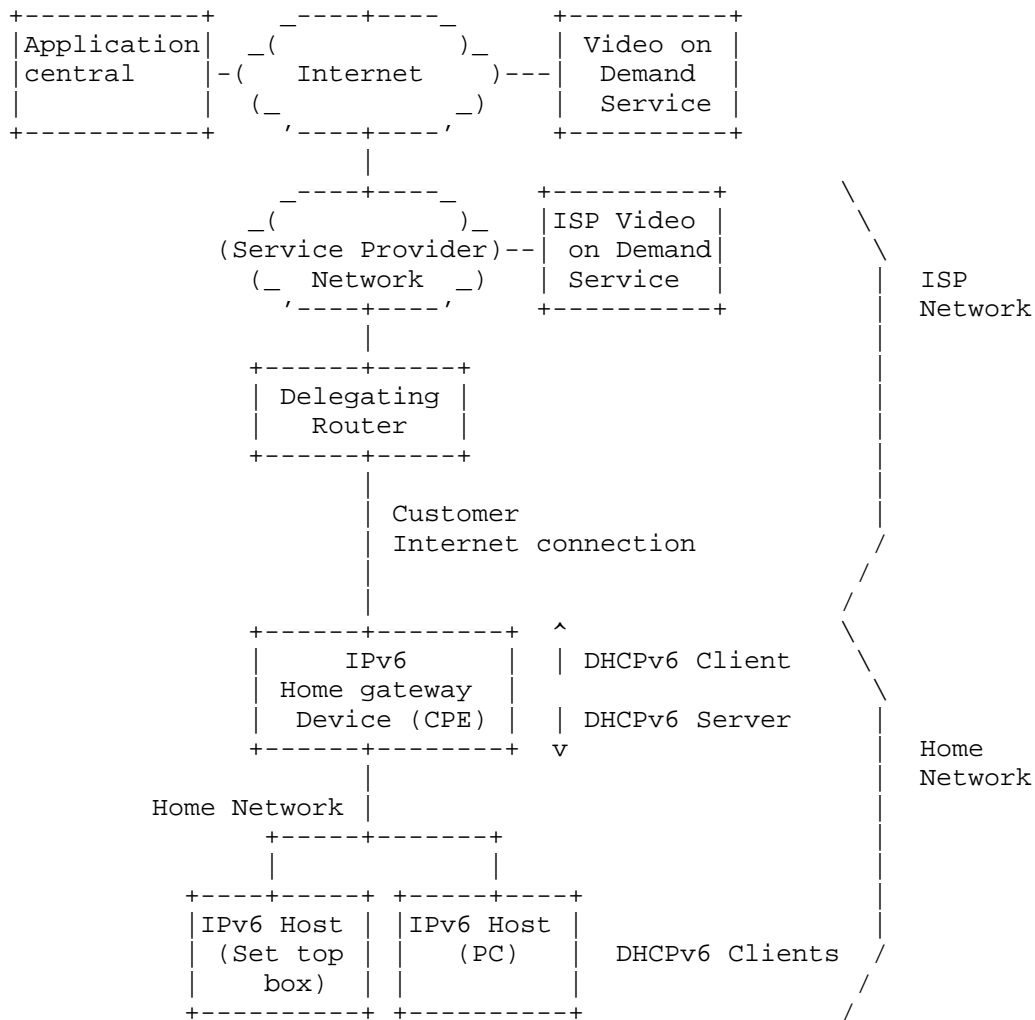
The prefixes that are globally anchored and hence have mobility can be advertised with OPTION\_PREFIX\_PROPERTY set to 0x0002 to convey that the prefix provides network based mobility as listed in Section 6.1. If the prefix also provides security guarantees OPTION\_PREFIX\_PROPERTY can be set to 0x000A to indicate mobility and security guarantees by bitwise ORing of both the properties.

### 3.2. Homenet Example

The following sub-section describes an example of class based prefix delegation in a home network environment. The network consists of the following elements:

- o Home Gateway (HGW) device: a routing device located in the customer's premises that provides connectivity between the customer and the service provider. In this example, the HGW is functioning as both a DHCP client towards the service provider's DHCP infrastructure and a DHCP server towards hosts located in the home network.
- o IPv6 Set Top Box (STB): A dedicated, IPv6 attached, video on demand device.
- o IPv6 PC: An IPv6 attached personal computer
- o Delegating Router: The router in the ISPs network acting as a DHCP server for the IA\_PD request.
- o ISP Video On Demand (ISP-VOD) service: An ISP provided service offering unicast based streaming video content to subscribers.
- o Video On Demand (VOD) service: A server providing unicast based streaming video content to subscribers
- o On demand Video Application: Application hosted on the IPv6 PC
- o Application Central: Application server hosted in the Internet that the On demand Video Application communicates with to access VOD service





Simple home network with Data and Video devices

### 3.2.1. Class based prefix delegation to the HGW

In this example, three different services are being run on the same network. The service provider wishes that traffic is sourced from different prefixes by the home network clients [I-D.jiang-v6ops-semantic-prefix]. The HGW (requesting router) has been configured to request prefix delegation from the ISPs delegating router with the usage classes "video" (1) and "internet"(2) and "video-app" (3) the meaning of these being of relevance to the ISP

operating this and application that are configured out of band to utilize it.

The delegating router is preconfigured to advertise prefixes with these service classes. The HGW sends three IA\_PD options within the SOLICIT message, one with OPTION\_PREFIX\_CLASS "video" (1), the second with "internet" (2) and a third with "video-app" (3). The HGW receives an advertise with the following prefixes in the IA\_PD option:

1. P1: IA\_PD Prefix option with a prefix 3001:5::/56 containing OPTION\_PREFIX\_CLASS set to "video" (1) with OPTION\_PREFIX\_PROPERTY set to 0x0001 indicating there is no internet reach
2. P2: IP\_PD Prefix option with a prefix 3001:6::/56 containing OPTION\_PREFIX\_CLASS set to "internet" (2)
3. P3: IP\_PD Prefix option with a prefix 3001:7::/56 containing OPTION\_PREFIX\_CLASS set to "video-app" (3) with property set to 0x0040 indicating the prefix provides Internet service SLA

It sends a REQUEST message with all of the above prefixes and receives a REPLY message with prefixes allocated for each of the requested classes. The HGW then configures a /64 prefix from each of the delegated prefixes on its LAN interface [RFC6204] and sends out router advertisements (RAs) with the "M" and "O" bits set.

### 3.2.2. IPv6 Assignment to Homenet hosts using stateful DHCPv6

1. STB sends a DHCPv6 SOLICIT message with the OPTION\_PREFIX\_CLASS option set to "video" (1) within the IA\_NA. The HGW checks the requested prefix class against the available prefixes it has been delegated and advertises 3001:5::1 to the STB. The STB then configures this address on its LAN interface and uses it for sourcing requests to the VOD service.
2. The PC sends a DHCPv6 SOLICIT message requesting for IA\_NA with the OPTION\_PREFIX\_CLASS option in ORO indicating support for prefix class. The HGW checks the available prefixes it has been delegated and advertises IA\_NA from P1 (3001:5:2 with property set to 0x0001) , P2 (3001:6::1) and P3 (3001:7::1) to the PC or in absence of OPTION\_PREFIX\_CLASS in the solicit HGW is preconfigured to assign from the "internet"(2) class as the default. The PC then configures these addresses on its LAN interface and uses it for sourcing requests to the Internet.
3. The On demand Video Application on the PC communicates with its well known Application Central using the "internet" prefix and is

directed to use "video-app" prefix if available based on agreement between service provider and on demand video application service provider. The On demand Video Application then connects using the address assigned from P3 that will offer better experience based on the SLA between the providers.

4. If the homenet hosts use SLAAC prefix delegation with the class will use the options and procedure defined in [I-D.korhonen-6man-prefix-properties]

#### 4. Acknowledgements

The authors would like to acknowledge review and guidance received from Frank Brockners, Wojciech Dec, Richard Johnson, Erik Nordmark, Hemant Singh, Mark Townsley, Ole Troan, Bernie Volz, Maciek Konstantynowicz

#### 5. Contributors

Authors would like to thank contributions to use cases and text for various sections received from Sindhura Bandi.

#### 6. IANA Considerations

IANA is requested to assign an option code to OPTION\_PREFIX\_PROPERTY (TBD1) and OPTION\_PREFIX\_CLASS (TBD2) from the "DHCPv6 and DHCPv6 options" registry (<http://www.iana.org/assignments/dhcpv6-parameters/dhcpv6-parameters.xml>).

##### 6.1. OPTION\_PREFIX\_PROPERTY values

IANA is requested to reserve and maintain registry of OPTION\_PREFIX\_PROPERTY values and manage allocation of values as per as per policy defined in [RFC5226] with IETF assigned values requiring IETF consensus, RFC Required policy, any other values other than the ones listed below are not valid.

1. 0x0001 The prefix cannot be used to reach the Internet
2. 0x0002 The prefix provides network based mobility
3. 0x0004 The prefix requires authentication
4. 0x0008 The prefix is assigned on an interface that provides security guarantees
5. 0x0010 Usage is charged

6. 0x0020 The prefix provides multi-homed redundancy
  7. 0x0040 The prefix provides Internet service SLA, based on associated OPTION\_PREFIX\_CLASS
  8. 0x0080 Unassigned
  9. 0x0100 Unassigned
  10. 0x0200 Unassigned
  11. 0x0400 Unassigned
  12. 0x0800 Unassigned
  13. 0x1000 Unassigned
  14. 0x2000 Unassigned
  15. 0x4000 Unassigned
  16. 0x8000 Unassigned
7. Security Considerations
- Security issues related to DHCPv6 which are described in section 23 of [RFC3315] and [RFC3633] apply for scenarios mentioned in this draft as well.
8. Change History (to be removed prior to publication as an RFC)
- Changes from -00 to -01
- a. Modified motivation section to focus on mobile networks
  - b. Modified example with a mobile network and class based prefix delegation in it
- Changes from -01 to -02
- a. Modified option format to be enumerated values
  - b. Added IANA section to request managing of registry for the enumerated values
  - c. Added initial values for the class

- d. Added section for applications to select address with a specific property

Changes from -02 to -03

- a. Added server behaviour for IA\_NA and IA\_PD allocation
- b. Added Class based Information-Request usage

Changes from -03 to -04

- a. Added homenet use case
- b. Split usage class into a fixed IANA maintained properties registry and a prefix class

Changes from -04 to -05

- a. Added on demand video application use case and modified the example section
- b. Added additional properties and reference for SLAAC/ND procedure

## 9. References

### 9.1. Normative References

- [I-D.ietf-dhc-dhcpv4-over-ipv6]  
Cui, Y., Wu, P., Wu, J., and T. Lemon, "DHCPv4 over IPv6 Transport", draft-ietf-dhc-dhcpv4-over-ipv6-06 (work in progress), March 2013.
- [I-D.jiang-v6ops-semantic-prefix]  
Jiang, S., Sun, Q., Farrer, I., and Y. Bo, "A Framework for Semantic IPv6 Prefix", draft-jiang-v6ops-semantic-prefix-03 (work in progress), May 2013.
- [I-D.korhonen-6man-prefix-properties]  
Korhonen, J., Patil, B., Gundavelli, S., Seite, P., and D. Liu, "IPv6 Prefix Properties", draft-korhonen-6man-prefix-properties-02 (work in progress), July 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

- [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.
- [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.
- [RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC6204] Singh, H., Beebe, W., Donley, C., Stark, B., and O. Troan, "Basic Requirements for IPv6 Customer Edge Routers", RFC 6204, April 2011.
- [RFC6724] Thaler, D., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, September 2012.

## 9.2. Informative References

- [RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.

## Authors' Addresses

Shwetha Bhandari  
Cisco Systems  
Cessna Business Park, Sarjapura Marathalli Outer Ring Road  
Bangalore, KARNATAKA 560 087  
India

Email: shwethab@cisco.com

Gaurav Halwasia  
Cisco Systems  
Cessna Business Park, Sarjapura Marathalli Outer Ring Road  
Bangalore, KARNATAKA 560 087  
India

Phone: +91 80 4426 1321  
Email: ghalwasi@cisco.com

Sri Gundavelli  
Cisco Systems  
170 West Tasman Drive  
San Jose, CA 95134  
USA

Email: sgundave@cisco.com

Hui Deng  
China Mobile  
53A, Xibianmennei Ave., Xuanwu District  
Beijing 100053  
China

Email: denghui02@gmail.com

Laurent Thiebaut  
Alcatel-Lucent  
France

Email: laurent.thiebaut@alcatel-lucent.com

Jouni Korhonen  
Renesas Mobile  
Linnoitustie 6  
FIN-02600 Espoo  
Finland

Email: jouni.nospam@gmail.com

Ian Farrer  
Deutsche Telekom AG  
GTN-FM4, Landgrabenweg 151  
Bonn 53227  
Bonn 53227

Email: [ian.farrer@telekom.de](mailto:ian.farrer@telekom.de)



DHC Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: August 29, 2013

S. Jiang  
Huawei Technologies Co., Ltd  
G. Chen  
China Mobile  
S. Krishnan  
Ericsson  
February 25, 2013

A Generic IPv6 Addresses Registration Solution Using DHCPv6  
draft-ietf-dhc-addr-registration-02

Abstract

In networks that are centrally managed, self-generated addresses cause traceability issues due to their decentralized nature. To minimize the issues due to lack of traceability, these self-generated addresses can be registered with the network for allowing centralized address administration. This document defines a generic address registration solution using DHCPv6, using a new ND option and a new DHCPv6 option in order to communicate the use of self-generated addresses. A new Addr-registration-request message type is defined for initiate the address registration request, among with two new Status codes to indicate registration errors on the server side.

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 August 29, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

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

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Overview of Generic Address Registration Solution . . . . .	3
4. Propagating the Address Registration Solicitation . . . . .	4
4.1. ND Address Registration Solicitation Option . . . . .	5
4.2. DHCPv6 Address Registration Solicitation Option . . . . .	5
5. DHCPv6 Addr-registration-request Message . . . . .	6
6. DHCPv6 Address Registration Procedure . . . . .	6
6.1. DHCPv6 Address Registration Request . . . . .	6
6.2. DHCPv6 Address Registration Acknowledge . . . . .	7
7. Security Considerations . . . . .	7
8. IANA Considerations . . . . .	7
9. Acknowledgements . . . . .	8
10. References . . . . .	8
10.1. Normative References . . . . .	8
10.2. Informative References . . . . .	9
Authors' Addresses . . . . .	9

## 1. Introduction

In several common network scenarios, IPv6 addresses are self-generated by the end-hosts using some information propagated to them by the network (i.e. the network prefix). Examples of self-generated addresses include those created using IPv6 Stateless Address Configuration [RFC4862], temporary addresses [RFC4941] and Cryptographically Generated Addresses (CGA) [RFC3972] etc. These addresses are potentially incompatible with networks with a centrally managed address architecture such as DHCPv6 [RFC3315] as they lack traceability and stability.

Many operators of enterprise networks and similarly tightly administered networks have expressed the desire to be at least aware of the hosts' self-generated addresses when migrating to IPv6.

One potential way to provide network administrators with most of their needs while retaining compatibility with normal stateless configuration would be to register the self-generated addresses with the systems in place to centrally administer the addresses. The edge router that observes hosts' addresses through the Neighbor Discovery protocol is the most suitable devices to register these addresses.

This document introduces a new IPv6 Neighbor Discovery option and a new DHCPv6 option to solicit edge routers to register addresses. The DHCPv6 protocol is used to perform the address registration procedure while the address registration server role may be performed by a DHCPv6 server or a stand-alone server, which is also considered as a DHCPv6 server from the DHCPv6 protocol perspective. A new Address-registration-request DHCPv6 message type is defined to initiate the address registration request, and two new Status codes is defined to indicate registration errors on the server side.

## 2. Terminology

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. Overview of Generic Address Registration Solution

In the generic address registration solution, the network management system solicits the edge routers to register addresses, by sending solicitation messages from either upstream router (step 1a in Figure 1) or DHCPv6 server (step 1b in Figure 1).

After receiving such solicitations, an edge router implementing this specification SHOULD send an Addr-registration-request message to the address registration server (step 2 in Figure 1, defined in Section 5 of this document). The address registration server may be acted by a DHCPv6 server. By received the address registration request, the address registration server records the requested address in the address registration database, which MAY be used by other network functions, such as DNS or ACL, etc. An acknowledgement MAY be sent back to the edge router (step 3 in Figure 1).

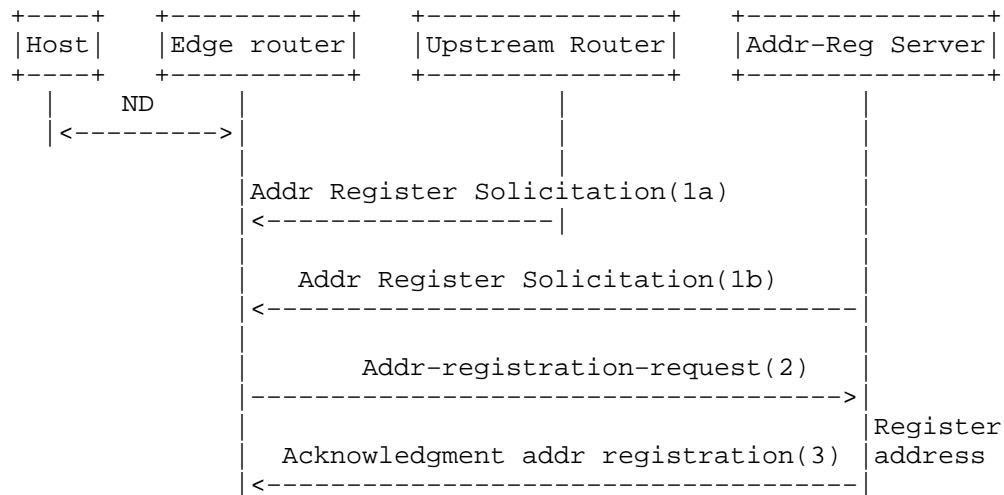


Figure 1: Address Registration Procedure

#### 4. Propagating the Address Registration Solicitation

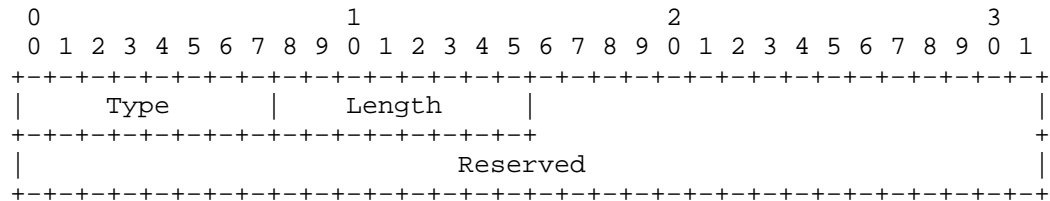
In order to notify the edge routers the availability of the address registration service, new solicitation options are needed. There is more than one mechanism by which configuration parameters could be pushed to the edge routers. The address registration solicitation option can be carried in Router Advertisement (RA) message, which is broadcasted by upstream routers. In the DHCPv6 managed network, it can also be carried in DHCPv6 messages. This document defines a new ND option and a new DHCPv6 option for this purpose. Since the address registration process is through the standard DHCPv6 client/server communication - send packets to ff02::1:2, the All\_DHCP\_Relay\_Agents\_and\_Servers multicast address, these solicitation options do not contain the IP address of address registration server.

After receiving a message containing an address registration solicitation option, an edge router implementing this specification

SHOULD register addresses to the address registration server.

#### 4.1. ND Address Registration Solicitation Option

The ND Address Registration Solicitation Option allows an upstream router to propagate the solicitation for edge routers to register addresses. The format of the ND Address Registration Solicitation Option is described as follows:



Type                   TBA1

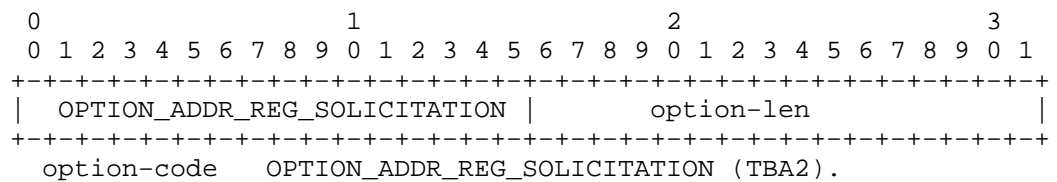
Length                1 (in units of 8 octets, Type and Length themselves are included).

Reserved             Padding bits. For future use also. The value MUST be initialized to zero by the sender, and MUST be ignored by the receiver.

#### ND Address Registration Solicitation Option

#### 4.2. DHCPv6 Address Registration Solicitation Option

The DHCPv6 Address Registration Solicitation Option allows a DHCPv6 server to propagate the solicitation for edge routers to register addresses. This option MAY be propagated together with DHCPv6 Prefix Delegation Option, [RFC3633]. The format of the DHCPv6 Address Registration Solicitation Option is described as follows:

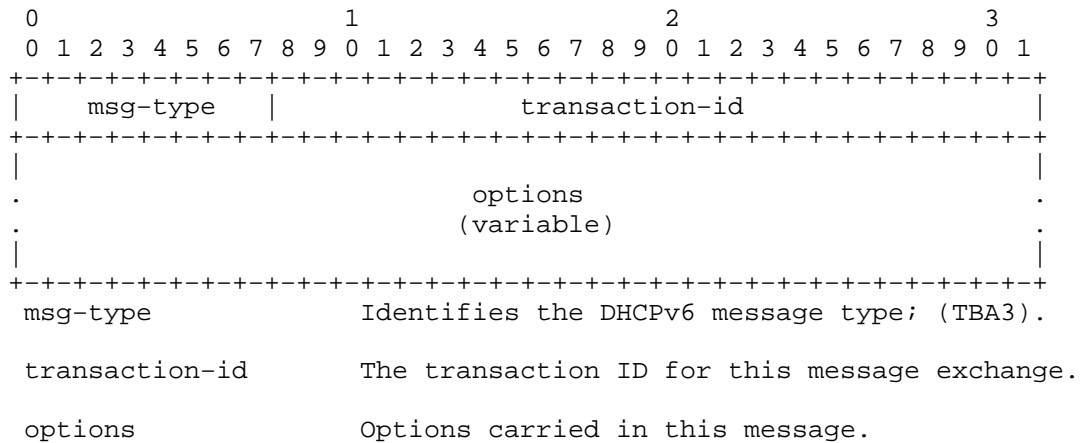


option-len        0, Length of this option in octets (not including option-code and option-len).

#### DHCPv6 Addr Registration Solicitation Option

## 5. DHCPv6 Addr-registration-request Message

A DHCPv6 client (the edge router) sends an Addr-registration-request message to a server to request addresses to be registered. The format of the Addr-registration-request message is described as follows, compliant with Section 6 in [RFC3315]:



DHCPv6 Addr-Registration-Request message

This Addr-registration-request message MUST NOT contain server-identifier option and SHOULD only contain IA\_NA option(s) and Client Identifier option.

Clients MUST discard any received Addr-registration-request messages. Servers MUST discard any Addr-Registration-Request messages that do not include a Client Identifier option or that do include a Server Identifier option.

## 6. DHCPv6 Address Registration Procedure

The DHCPv6 protocol is reused as the address registration protocol while a DHCPv6 server can play the role of an address registration server. The IA\_NA DHCPv6 option [RFC3315] is reused in order to fulfill the address registration interactions.

### 6.1. DHCPv6 Address Registration Request

The edge router sends a DHCPv6 Addr-registration-request message to the address registration server to ff02::1:2, the All\_DHCP\_Relay\_Agents\_and\_Servers multicast address.

The edge router MUST include a Client Identifier option in the Addr-registration-request message to identify itself to the server. The DHCPv6 Addr-registration-request message SHOULD contain at least one IA\_NA option. The IA\_NA option SHOULD contain at least one IA Address option.

After receiving this Addr-Registration-Request message, the address registration server MUST register the requested address(es) in its address registration database, which may further be used by other network functions, such as DNS, network access control lists, etc. If the DHCPv6 server does not support address registration function, a Reply message with includes a Status Code option with the value the RegistrationNotSupported (TBA4) MAY be sent back to the initiated client.

#### 6.2. DHCPv6 Address Registration Acknowledge

After all the addresses have been processed, the address registration server MAY send a Reply message as the response to registration requests. The server generates a Reply message and includes a Status Code option with value Success, a Server Identifier option with the server's DUID, and a Client Identifier option with the client's DUID. For each IA in the Release message for which the server does not register, the server adds an IA option using the IAID from the Addr-registration-request message, and includes a Status Code option with the value RegistrationDenied (TBA5) in the IA option. No other options are included in the IA option.

#### 7. Security Considerations

An attacker may register large number of fake addresses with the network in order to overwhelm the address registration server. These attacks may be prevented generic DHCPv6 protection by using the AUTH option [RFC3315] or Secure DHCPv6 [I-D.ietf-dhc-secure-dhcpv6].

#### 8. IANA Considerations

This document defines a new IPv6 Neighbor Discovery option, the Address Registration Solicitation Option (TBA1) described in Section 4.1, that requires an allocation out of the registry defined at

<http://www.iana.org/assignments/icmpv6-parameters>

This document defines a new DHCPv6 option, the OPTION\_ADDR\_REG\_SOLICITATION (TBA2) described in Section 4.2, that requires an allocation out of the registry defined at

<http://www.iana.org/assignments/dhcpv6-parameters/>

This document defines a new DHCPv6 message, the Addr-registration-request message (TBA3) described in Section 5, that requires an allocation out of the registry defined at

<http://www.iana.org/assignments/dhcpv6-parameters/>

This document defines two new DHCPv6 Status code, the RegistrationNotSupported (TBA4) and RegistrationDenied (TBA5) described in Section 6, that requires an allocation out of the registry defined at

<http://www.iana.org/assignments/dhcpv6-parameters/>

## 9. Acknowledgements

The authors would like to thank Ralph Droms, Ted Lemon, Bernie Volz, Sten Carlsen, Erik Kline, Lorenzo Colitti, Joel Jaeggli, Sten Carlsen, Mark Smith and other members of dhc and v6ops working groups for their valuable comments.

## 10. References

### 10.1. 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.
- [RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.
- [RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, March 2005.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.



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

## 10.2. Informative References

[I-D.ietf-dhc-secure-dhcpv6]  
Jiang, S. and S. Shen, "Secure DHCPv6 Using CGAs",  
draft-ietf-dhc-secure-dhcpv6-07 (work in progress),  
September 2012.

## Authors' Addresses

Sheng Jiang  
Huawei Technologies Co., Ltd  
Q14, Huawei Campus  
No.156 Beiqing Road  
Hai-Dian District, Beijing 100095  
P.R. China

Email: jiangsheng@huawei.com

Gang Chen  
China Mobile  
53A, Xibianmennei Ave., Xuanwu District, Beijing  
P.R. China

Phone: 86-13910710674  
Email: phdgang@gmail.com

Suresh Krishnan  
Ericsson  
8400 Decarie Blvd.  
Town of Mount Royal, QC  
Canada

Phone: +1 514 345 7900 x42871  
Email: suresh.krishnan@ericsson.com



DHC Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: March 15, 2015

S. Jiang  
Huawei Technologies Co., Ltd  
G. Chen  
China Mobile  
S. Krishnan  
Ericsson  
R. Asati  
Cisco Systems, Inc.  
September 11, 2014

Registering Self-generated IPv6 Addresses in DNS using DHCPv6  
draft-ietf-dhc-addr-registration-07

Abstract

In networks that are centrally managed, self-generated addresses cause some traceability issues due to their decentralized nature. One of the most important issues in this regard is the inability to register such addresses in DNS. This document defines a mechanism to register self-generated and statically configured addresses in DNS through a 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 <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 March 15, 2015.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	3
3. Solution Overview . . . . .	3
4. DHCPv6 ADDR-REGISTRATION-REQUEST Message . . . . .	4
5. DHCPv6 Address Registration Procedure . . . . .	5
5.1. DHCPv6 Address Registration Request . . . . .	6
5.2. Registration Expiry and Refresh . . . . .	6
5.3. Acknowledging Registration and Retransmission . . . . .	6
6. Security Considerations . . . . .	7
7. IANA Considerations . . . . .	8
8. Acknowledgements . . . . .	8
9. References . . . . .	8
9.1. Normative References . . . . .	8
9.2. Informative References . . . . .	9
Authors' Addresses . . . . .	9

## 1. Introduction

In several common network scenarios, IPv6 addresses are self-generated by the end-hosts by appending a self-generated interface identifier to a network-specified prefix. Examples of self-generated addresses include those created using IPv6 Stateless Address Configuration [RFC4862] , temporary addresses [RFC4941] and Cryptographically Generated Addresses (CGA) [RFC3972] etc. In several tightly controlled networks, hosts with self-generated addresses may face some limitations. One such limitation is related to the inability of nodes with self-generated addresses to register their IPv6-address-to-FQDN bindings in DNS. This is related to the fact that, in such networks, only certain nodes (e.g. The DHCPv6 server) are allowed to update these bindings in order to prevent end-hosts from registering arbitrary addresses for their FQDNs or associating their addresses with arbitrary domain names. The administrators may not want to distribute the address of authoritative name-server. Also, there is no way to propagate the address of authoritative name server by any protocols. It is preferred that the address registration server, which is under the same management with the authoritative name-server, to know the address of the authoritative name-server and make registration requests on behalf of clients. It is preferred by administrators to

establish and manage one trust relationship between a single DHCPv6 (address registration) server and the DNS authoritative name-server, rather than to distribute and manage trust relationships between many clients and the DNS authoritative name-server.

For nodes that obtain their addresses through DHCPv6, a solution has been specified in [RFC4704]. The solution works by including a Client FQDN option in the SOLICIT, REQUEST, RENEW or REBIND messages during the process of acquiring an address through DHCPv6. This document provides an analogous mechanism to register self-generated addresses in DNS.

A new ADDR-REGISTRATION-REQUEST DHCPv6 message type is defined to initiate the address registration request, and two new Status codes are defined to indicate registration errors on the server side.

## 2. Terminology

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].

**Certificate** In this document, the term "Certificate" is all referred to public key certificate.

## 3. Solution Overview

After successfully assigning a self-generated IPv6 address on one of its interfaces, an end-host implementing this specification SHOULD send an ADDR-REGISTRATION-REQUEST message to a DHCPv6 address registration server. After receiving the address registration request, the DHCPv6 server registers the IPv6 address to FQDN binding towards a configured DNS server. An acknowledgement MUST be sent back to the end host to indicate whether or not the registration operation succeeded.

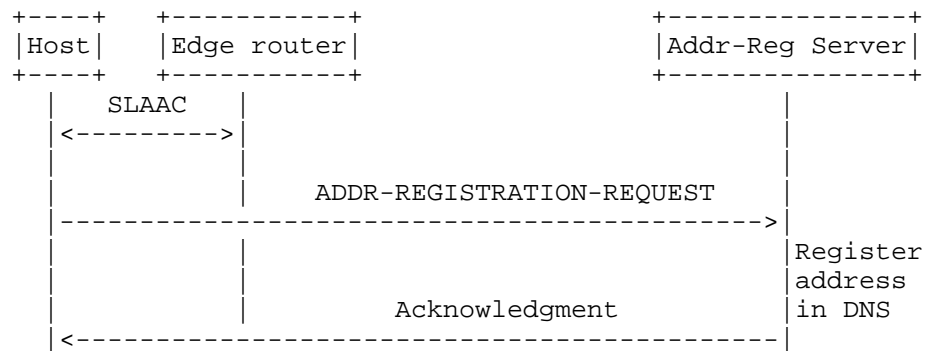


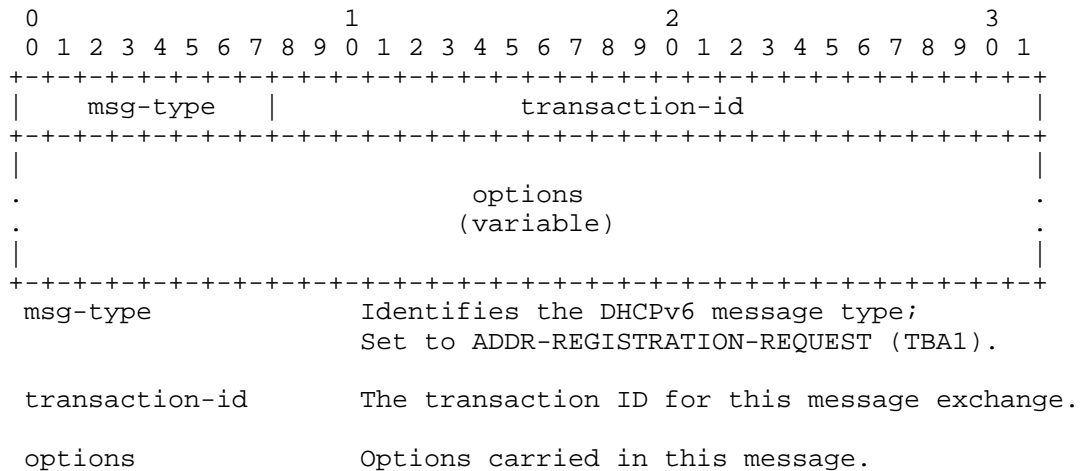
Figure 1: Address Registration Procedure

Furthermore, the registration server MAY apply certain filter/accept criteria for the address registration requests, particularly for the client chosen domain names.

It is RECOMMENDED to only set up one addressregistration server within an administration domain, although there may be multiple DHCPv6 servers. While using multiple address registration servers does potentially increase the load on DNS, because of how [RFC4703] and [RFC4704] work, this should NOT be an issue - the servers should work correctly in updating DNS (either adding or removing the entries). The broken part with multiple servers is the 'extension' of the registration. If there are two address registration servers and both receive the initial registration and (correctly) update DNS, the problem comes when the client extends this but one of the servers does not receive this extension. Then, the server that missed the extension removes the entry prematurely (i.e., when it expired originally).

#### 4. DHCPv6 ADDR-REGISTRATION-REQUEST Message

The DHCPv6 client sends an ADDR-REGISTRATION-REQUEST message to a server to request an address to be registered in the DNS. The format of the ADDR-REGISTRATION-REQUEST message is described as follows:



#### DHCPv6 ADDR-REGISTRATION-REQUEST message

The ADDR-REGISTRATION-REQUEST message MUST NOT contain server-identifier option and MUST contain the IA Address option and the DHCPv6 FQDN option [RFC4704]. The ADDR-REGISTRATION-REQUEST message is dedicated for clients to initiate an address registration request toward an address registration server. Consequently, clients MUST NOT put any Option Request Option(s) in the ADDR-REGISTRATION-REQUEST message.

Clients MUST discard any received ADDR-REGISTRATION-REQUEST messages.

Servers MUST discard any ADDR-REGISTRATION-REQUEST messages that meet any of the following conditions:

- o the message does not include a Client Identifier option;
- o the message includes a Server Identifier option;
- o the message does not include at least one IA Address option;
- o the message does not include FQDN option (or include multiple FQDN options);
- o the message includes an Option Request Option.

#### 5. DHCPv6 Address Registration Procedure

The DHCPv6 protocol is used as the address registration protocol when a DHCPv6 server performs the role of an address registration server. The DHCPv6 IA Address option [RFC3315] and the DHCPv6 FQDN option

[RFC4704] are adopted in order to fulfill the address registration interactions.

#### 5.1. DHCPv6 Address Registration Request

The end-host sends a DHCPv6 ADDR-REGISTRATION-REQUEST message to the address registration server to the All\_DHCP\_Relay\_Agents\_and\_Servers multicast address (ff02::1:2).

The end-host MUST include a Client Identifier option in the ADDR-REGISTRATION-REQUEST message to identify itself to the server. The DHCPv6 ADDR-REGISTRATION-REQUEST message MUST contain at least one IA Address option and exactly one FQDN option. The valid-lifetime field of the IA Address option MUST be set to the period for which the client would like to register the binding in DNS.

After receiving this ADDR-REGISTRATION-REQUEST message, the address registration server MUST register the binding between the provided FQDN and address(es) in DNS. If the DHCPv6 server does not support address registration function, it MUST silently drop the message.

#### 5.2. Registration Expiry and Refresh

For every successful binding registration, the address registration server MUST record the IPv6-address-to-FQDN bindings and associated valid-lifetimes in its storage.

The address registration client MUST refresh the registration before it expires (i.e. before the valid-lifetime of the IA address elapses) by sending a new ADDR-REGISTRATION-REQUEST to the address registration server. If the address registration server does not receive such a refresh after the valid-lifetime has passed, it SHOULD remove the IPv6-address-to-FQDN bindings in DNS, also the local record.

It is RECOMMENDED that clients initiate a refresh at about 85% of the valid-lifetime. Because RAs may periodically 'reset' the valid-lifetime, the refresh timer MUST be independently maintained from the address valid-lifetime. Clients SHOULD set a refresh timer to 85% of the valid-lifetime when they complete a registration operation and only update this timer if 85% of any updated valid-lifetime would be sooner than the timer.

#### 5.3. Acknowledging Registration and Retransmission

After an address registration server accepts an address registration request, it MUST send a Reply message as the response to the client. The acceptance reply only means that the server has taken



responsibility to registry for the client. It may not have actually completed the update yet. The server is responsible to register all the addresses in DNS. The server generates a Reply message and includes a Status Code option with value Success, a Server Identifier option with the server's DUID, and a Client Identifier option with the client's DUID.

If there is no reply received within some interval, the client SHOULD retransmits the message according to section 14 of [RFC3315], using the following parameters:

- o IRT ADDR\_REG\_TIMEOUT
- o MRT ADDR\_REG\_MAX\_RT
- o MRC ADDR\_REG\_MAX\_RC
- o MRD 0

The below presents a table of values used to describe the message transmission behavior of clients and servers:

Parameter	Default	Description
ADDR_REG_TIMEOUT	1 secs	Initial Addr Registration Request timeout
ADDR_REG_MAX_RT	60 secs	Max Addr Registration Request timeout value
ADDR_REG_MAX_RC	5	Max Request retry attempts

For each IA Address option in the ADDR-REGISTRATION-REQUEST message for which the server does not accept its associated registration request, the server adds an IA Address option with the associated IPv6 address, and includes a Status Code option with the value RegistrationDenied (TBA2) in the IA Address option. No other options are included in the IA Address option.

Upon receiving a RegistrationDenied error status code, the client MAY also resend the message following normal retransmission routines defined in [RFC3315] with above parameters. The client MUST wait out the retransmission time before retrying.

## 6. Security Considerations

An attacker may attempt to register large number of addresses in quick succession in order to overwhelm the address registration server. These attacks may be prevented generic DHCPv6 protection by using the AUTH option [RFC3315] or Secure DHCPv6 [I-D.ietf-dhc-sedhcpv6].

## 7. IANA Considerations

This document defines a new DHCPv6 message, the ADDR-REGISTRATION-REQUEST message (TBA1) described in Section 4, that requires an allocation out of the registry of Message Types defined at <http://www.iana.org/assignments/dhcpv6-parameters/>

Value	Description	Reference
TBA1	ADDR-REGISTRATION-REQUEST	this document

This document defines a new DHCPv6 Status code, the RegistrationDenied (TBA2) described in Section 5, that requires an allocation out of the registry of Status Codes defined at <http://www.iana.org/assignments/dhcpv6-parameters/>

Code	Name	Reference
TBA2	RegistrationDenied	this document

## 8. Acknowledgements

The authors would like to thank Ralph Droms, Ted Lemon, Bernie Volz, Sten Carlsen, Erik Kline, Lorenzo Colitti, Joel Jaeggli, Sten Carlsen, Mark Smith, Marcin Siodelski, Darpan Malhotra, Tomek Mrugalski, Jinmei Tatuya and other members of dhc and v6ops working groups for their valuable comments.

## 9. References

### 9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2136] Vixie, P., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 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.
- [RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, March 2005.

- [RFC4703] Stapp, M. and B. Volz, "Resolution of Fully Qualified Domain Name (FQDN) Conflicts among Dynamic Host Configuration Protocol (DHCP) Clients", RFC 4703, October 2006.
- [RFC4704] Volz, B., "The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Client Fully Qualified Domain Name (FQDN) Option", RFC 4704, October 2006.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.
- [RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, September 2007.

## 9.2. Informative References

- [I-D.ietf-dhc-sedhcpv6]  
Jiang, S., Shen, S., Zhang, D., and T. Jinmei, "Secure DHCPv6 with Public Key", draft-ietf-dhc-sedhcpv6-03 (work in progress), June 2014.

## Authors' Addresses

Sheng Jiang  
Huawei Technologies Co., Ltd  
Q14, Huawei Campus  
No.156 Beiqing Road  
Hai-Dian District, Beijing 100095  
P.R. China

Email: jiangsheng@huawei.com

Gang Chen  
China Mobile  
53A, Xibianmennei Ave., Xuanwu District, Beijing  
P.R. China

Phone: 86-13910710674  
Email: phdgang@gmail.com

Suresh Krishnan  
Ericsson  
8400 Decarie Blvd.  
Town of Mount Royal, QC  
Canada

Phone: +1 514 345 7900 x42871  
Email: suresh.krishnan@ericsson.com

Rajiv Asati  
Cisco Systems, Inc.  
7025 Kit Creek road  
Research Triangle Park, NC 27709-4987  
USA

Email: rajiva@cisco.com

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: March 20, 2013

Y. Cui  
P. Wu  
J. Wu  
Tsinghua University  
T. Lemon  
Nominum, Inc.  
September 16, 2012

DHCPv4 over IPv6 Transport  
draft-ietf-dhc-dhcpv4-over-ipv6-05

Abstract

In IPv6 networks, there remains a need to provide IPv4 service for some residual devices. This document describes a mechanism for allocating IPv4 addresses to such devices, using DHCPv4 with an IPv6 transport. It is done by putting a special relay agent function (Client Relay Agent) on the client side, as well as extending the behavior of the server; in the case where DHCP server only supports IPv4 transport, a relay agent is extended to support IPv6 transport (IPv6-Transport Relay Agent) and relay DHCP traffic for the server, with a new Relay Agent Information sub-option added to carry the IPv6 address of the Client Relay Agent.

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 March 20, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

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

## Table of Contents

1. Introduction . . . . .	3
2. Requirements Language . . . . .	3
3. Terminology . . . . .	3
4. Protocol Summary . . . . .	4
5. Client Relay Agent IPv6 Address Sub-option . . . . .	6
6. Client Relay Agent Behavior . . . . .	6
7. IPv6-Transport Server Behavior . . . . .	7
8. IPv6-Transport Relay Agent Behavior . . . . .	8
9. Security Consideration . . . . .	8
10. IANA consideration . . . . .	9
11. Contributors . . . . .	9
12. References . . . . .	9
12.1. Normative References . . . . .	9
12.2. Informative References . . . . .	10
Appendix A. Motivation for selecting this particular solution . .	10
A.1. Configuring IPv4 with DHCPv6 . . . . .	11
A.2. Tunnel DHCPv4 over IPv6 . . . . .	11
A.3. DHCPv4 relayed over IPv6 . . . . .	12
Appendix B. Discussion on One Host Retrieving Multiple Addresses through One CRA . . . . .	12
Authors' Addresses . . . . .	13

## 1. Introduction

DHCPv4 [RFC2131] was not designed with IPv6 in mind: DHCPv4 cannot operate on an IPv6 network. However, as dual-stack networks become a reality, the need arises to allocate IPv4 addresses in an IPv6 environment. To meet this demand, this document extends the DHCPv4 protocol to allow the use of an IPv6 network for transport.

A typical scenario that probably requires this feature is IPv4-over-IPv6 hub and spoke tunnel [RFC4925]. In this scenario, IPv4-over-IPv6 tunnel is used to provide IPv4 connectivity to end users (hosts or end networks) across an IPv6 network. If the IPv4 addresses of the end users are provisioned by the concentrator side, then the provisioning process should be able to cross the IPv6 network. One such tunnel mechanism is demonstrated in [I-D.ietf-software-public-4over6]. DHCPv4 over IPv6 would be a generic solution for this scenario.

## 2. Requirements Language

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

## 3. Terminology

This document makes use of the following terms:

- o DHCPv4: IPv4 Dynamic Host Configuration Protocol [RFC2131].
- o Client Relay Agent(CRA): a special DHCPv4 Relay Agent which relays between DHCPv4 client and DHCPv4 server using an IPv6 network. A CRA either sits on the same, IPv6-accessable host with the DHCPv4 client, or sits on the same link with the host.
- o Host Client Relay Agent(HCRA): a CRA which sits on the same, IPv6-accessible host with the DHCPv4 client.
- o On-Link Client Relay Agent(LCRA): a CRA which sits on the same link with the host that runs DHCPv4 client.
- o IPv6-Transport Server(TSV): a DHCPv4 Server that supports IPv6 transport. TSV listens on IPv6 for incoming DHCPv4 messages, and sends DHCPv4 messages in IPv6 packets.

- o IPv6-Transport Relay Agent(TRA): a DHCPv4 Relay Agent that supports IPv6 transport. TRA sits on a machine which has both IPv6 and IPv4 connectivity, and relays DHCP messages between CRA and regular DHCPv4 server. Unlike CRA, TRA sits on the remote end of IPv6 network, and communicates with DHCPv4 server through IPv4.
- o Client Relay Agent IPv6 Address Sub-option (CRA6ADDR sub-option): a new sub-option of the DHCP Relay Agent Information Option [RFC3046], defined in this document, which is used to carry the IPv6 address of the CRA.

#### 4. Protocol Summary

The scenario for DHCPv4 over IPv6 transport is shown in Figure 1. DHCPv4 clients and DHCPv4 server/relay are separated by an IPv6 network in the middle. DHCP messages between a client and the server/relay cannot naturally be forwarded to each other because they are IPv4 UDP packets, either unicast or broadcast. To bridge this gap, both the client side and the server/relay side must enable DHCPv4 over IPv6 transport. More precisely, they must support delivering and receiving DHCP messages in IPv6 UDP packets and thereby traverse the IPv6 network.

On the client side, a special relay agent called Client Relay Agent is placed on the same host with the client, or on the link of the host. CRA is used to relay DHCP messages from the client to the server, and from the server to the client. CRA sends DHCPv4 messages to the server through unicast IPv6 UDP, and receives unicast IPv6 UDP packets with the DHCPv4 messages from the server. By using CRA, no extension is required on the DHCP client.

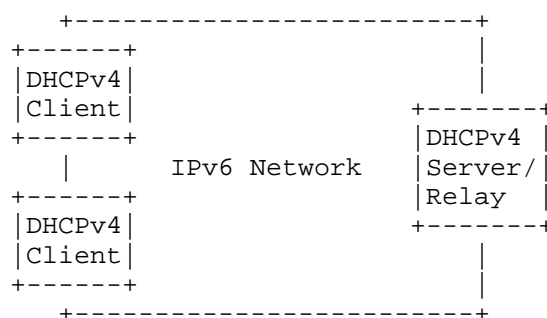


Figure 1 Scenario of DHCPv4 over IPv6 Transport



The IPv6-Transport DHCPv4 server can receive DHCP messages delivered in IPv6 UDP from CRA, and send out DHCP messages to CRA using IPv6 UDP (figure 2(a)). TSV should send DHCP messages to the IPv6 address from which it receives relevant DHCP messages earlier.

When CRAs communicate with an IPv6-Transport Relay Agent rather than with a server directly, the situation becomes a little more complicated. Besides the IPv6 communication with CRA, TRA also communicates with a regular DHCPv4 server through IPv4. Therefore, when TRA relays DHCP messages between a CRA and the DHCPv4 server, it receives DHCP message from the CRA in IPv6 and sends it to the server in IPv4, as well as receives DHCP message from the server in IPv4 and sends it to the CRA in IPv6.

TRA sends the IPv6 address of the CRA to the DHCP server using the Client Relay Agent IPv6 Address suboption, defined in this document. The DHCP server returns this suboption to the TRA as required in [RFC3046]. The TRA then uses the returned CRA6ADDR suboption to determine the destination address to which to relay the response.

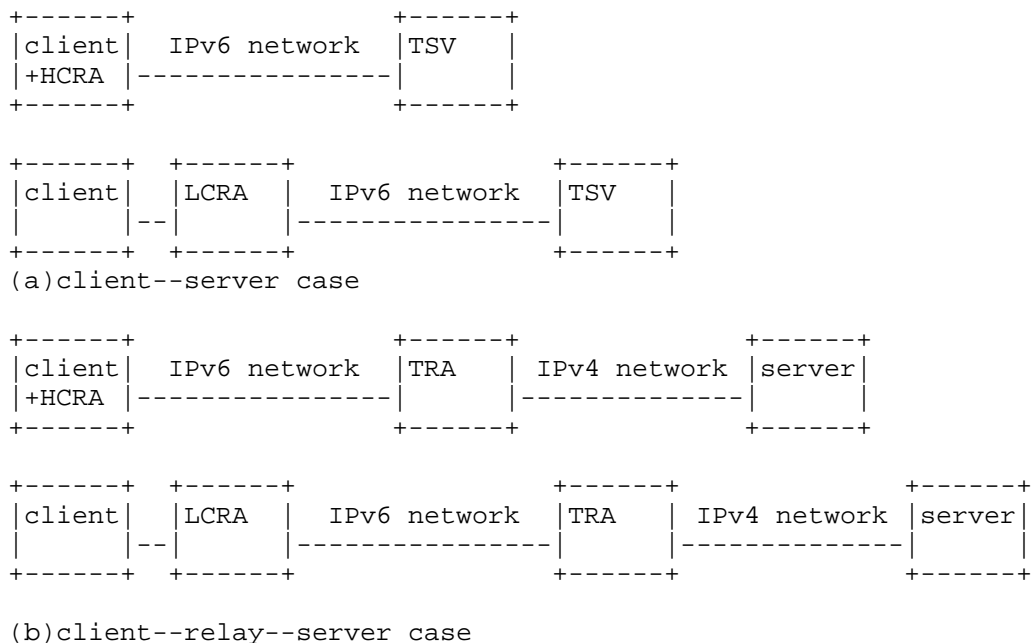


Figure 2 Protocol Summary

## 5. Client Relay Agent IPv6 Address Sub-option

The CRA6ADDR suboption is a suboption of the Relay Agent Information Option [RFC3046]. It encodes the IPv6 address of the machine from which a DHCPv4-in-IPv6 CRA-to-TRA message was received. It is used by the TRA to relay DHCPv4 replies back to the proper CRA. The TRA uses the IPv6 address encoded in this suboption as the destination IPv6 address when relaying a DHCPv4 message from the DHCP server to the CRA.

The CRA6ADDR sub-option has a fixed length of 18 octets. The SubOpt code is tbd by IANA, the length field is 16, and the following 16 octets contain the CRA IPv6 address.

SubOpt	Len	Agent Remote ID					
+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+
tbd	16	a1	a2	a3	...	a16	
+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+

Figure 3 Client Relay Agent IPv6 Address Sub-option format

## 6. Client Relay Agent Behavior

A Client Relay Agent sits on the same host with the DHCPv4 client (HCRA), or on the same link as the host (LCRA). CRA listens for DHCP packets on IPv4 on port 67, and also listens for DHCP packets on IPv6 on port 67.

A CRA is configured with one or more IPv6 addresses of TSV/TRA, using a DHCPv6 option or some other mechanism. The CRA cannot forward DHCPv4 messages before it is configured with an IPv6 address itself, so to function properly, the IPv6 address for the CRA SHOULD be configured before the DHCPv4 client starts. The CRA SHOULD use a global IPv6 address.

When the CRA receives any DHCP message on IPv4 with BOOTP op field set to 1, it forwards the message over UDP on IPv6 using a standard DHCP message format, with source port 67 and destination port 67. The CRA forwards the message to each TSV or TRA address with which it is configured.

When the CRA receives any message on IPv6 with BOOTP op field set to 2, the CRA checks to see if the message contains option 82. If it does, the CRA silently discards the message. Otherwise, it relays the message to the DHCP client using IPv4.

When the CRA receives any message on IPv6 with BOOTP op field set to 4, it decapsulates the message as specified in DHCPv4 Relay Agent Encapsulation [I-D.ietf-dhc-dhcpv4-relay-encapsulation]. If the CRA does not support encapsulation, it MUST silently discard the message.

The LCRA or HCRA MUST NOT use the Relay Agent Information Option [RFC3046]. If either type of CRA needs to send relay agent options, it MUST use relay agent encapsulation as defined in [I-D.ietf-dhc-dhcpv4-relay-encapsulation].

An HCRA MUST only serve the client inside the same host, while the LCRA SHOULD serve any client on the link. When the IPv6 address of TSV/TRA is provisioned to the host running the DHCP client, it uses HCRA; else the client depends on LCRA. A HCRA serves only one link; the multiple link case MUST be handled by multiple HCRA instances. A LCRA does not necessarily need an IPv4 address, though it may be configured with one.

In HCRA case, the DHCPv6 client (or other IPv6 configuration processes), DHCPv4 client and CRA runs on the same physical interface. If possible, the host running the DHCPv4 client and CRA SHOULD defer the operation of the DHCPv4 client until an IPv6 address of the interface has been acquired, as well as the TSV/TRA address information. If this is not done, the DHCPv4 client may send several messages that the CRA cannot relay, and this could result in long delays before the DHCPv4 client actually gets an IPv4 address.

## 7. IPv6-Transport Server Behavior

To support IPv6 transport, the behavior of DHCPv4 server is extended. The IPv6-Transport Server can listen on IPv6 port 67 for DHCPv4 messages, and send DHCPv4 messages through IPv6.

A TSV listens for DHCP messages on IPv6 UDP port 67 and IPv4 UDP port 67. When it receives a DHCP message on IPv6, it MUST retain the IPv6 source address of that message until it has sent a response. When it sends a response, it MUST send the response to this IPv6 address, with destination port 67.

The TSV MUST send a server identifier option [RFC2132] containing an IPv4 address which will be reachable from the client once the residual IPv4 service is set up. This follows the server id option requirement in [RFC2131].

The rest of TSV DHCP process is the same with normal DHCPv4 server. A TSV MUST also listen on IPv4 UDP port 67 like a normal DHCPv4 server, and process IPv4 DHCPv4 messages normally. This requirement

exists because when a DHCPv4 client renews, it sends its renewal messages directly to the server, rather than broadcasting them.

Because the CRA may use relay agent encapsulation [I-D.ietf-dhc-dhcpv4-relay-encapsulation], the TSV SHOULD support it. A TSV that does not support it will not interoperate with a CRA that sends relay agent options.

## 8. IPv6-Transport Relay Agent Behavior

An IPv6-Transport Relay Agent sits between IPv6 network and IPv4 network, and relays DHCPv4 message between CRAs and IPv4-only DHCPv4 server. The communication between CRAs and the TRA uses IPv6, while the communication between the TRA and the server uses IPv4. A TRA listens on IPv6 UDP port 67 for DHCP messages with BOOTP op field set to 1 or 3, as well as IPv4 UDP port 67 for DHCP messages with BOOTP op field set to 2 or 4.

When relaying a DHCP message from CRA to server, TRA MUST add a CRA6ADDR suboption. The TRA sets the contents of this suboption to the IPv6 source address of the message. The TRA MUST also store one its own IPv4 addresses in the giaddr field of the DHCP message. The TRA MAY include a Link Selection sub-option [RFC3527] to indicate to the DHCP server which link to use when choosing an IP address. If the received message is a RELAYFORWARD message, the TRA MUST encapsulate the message in a new RELAYFORWARD message and store the CRA6ADDR in the new relay segment. If it is some other message, the TRA SHOULD append a Relay Agent Information Option as described in [RFC3046], but MAY encapsulate it in the same way as RELAYFORWARD message instead.

When receiving a DHCP message from the DHCP server, if the message contains no CRA6ADDR suboption, the TRA MUST discard the message. Otherwise, it processes it as required by [RFC3046] and [I-D.ietf-dhc-dhcpv4-relay-encapsulation], and forwards it to the IPv6 address recorded in the CRA6ADDR sub-option, with source port 67 and destination port number 67.

## 9. Security Consideration

This mechanism may rise a new form of DHCP protocol attack. A malicious attacker in IPv6 can interference with the DHCPv4 process by inject fake DHCPv4-in-IPv6 messages which will be handled by TSV or TRA. However, the damage is the same with the known DHCPv4 attack happened in IPv4. The only difference is the attacker and the victim could locate in different address families.

Another impact is DHCP filtering. There are firewalls today capable of filtering DHCP traffic (DHCPv4 over IPv4 and DHCPv6 over IPv6 packages). The DHCP messages with the new, DHCPv4-in-IPv6 style may bypass these firewalls. Nevertheless it is not difficult for them to make some slight modification and adapt to the new DHCPv4 message pattern.

## 10. IANA consideration

IANA is requested to assign one new sub-option code from the registry of DHCP Agent Sub-Option Codes maintained in <http://www.iana.org/assignments/bootp-dhcp-parameters>. This sub-option code will be assigned to the Client Relay Agent IPv6 Address Sub-option.

## 11. Contributors

The following gentlemen also contributed to the effort:

Francis Dupont  
Internet Systems Consortium, Inc.

Email: [fdupont@isc.org](mailto:fdupont@isc.org)

Tomasz Mrugalski  
Internet Systems Consortium, Inc.

Email: [tomasz.mrugalski@gmail.com](mailto:tomasz.mrugalski@gmail.com)

Dmitry Anipko  
Microsoft Corporation

Email: [danipko@microsoft.com](mailto:danipko@microsoft.com)

## 12. References

### 12.1. Normative References

[I-D.ietf-dhc-client-id]  
Swamy, N., Halwasia, G., and S. Unit, "Client Identifier Option in DHCP Server Replies",

draft-ietf-dhc-client-id-05 (work in progress),  
September 2012.

- [I-D.ietf-dhc-dhcpv4-relay-encapsulation]  
Lemon, T., Deng, H., and L. Huang, "Relay Agent  
Encapsulation for DHCPv4",  
draft-ietf-dhc-dhcpv4-relay-encapsulation-01 (work in  
progress), July 2011.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate  
Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2131] Droms, R., "Dynamic Host Configuration Protocol",  
RFC 2131, March 1997.
- [RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor  
Extensions", RFC 2132, March 1997.
- [RFC3046] Patrick, M., "DHCP Relay Agent Information Option",  
RFC 3046, January 2001.
- [RFC3527] Kinnear, K., Stapp, M., Johnson, R., and J. Kumarasamy,  
"Link Selection sub-option for the Relay Agent Information  
Option for DHCPv4", RFC 3527, April 2003.
- [RFC4361] Lemon, T. and B. Sommerfeld, "Node-specific Client  
Identifiers for Dynamic Host Configuration Protocol  
Version Four (DHCPv4)", RFC 4361, February 2006.
- [RFC4925] Li, X., Dawkins, S., Ward, D., and A. Durand, "Softwire  
Problem Statement", RFC 4925, July 2007.

## 12.2. Informative References

- [I-D.ietf-softwire-public-4over6]  
Cui, Y., Wu, J., Wu, P., Vautrin, O., and Y. Lee, "Public  
IPv4 over IPv6 Access Network",  
draft-ietf-softwire-public-4over6-03 (work in progress),  
August 2012.

## Appendix A. Motivation for selecting this particular solution

We considered three possible solutions to the problem of configuring  
IPv4 addresses on an IPv6 network.

#### A.1. Configuring IPv4 with DHCPv6

Use DHCPv6 instead of DHCPv4, to provision IPv4-related connectivity. In DHCPv6, the provisioned IPv4 address can be embedded into IPv6 address, or carried within a new option. Along with that, dedicated options are needed to convey IPv4-related information, such as the IPv4 address of DNS server, NTP server, etc. Therefore it will put a certain amount of IPv6-unrelated information into DHCPv6 protocol.

This solution was rejected for two reasons. First, the DHCPv6 protocol does not currently provide a mechanism for recording bindings between IPv4 addresses and DHCPv6 clients. Extending DHCPv6 to provide this functionality would be a substantial change to the existing protocol.

Second, a deliberate choice was made when the DHCPv6 protocol was defined to avoid simply copying existing functionality from DHCPv4. While it is possible, using DHCPv6, to deliver IPv4 addresses as IPv6-encoded IPv4 addresses, it might be necessary to add additional DHCPv6 options simply to support IPv4. These options would then remain in the protocol, long after the need for IPv4 has gone.

By comparison, any extensions to DHCPv4 will naturally be forgotten when DHCPv4 is no longer needed. This means that whatever extensions we make to DHCPv4 to solve the problem, we can stop maintaining as soon as IPv4 is no longer needed.

#### A.2. Tunnel DHCPv4 over IPv6

Use DHCPv4 for configuration, and tunnel DHCPv4-in-IPv4 messages over IPv6. Unlike the previous approach where DHCPv6 is used for both IPv4 and IPv6 connectivity, this approach preserves the separation between IPv4 and IPv6 connectivity information. It maintains the IPv4 service without major modifications to IPv6-related provisioning resources, and sustains DHCPv4 to be the IPv4-related information carrier.

This approach was not chosen because it adds a requirement for DHCPv4 to operate over an IPv4-in-IPv6 tunnel. DHCPv4 clients generally operate on broadcast networks, not on tunnels. To make DHCPv4 operate over a tunnel would require substantial changes to the DHCPv4 client, as well as maintaining a tunnel over which to deliver DHCPv4 traffic.

This also creates a chicken-and-egg problem: how do we set up an IPv4 tunnel when we do not know our IPv4 address? Solutions to these problems were proposed, but they require significant changes to the DHCP client and significant additional work to make a tunnel that

could carry the DHCP packets.

#### A.3. DHCPv4 relayed over IPv6

Use DHCPv4 for configuration, and extend it to use an IPv6 transport for relayed messages. Essentially this involves a single change to the protocol, to allow DHCPv4 servers or relay agents to send and receive packets using an IPv6 transport. No changes are required on the client.

The working group chose this third solution because, of the three, it required the fewest changes to the DHCP protocol, so that it was easiest to specify and easiest to implement.

#### Appendix B. Discussion on One Host Retrieving Multiple Addresses through One CRA

This document is written with the intention of supporting a use case where a single DHCP client is configuring a single tunnel endpoint per physical link. The technique described in this document could be used by a host needing to configure more than one tunnel endpoint on the same physical link, i.e., to retrieve multiple addresses through the same CRA. However, the following additional behavior is REQUIRED to support this case.

DHCP server implementing this specification MUST implement Client Identifier Option in DHCP server replies [I-D.ietf-dhc-client-id].

In general this specification is intended not to require modification of DHCP clients. However, DHCP clients being used to configure multiple tunnel endpoints have to be modified; otherwise there is no way for such DHCP clients to differentiate between DHCP responses. Therefore, in such case, the DHCP client using this specification MUST use a different client identifier for each tunnel endpoint being configured. Such DHCP clients MUST examine the response from the DHCP server and use the client identifier to differentiate between the DHCP client state machines for each tunnel endpoint.

In order to satisfy the requirement that client identifiers be unique, DHCP clients configuring multiple tunnel endpoints MUST implement Node-specific Client Identifiers for DHCPv4 [RFC4361]. Such clients MUST use a different IAID for each tunnel endpoint.

It is assumed here that every client state machine on a multiple-tunnel-endpoint link can hear all the DHCP messages (and subsequently accept the messages intended for it). How this is accomplished is left to the implementor. However, implementations MUST follow this



requirement; otherwise, it will be impossible for multiple tunnel endpoints to be successfully configured. The easiest way to accomplish this is to have a single DHCP client process with multiple DHCP state machines, and to dispatch each DHCP message to the correct DHCP client state machine using the client identifier. However, this is not REQUIRED; any mechanism that results in client state machines receiving the messages that are intended for them will suffice.

#### Authors' Addresses

Yong Cui  
Tsinghua University  
Department of Computer Science, Tsinghua University  
Beijing 100084  
P.R.China

Phone: +86-10-6260-3059  
Email: yong@csnet1.cs.tsinghua.edu.cn

Peng Wu  
Tsinghua University  
Department of Computer Science, Tsinghua University  
Beijing 100084  
P.R.China

Phone: +86-10-6278-5822  
Email: pengwu.thu@gmail.com

Jianping Wu  
Tsinghua University  
Department of Computer Science, Tsinghua University  
Beijing 100084  
P.R.China

Phone: +86-10-6278-5983  
Email: jianping@cernet.edu.cn

Ted Lemon  
Nominum, Inc.  
2000 Seaport Blvd  
Redwood City, CA 94063  
USA

Phone: +1-650-381-6000  
Email: mellon@nominum.com



DHC WG  
Internet-Draft  
Intended status: Informational  
Expires: August 19, 2013

B. Rajtar  
Hrvatski Telekom  
I. Farrer  
Deutsche Telekom AG  
February 15, 2013

Provisioning IPv4 Configuration Over IPv6 Only Networks  
draft-rajtar-dhc-v4configuration-01

Abstract

As IPv6 becomes more widely deployed, some service providers are taking the approach of deploying IPv6 only networks, without dual-stack functionality for IPv4. However, access to IPv4 based services is still an ongoing requirement and approaches such as IPv4-in-IPv6 software tunnels are being developed to meet this need.

In order to provision end-user's hosts with the necessary IPv4 configuration, a number of different mechanisms have been proposed. This memo discusses the benefits and drawbacks of each and recommend a single approach to use for the basis for future work.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 August 19, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

## Table of Contents

1. Introduction . . . . .	3
2. Approaches for Configuring IPv4 Parameters . . . . .	3
2.1. DHCPv4o6 Based Provisioning - Functional Overview . . . . .	4
2.2. DHCPv6 Based Provisioning - Functional Overview . . . . .	4
2.3. DHCPv4oSW Based Provisioning - Functional Overview . . . . .	5
3. Comparison of the Three Approaches . . . . .	5
3.1. DHCPv4o6 Based Provisioning . . . . .	5
3.1.1. Pros . . . . .	5
3.1.2. Cons . . . . .	6
3.2. DHCPv6 Based Provisioning . . . . .	6
3.2.1. Pros . . . . .	6
3.2.2. Cons . . . . .	6
3.3. DHCPv4oSW Based Provisioning . . . . .	7
3.3.1. Pros . . . . .	7
3.3.2. Cons . . . . .	7
4. Conclusion . . . . .	8
5. IANA Considerations . . . . .	8
6. Security Considerations . . . . .	8
7. Acknowledgements . . . . .	8
8. References . . . . .	8
8.1. Normative References . . . . .	8
8.2. Informative References . . . . .	8
Authors' Addresses . . . . .	9

## 1. Introduction

A service provider with an IPv6-only core network must also be able to provide customers with access to the Internet and other services over IPv4. Software based IPv4-in-IPv6 tunneling mechanisms are an obvious example of this, such as the ones described in: [I-D.cui-software-b4-translated-ds-lite], [I-D.ietf-software-map] and [I-D.bfmk-software-unified-cpe].

A general trend here is to distribute NAT functionality and IPv4 address sharing from the centralized tunnel concentrator to the CPE in order to achieve better scalability. This results in a number of configuration parameters needing to be provisioned to the CPE such as the external public IPv4 address and a restricted port-range to use for NAT.

In order to configure customer's devices for software function, a dynamic provisioning mechanism is necessary. In IPv4 only networks, DHCPv4 has often been used to provide configuration, but in an IPv6 only network, DHCPv4 messages cannot be transported.

This document compares three different approaches which have been proposed for resolving this problem.

## 2. Approaches for Configuring IPv4 Parameters

In order to resolve the problem described above, the following approaches for transporting IPv4 configuration parameters have been suggested:

1. Adapt DHCPv4 format messages to be transported over IPv6 as described in [I-D.ietf-dhc-dhcpv4-over-ipv6]. For brevity, this is referred to as DHCPv4o6.
2. Extend DHCPv6 with new options for IPv4 configuration, such as [I-D.mdt-software-map-dhcp-option] describes.
3. Use DHCPv6 as above for external IPv4 address and source port configuration. Use DHCPv4 over IPv4 messages within an IPv6 software for configuring additional parameters. For brevity, this is referred to as DHCPv4oSW.

At the time of writing, working examples of the first two approaches have been developed and successfully tested in several different operators networks. The third approach is still only theoretical.

Each of these approaches are described in more detail underneath.

## 2.1. DHCPv4o6 Based Provisioning - Functional Overview

In order to receive IPv4 configuration parameters, IPv4-only clients initiate and exchange DHCPv4 messages with the DHCPv4 server. In order to adapt this to an IPv6-only network, an existing DHCPv4 client implements a 'Client Relay' (CRA) function, which takes DHCPv4 messages and puts them into UDPv6 and IPv6.

As the mechanism involves unicast based communications, the IPv6 address of the server must be provisioned to the client. A new DHCPv6 option has been defined for this purpose.

The DHCPv4o6 server must either provide an IPv6 interface to the client, or an intermediary 'Transport Relay Agent' device can act as the gateway between the IPv4 and IPv6 domains.

The DHCPv4o6 server needs to be extended to support the new functionality, such as storing the IPv6 address of DHCPv4o6 clients.

This approach currently uses functional elements for ingress and egress of the IPv6-only transport domain--the CRA on the host and the TRA or TSV on the server. As a result, this approach has sometimes been referred to as a tunneling approach. However, relay agent encapsulation is not a tunnel, since it carries only DHCP traffic; it would be more accurate to describe it as an encapsulation.

It is worth noting that there is no technical reason for using relay encapsulation for DHCPv4o6; this approach was taken because the authors of the draft originally imagined that it might be used to provide configuration information for an unmodified DHCPv4 client. However, this turns out not to be a viable approach: in order for this to work, there would have to be IPv4 routing on the local link to which the client is connected. In that case, there's no need for DHCPv4o6.

Given that this is the case, there is no technical reason why DHCPv4o6 can't simply use the IPv6 transport directly, without any relay encapsulation. This would greatly simplify the specification and the implementation, and would still address the requirements stated in this document.

This solution is described in detail in [I-D.ietf-dhc-dhcpv4-over-ipv6].

## 2.2. DHCPv6 Based Provisioning - Functional Overview

In this approach, DHCPv6 would be extended with new DHCPv6 options for configuring IPv4 based functions.

An example of this approach is described in [I-D.mdt-softwire-map-dhcp-option], where a DHCPv6 message is used to convey parameters necessary for IPv4 in IPv6 softwire configuration.

### 2.3. DHCPv4oSW Based Provisioning - Functional Overview

In this approach, the configuration of IPv4 address and source ports (if required) is carried out as described in section 2.2 above. Additional IPv4 configuration parameters are then provisioned using a DHCPv4 messages transported within IPv6 in the softwire in the same manner as any other IPv4 based traffic.

On receipt at the tunnel concentrator (e.g. MAP Border Router or a Lightweight 4over6 lwaFTR), the DHCPv4 message removed from the softwire and forwarded to the DHCPv4 server in the same way as any other IPv4 packet is handled.

As the client is already configured with its external IPv4 address and source ports, the messages exchanged between the DHCPv4 client and server would be strictly DHCPINFORM/DHCPACK messages, for the configuration of additional IPv4 parameters.

For this approach to function, a mechanism for the DHCPv4 client to learn the IPv4 address of the DHCPv4 server is needed. This could be done by defining a well-known IPv4 address for the DHCPv4 server, implementing a DHCPv4 relay function within the tunnel concentrator or other configuration methods.

From a transport perspective, the key difference between this method and DHCPv4o6 (described above) is that here, the DHCPv4 message is put into UDPv4 and IPv4 and then put into the IPv6 softwire, instead of directly placing the DHCPv4 message into UDPv6 and IPv6.

## 3. Comparison of the Three Approaches

The following section of the document provides the pros and cons of the approaches.

### 3.1. DHCPv4o6 Based Provisioning

#### 3.1.1. Pros

1. Once implemented, all existing DHCPv4 options will be be available with no further ongoing development work necessary.
2. IPv4 and IPv6 based provisioning can be separated from each other if required, allowing flexibility in network design.



3. Easy to implement through minor adaptation of existing DHCPv4 client/server code.
4. Simple, in that no additional functional elements are necessary except the DHCPv4o6 client and server. The Transport Relay Agent is completely optional.

#### 3.1.2. Cons

1. More complex, in that there are more new functional elements within the architecture than are necessary in DHCPv6 based provisioning.
2. A new DHCPv6 option is necessary in order to provision the IPv6 address of the DHCPv4 server to the end device.
3. DHCPv4 clients needs to be updated to implement the IPv6 encapsulation and decapsulation function.
4. The DHCPv4 server needs to be updated to implement new DHCPv4o6 functionality.

### 3.2. DHCPv6 Based Provisioning

#### 3.2.1. Pros

1. Simpler, in that no additional functional elements are required except the DHCPv6 client and server.
2. A single protocol is used to deliver configuration information for IPv4 and IPv6.
3. A single provisioning point for all configuration parameters.

#### 3.2.2. Cons

1. Any required DHCPv4 options must be ported to DHCPv6, which will require a large amount of re-development work. All functional elements in the DHCPv6 implementation (clients, servers, relays) would need to be updated for each change.
2. Means that DHCPv4 'legacy' options, which will be of decreasing relevance in the future will remain in DHCPv6 for the lifetime of the protocol.
3. Each time that a DHCPv4 option is ported to DHCPv6, all clients and servers would need to be updated to implement the new option.

4. Does not provide an architecture for keeping IPv4 and IPv6 domains separated.

### 3.3. DHCPv4oSW Based Provisioning

#### 3.3.1. Pros

1. Once implemented, all existing DHCPv4 options will be available with no further ongoing development work necessary.
2. Uses the existing DHCPv4 and DHCPv6 architectures in order to provide IPv4 configuration in an IPv6 only environment.
3. DHCPv4 and DHCPv6 based provisioning can be separated from each other if required, allowing flexibility in network design.

#### 3.3.2. Cons

1. More complex, in that there are more new functional elements within the architecture than are necessary in DHCPv6 based provisioning.
2. IPv4 over IPv6 software approaches which distribute NAT to the CPE and allow for IP address sharing (MAP-E & LW4o6) forbid the use of reserved TCP/UDP ports (e.g. 0-1024). Every DHCPv4 client sharing the same address needs to have a UDP listener running on UDP port 68. To resolve this would require significant rework to either the software mechanisms and/or the DHCPv4 client implementation.
3. From the current specification, DHCPINFORM is not suitable for use over a software. Additional work, such as the development of 'shims' would be necessary
4. The current DHCPINFORM specification has a number of unclear points, such as those described in [I-D.ietf-dhc-dhcpinform-clarify]. Substantial work would be required to resolve this.
5. Links the deployment of IPv4 configuration over IPv6 to a software implementation (e.g. requiring a software concentrator to act as a DHCPv4 relay). Whilst softwares are the only application for this functionality at the moment, this may not always be the case.
6. A new mechanism must be defined in order to provide the DHCPv4 client with the IPv4 address of the DHCPv4 server so that unicast DHCPINFORM messages can be sent.

#### 4. Conclusion

Discussion: This chapter will be updated to reflect the consensus of the DHC Working Group.

#### 5. IANA Considerations

This document makes no request of IANA.

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

#### 6. Security Considerations

#### 7. Acknowledgements

#### 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.

##### 8.2. Informative References

[I-D.bfmk-software-unified-cpe]  
Boucadair, M. and I. Farrer, "Unified IPv4-in-IPv6 Software CPE", draft-bfmk-software-unified-cpe-02 (work in progress), January 2013.

[I-D.cui-software-b4-translated-ds-lite]  
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-cui-software-b4-translated-ds-lite-09 (work in progress), October 2012.

[I-D.ietf-dhc-dhcpinform-clarify]  
Hankins, D., "Dynamic Host Configuration Protocol DHCPINFORM Message Clarifications", draft-ietf-dhc-dhcpinform-clarify-06 (work in progress), October 2011.

[I-D.ietf-dhc-dhcpv4-over-ipv6]  
Cui, Y., Wu, P., Wu, J., and T. Lemon, "DHCPv4 over IPv6

Transport", draft-ietf-dhc-dhcpv4-over-ipv6-05 (work in progress), September 2012.

[I-D.ietf-softwire-map]

Troan, O., Dec, W., Li, X., Bao, C., Matsushima, S., and T. Murakami, "Mapping of Address and Port with Encapsulation (MAP)", draft-ietf-softwire-map-04 (work in progress), February 2013.

[I-D.mdt-softwire-map-dhcp-option]

Mrugalski, T., Troan, O., Bao, C., and W. Dec, "DHCPv6 Options for Mapping of Address and Port", draft-mdt-softwire-map-dhcp-option-03 (work in progress), July 2012.

#### Authors' Addresses

Branimir Rajtar  
Hrvatski Telekom  
Zagreb  
Croatia

Email: [branimir.rajtar@t.ht.hr](mailto:branimir.rajtar@t.ht.hr)

Ian Farrer  
Deutsche Telekom AG  
Bonn  
Germany

Email: [ian.farrer@telekom.de](mailto:ian.farrer@telekom.de)

DHC WG  
Internet-Draft  
Intended status: Informational  
Expires: October 18, 2013

B. Rajtar  
Hrvatski Telekom  
I. Farrer  
Deutsche Telekom AG  
April 16, 2013

Provisioning IPv4 Configuration Over IPv6 Only Networks  
draft-rajtar-dhc-v4configuration-02

Abstract

As IPv6 becomes more widely adopted, some service providers are taking the approach of deploying IPv6 only networks, without dual-stack functionality for IPv4. However, access to IPv4 based services is still an ongoing requirement and approaches such as IPv4-in-IPv6 software tunnels are being developed to meet this need.

In order to provision end-user's hosts with the necessary IPv4 configuration, a number of different mechanisms have been proposed. This memo discusses the benefits and drawbacks of each, with the aim of recommending a single approach as the basis for future work.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 October 18, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

## Table of Contents

1. Introduction . . . . .	2
1.1. Overview of IPv4 Parameter Configuration Approaches . . .	3
1.2. DHCPv4o6 Based Provisioning - Functional Overview . . . .	4
1.3. DHCPv6 Based Provisioning - Functional Overview . . . . .	5
1.4. DHCPv4oSW Based Provisioning - Functional Overview . . . .	5
1.5. DHCPv4oDHCPv6 Based Provisioning - Functional Overview . .	6
2. Requirements for the Solution Evaluation . . . . .	7
3. Comparison of the Four Approaches . . . . .	8
3.1. Pros and Cons of the Different Approaches . . . . .	8
3.1.1. DHCPv4o6 Based Provisioning . . . . .	8
3.1.2. DHCPv6 Based Provisioning . . . . .	9
3.2. DHCPv4oSW Based Provisioning . . . . .	10
3.2.1. Pros . . . . .	10
3.2.2. Cons . . . . .	10
3.3. DHCPv4oDHCPv6 Based Provisioning . . . . .	11
3.3.1. Pros . . . . .	11
3.3.2. Cons . . . . .	11
4. Conclusion . . . . .	11
5. IANA Considerations . . . . .	12
6. Security Considerations . . . . .	12
7. Acknowledgements . . . . .	12
8. References . . . . .	12
8.1. Normative References . . . . .	12
8.2. Informative References . . . . .	12
Authors' Addresses . . . . .	13

## 1. Introduction

A service provider with an IPv6-only network must also be able to provide customers with access to the Internet and other services over IPv4. Software based IPv4-in-IPv6 tunneling mechanisms are an obvious example of this, such as the ones described in:

- o [I-D.ietf-softwire-lw4over6]
- o [I-D.ietf-softwire-map]
- o [I-D.ietf-softwire-unified-cpe]

A general trend here is to relocate NAT44 functionality and IPv4 address sharing from the centralized tunnel concentrator to the CPE in order to achieve better scalability. This results in the need to provision a number of configuration parameters to the CPE, such as the external public IPv4 address and a restricted port-range to use for NAT.

In order to configure customer's devices for softwire functionality, a dynamic provisioning mechanism is necessary. In IPv4 only networks, DHCPv4 has often been used to provide configuration, but in an IPv6 only network, DHCPv4 messages cannot be transported natively.

Although softwire mechanisms are currently the only use-case for dhcp based configuration of IPv4 parameters in IPv6 only networks, a suitable approach must not be limited to only supporting softwire configuration.

This document compares four different approaches which have been proposed for resolving this problem.

#### 1.1. Overview of IPv4 Parameter Configuration Approaches

In order to resolve the problem described above, the following approaches for transporting IPv4 configuration parameters have been suggested:

1. Adapt DHCPv4 format messages to be transported over IPv6 as described in [I-D.ietf-dhc-dhcpv4-over-ipv6]. For brevity, this is referred to as DHCPv4o6.
2. Extend DHCPv6 with new options for IPv4 configuration, such as [I-D.ietf-softwire-map-dhcp] describes.
3. Use DHCPv6 as above for external IPv4 address and source port configuration. Use DHCPv4 over IPv4 messages within an IPv6 softwire for configuring additional parameters. This is referred to as DHCPv4oSW.
4. Use DHCPv4 format messages, transporting them within a new DHCPv6 message type as described in [I-D.scskf-dhc-dhcpv4-over-dhcpv6]. This is referred to as DHCPv4oDHCPv6.

At the time of writing, working examples of the first two approaches have been developed and successfully tested in several different operators networks. The third and fourth methods are still theoretical.

The following sections provide more detail for each approach.

## 1.2. DHCPv4o6 Based Provisioning - Functional Overview

In order to receive IPv4 configuration parameters, IPv4-only clients initiate and exchange DHCPv4 messages with the DHCPv4 server. In order to adapt this to an IPv6-only network, an existing DHCPv4 client implements a 'Client Relay' (CRA) function, which takes DHCPv4 messages and puts them into UDPv6 and IPv6.

As the mechanism involves unicast based communications, the IPv6 address of the server must be provisioned to the client. This option is described in [I-D.mrugalski-softwire-dhcpv4-over-v6-option].

The DHCPv4o6 server must either provide an IPv6 interface to the client, or an intermediary 'Transport Relay Agent' device can act as the gateway between the IPv4 and IPv6 domains.

For the dynamic allocation of IPv4 addresses, the DHCPv4o6 server needs to be extended to support the new functionality, such as storing the IPv6 address of DHCPv4o6 clients. The CRA6ADDR option must also be implemented.

This approach currently uses functional elements for ingress and egress of the IPv6-only transport domain--the CRA on the host and the TRA or TSV on the server. As a result, this approach has sometimes been referred to as a tunneling approach. However, relay agent encapsulation is not a tunnel, since it carries only DHCP traffic; it would be more accurate to describe it as an encapsulation.

It is worth noting that there is no technical reason for using relay encapsulation for DHCPv4o6; this approach was taken because the authors of the draft originally imagined that it might be used to provide configuration information for an unmodified DHCPv4 client. However, this turns out not to be a viable approach: in order for this to work, there would have to be IPv4 routing on the local link to which the client is connected. In that case, there's no need for DHCPv4o6.



Given that this is the case, there is no technical reason why DHCPv4o6 can't simply use the IPv6 transport directly, without any relay encapsulation. This would greatly simplify the specification and the implementation, and would still address the requirements stated in this document.

[I-D.ietf-dhc-dhcpv4-over-ipv6] describes this solution in detail.

The protocol stack is as follows:

DHCPv4/UDPv6/IPv6

### 1.3. DHCPv6 Based Provisioning - Functional Overview

In this approach, DHCPv6 would be extended with new DHCPv6 options for configuring all IPv4 based services and functions. Any DHCPv4 options needed by IPv4 clients connected to the IPV6 network are updated as new DHCPv6 native options carrying IPv4 configuration parameters.

At the time of writing, it is not known how many such options would need to be ported from DHCPv4 to DHCPv6.

An example of this approach is described in [I-D.ietf-softwire-map-dhcp], where a DHCPv6 message is used to convey the parameters necessary for IPv4 in IPv6 softwire configuration.

The protocol stack is as follows:

DHCPv6/UDPv6/IPv6

### 1.4. DHCPv4oSW Based Provisioning - Functional Overview

In this approach, the configuration of IPv4 address and source ports (if required) is carried out using DHCPv6 as described in section 1.3 above. Any additional IPv4 configuration parameters which are required are then provisioned using a DHCPv4 messages transported within IPv6 in the configured softwire in the same manner as any other IPv4 based traffic.

On receipt at the tunnel concentrator (e.g. MAP Border Router or a Lightweight 4over6 lwAFTR), the DHCPv4 message removed from the softwire and forwarded to the DHCPv4 server in the same way as any other IPv4 packet is handled.

As the client is already configured with its external IPv4 address and source ports (using DHCPv6), the messages exchanged between the

DHCPv4 client and server would be strictly DHCPINFORM/DHCPACK messages, for the configuration of additional IPv4 parameters. Broadcast based DHCPDISCOVER messages can not be transported as they are not compatible with the softwire architecture.

For this approach to function, a mechanism for the DHCPv4 client to learn the IPv4 address of the DHCPv4 server is needed. This could be done by defining a well-known IPv4 address for the DHCPv4 server, implementing a DHCPv4 relay function within the tunnel concentrator or other configuration methods.

From a transport perspective, the key difference between this method and DHCPv4o6 (described above) is that here, the DHCPv4 message is put into UDPv4 and IPv4 and then put into the IPv6 softwire, instead of directly placing the DHCPv4 message into UDPv6 and IPv6.

Currently, this approach is only theoretical and does not have a corresponding Internet Draft providing more detail.

The protocol stack that would be used for obtaining additional IPv4 configuraion is as follows:

DHCPv4/UDPv4/IPv4/IPv6

#### 1.5. DHCPv4oDHCPv6 Based Provisioning - Functional Overview

[I-D.scskf-dhc-dhcpv4-over-dhcpv6] describes the transport of DHCPv4 messages within two new DHCPv6 messages types: BOOTREQUESTV6 and BOOTREPLYV6. These messages types must be implemented in both the DHCPv4oDHCPv6 client and server.

In this approach, the configuration of stateless IPv4 addresses and source ports (if required) is carried out using DHCPv6 as described in section 1.3 above. Dynamic IPv4 addressing, and/or any additional IPv4 configuration, is provided using DHCPv4 messages carried (without IPv4/UDPv4 headers) within a new OPTION\_BOOTP\_MSG DHCPv6 option.

OPTION\_BOOTP\_MSG enables the client and server to send BOOTP/DHCPv4 messages verbatim across the IPv6 network. When a DHCPv4oDHCPv6 server receives a DHCPv6 request containing OPTION\_BOOTP\_MSG within a BOOTREQUESTV6 message, it passes it to the DHCPv4 server engine. Likewise, the DHCPv4 server place its DHCPv4 response in the payload of OPTION\_BOOTP\_MSG and puts this into a BOOTPRPLYV6 message.

As the DHCPv4 messages are carried within DHCPv6 multicast messages, using the All\_DHCP\_Relay\_Agents\_and\_Servers, they can be relayed in exactly the same way as any other DHCPv6 multicasted message.

Optionally, DHCPv6 relays could be updated so that they forward the BOOTREQUESTV6 message to a different destination address, allowing for the separation of DHCPv4 and DHCPv4 provisioning infrastructure.

The protocol stack used for obtaining dynamic v4 addressing or additional IPv4 configuraion is as follows:

DHCPv4/DHCPv6/UDPv6/IPv6

## 2. Requirements for the Solution Evaluation

The following requirements have been defined for the evaluation of the different approaches:

1. Minimize the amount of work necessary to implement the solution through re-use of existing standards and implementations as much as possible.
2. Provide a method of supporting all existing DHCPv4 options so that they can be utilised without the need for further standardation.
3. Allow for the dynamic leasing of IPv4 addresses to clients. This allows for more efficient use of limited IPv4 resources.
4. Enable the separation of IPv4 and IPv6 host configuration.
5. Avoid leaving legacy IPv4 options in DHCPv6.
6. Provide a flexible architecture to give operators the option of only deploying the functional elements necessary for their specific requirements.

### 3. Comparison of the Four Approaches

The table below shows a comparison of the different approaches against the solution requirements described above.

Req. No.	DHCPv4o6	DHCPv6	DHCPv4oSW	DHCPv4oDHCPv6
1	No	Yes	Yes	Yes
2	Yes	No	Yes	Yes
3	Yes	No	No	Yes
4	Yes	No	Yes	Yes
5	Yes	No	Yes	Yes
6	Yes	No	Yes	Yes

Table 1: Approach Comparison

#### 3.1. Pros and Cons of the Different Approaches

The following sections of the document provide more details of the pros and cons relevant to each of the approaches.

##### 3.1.1. DHCPv4o6 Based Provisioning

###### 3.1.1.1. Pros

1. Once implemented, all existing DHCPv4 options will be available with no further ongoing development work necessary.
2. IPv4 and IPv6 based provisioning can be separated from each other if required, allowing flexibility in network design.
3. Easy to implement through minor adaptation of existing DHCPv4 client/server code.
4. Simple, in that no additional functional elements are necessary except the DHCPv4o6 client and server. The Transport Relay Agent is completely optional.
5. Suitable for the provisioning of dynamic IPv4 configuration as the existing DHCPv4 leasing mechanism can be used.
6. Implementations already exist, proving that the approach works.

###### 3.1.1.2. Cons

1. More complex, in that there are more new functional elements (CRA, DHCPv4o6 server and optionally TRA) within the architecture than are necessary in DHCPv6 based provisioning.
2. A new DHCPv6 option is necessary in order to provision the IPv6 address of the DHCPv4 server to the end device.
3. For a Host CRA (HCRA), DHCPv4 client host needs to be updated to implement the IPv6 encapsulation and decapsulation function. Otherwise a physically separate On-Link CRA (LCRA) functional element must be deployed.
4. A DHCPv4 server must be deployed and maintained.
5. The DHCPv4 server needs to be updated to implement new DHCPv4o6 functionality.

#### 3.1.2. DHCPv6 Based Provisioning

##### 3.1.2.1. Pros

1. Simpler, in that no additional functional elements are required except the DHCPv6 client and server.
2. A single protocol is used to deliver configuration information for IPv4 and IPv6.
3. A single provisioning point for all configuration parameters.
4. Implementations already exist, proving that the approach works.

##### 3.1.2.2. Cons

1. Any required DHCPv4 options must be ported to DHCPv6, which will require re-development work for each option. All functional elements in the DHCPv6 implementation (clients, servers, relays) would need to be updated for each change.
2. Means that DHCPv4 'legacy' options, which will be of decreasing relevance in the future will remain in DHCPv6 for the lifetime of the protocol.
3. Each time that a DHCPv4 option is ported to DHCPv6, all clients and servers would need to be updated to implement the new option.
4. Does not provide an architecture for keeping IPv4 and IPv6 domains separated.

5. Does not provide a mechanism for dynamic IPv4 address leasing. A DHCPv4 lease lifetime mechanism would need to be added to DHCPv6 for this.

### 3.2. DHCPv4oSW Based Provisioning

#### 3.2.1. Pros

1. Once implemented, all existing DHCPv4 options will be available with no further ongoing development work necessary.
2. Uses the existing DHCPv4 and DHCPv6 architectures in order to provide IPv4 configuration in an IPv6 only environment.
3. DHCPv4 and DHCPv6 based provisioning can be separated from each other if required, allowing flexibility in network design.

#### 3.2.2. Cons

1. More complex, in that there are more new functional elements within the architecture than are necessary in DHCPv6 based provisioning.
2. IPv4 over IPv6 software approaches which distribute NAT to the CPE and allow for IP address sharing (MAP-E & LW4o6) forbid the use of reserved TCP/UDP ports (e.g. 0-1024). Every DHCPv4 client sharing the same address needs to have a UDP listener running on UDP port 68. To resolve this would require significant rework to either the software mechanisms and/or the DHCPv4 client implementation.
3. From the current specification, DHCPINFORM is not suitable for use over a software. Additional work, such as the development of 'shims' would be necessary.
4. The current DHCPINFORM specification has a number of unclear points, such as those described in [I-D.ietf-dhc-dhcpinform-clarify]. Substantial work would be required to resolve this.
5. Links the deployment of IPv4 configuration over IPv6 to a software implementation (e.g. requiring a software concentrator to act as a DHCPv4 relay). Whilst softwares are the only application for this functionality at the moment, this may not always be the case.

6. A new mechanism must be defined in order to provide the DHCPv4 client with the IPv4 address of the DHCPv4 server so that unicast DHCPINFORM messages can be sent.
7. As only DHCPINFORM/DHCPACK DHCPv4 message types are supported, dynamic IPv4 address leasing (using DHCPDISCOVER messages) can not be used.
8. The approach is unproven as no existing implementations exist.

### 3.3. DHCPv4oDHCPv6 Based Provisioning

#### 3.3.1. Pros

1. Once implemented, all existing DHCPv4 options will be available with no further ongoing development work necessary.
2. Uses the existing DHCPv4 and DHCPv6 architectures in order to provide IPv4 configuration in an IPv6 only environment.
3. DHCPv4 and DHCPv6 based provisioning can be separated from each other if required, allowing flexibility in network design.
4. Suitable for the provisioning of dynamic IPv4 configuration as the existing DHCPv4 leasing mechanism can be used.

#### 3.3.2. Cons

1. More complex, in that there are more new functional elements within the architecture than are necessary in DHCPv6 based provisioning.
2. DHCPv6 clients needs to be updated to implement the new DHCPv6 message types.
3. The DHCPv6 server needs to be updated to implement new DHCPv4oDHCPv6 message types and functionality.
4. If separation of DHCPv4 and DHCPv4 provisioning infrastructure is required, DHCPv6 relay agents need to be updated to implement dedicated forwarding destinations based on message type.
5. The approach is currently unproven as no existing implementations exist.

### 4. Conclusion

Discussion: This chapter will be updated to reflect the consensus of the DHC Working Group.

## 5. IANA Considerations

This document makes no request of IANA.

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

## 6. Security Considerations

## 7. Acknowledgements

Thanks to Ted Lemon and Tomek Mrugalski for their input and reviews.

## 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.

### 8.2. Informative References

[I-D.ietf-dhc-dhcpinform-clarify]  
Hankins, D., "Dynamic Host Configuration Protocol DHCPINFORM Message Clarifications", draft-ietf-dhc-dhcpinform-clarify-06 (work in progress), October 2011.

[I-D.ietf-dhc-dhcpv4-over-ipv6]  
Cui, Y., Wu, P., Wu, J., and T. Lemon, "DHCPv4 over IPv6 Transport", draft-ietf-dhc-dhcpv4-over-ipv6-06 (work in progress), March 2013.

[I-D.ietf-softwire-lw4over6]  
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-00 (work in progress), April 2013.

[I-D.ietf-softwire-map-dhcp]  
Mrugalski, T., Troan, O., Dec, W., Bao, C., leaf.yeh.sdo@gmail.com, l., and X. Deng, "DHCPv6 Options for Mapping of Address and Port", draft-ietf-softwire-map-dhcp-03 (work in progress), February 2013.

[I-D.ietf-softwire-map]



Troan, O., Dec, W., Li, X., Bao, C., Matsushima, S., and T. Murakami, "Mapping of Address and Port with Encapsulation (MAP)", draft-ietf-softwire-map-05 (work in progress), March 2013.

[I-D.ietf-softwire-unified-cpe]

Boucadair, M. and I. Farrer, "Unified IPv4-in-IPv6 Softwire CPE", draft-ietf-softwire-unified-cpe-00 (work in progress), March 2013.

[I-D.mrugalski-softwire-dhcpv4-over-v6-option]

Mrugalski, T. and P. Wu, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for DHCPv4 over IPv6 Endpoint", draft-mrugalski-softwire-dhcpv4-over-v6-option-01 (work in progress), September 2012.

[I-D.scskf-dhc-dhcpv4-over-dhcpv6]

Sun, Q., Cui, Y., Siodelski, M., Krishnan, S., and I. Farrer, "DHCPv4 over DHCPv6 Transport", draft-scskf-dhc-dhcpv4-over-dhcpv6-01 (work in progress), April 2013.

#### Authors' Addresses

Branimir Rajtar  
Hrvatski Telekom  
Zagreb  
Croatia

Email: [branimir.rajtar@t.ht.hr](mailto:branimir.rajtar@t.ht.hr)

Ian Farrer  
Deutsche Telekom AG  
Bonn  
Germany

Email: [ian.farrer@telekom.de](mailto:ian.farrer@telekom.de)

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: August 22, 2013

Q. Sun  
Y. Cui  
Tsinghua University  
February 18, 2013

Dynamic Host Configuration Protocol version 6 (DHCPv6) Option for IPv4  
Configuration  
draft-sun-dhc-dhcpv6-opt-v4config-00

Abstract

This document defines a DHCPv6 option with two types of sub-options for IPv4 configurations in the case of IPv4/IPv6 transition. One is used for the assignment of IPv4 address and port set, the other is used for configuring existing DHCPv4 options required by clients for IPv4-over-IPv6 communications.

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 August 22, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

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

described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Requirements Language . . . . .	3
3. DHCPv6 Option for IPv4 Configuration . . . . .	3
3.1. Option Format . . . . .	3
3.2. Shared IPv4 address Sub-Option . . . . .	4
3.3. Sub-Option for Conveying Existing DHCPv4 Options . . . . .	5
4. Server Behavior . . . . .	6
5. Client Behavior . . . . .	6
6. Security Consideration . . . . .	6
6.1. Denial-of-Service . . . . .	6
7. IANA Consideration . . . . .	6
8. References . . . . .	7
8.1. Normative References . . . . .	7
8.2. Informative References . . . . .	7

## 1. Introduction

During the IPv4/IPv6 transition period, IPv4 and IPv6 will coexist for a period of time. There are still requirements for visiting IPv4 services. In order to continue IPv4 communications across IPv6-only access network, IPv4 information is needed to be configured across IPv6 networks. On the one hand, IPv4 address has run out, which raise requirements for address-sharing. On the other hand, some of the existing DHCPv4 options are likely to be configured over IPv6 to guarantee success of some IPv4 services.

To deal with the issues, [I-D.ietf-dhc-dhcpv4-over-ipv6] provides a clean solution that extends DHCPv4 over IPv6 transport to support IPv4 resources allocation and all DHCPv4 options natively. For circumstances that there are only DHCPv6 servers deployed, this document proposes a mechanism that introduces new DHCPv6 option for IPv4 configurations.

This proposal describes a new DHCPv6 option and two types of sub-options which allow the DHCPv6 server to assign a shared IPv4 address and optionally some demanded DHCPv4 options during the IPv6 address provisioning process. By assigning the same IPv4 address with non-overlaped port sets to multiple clients, the clients can share the IPv4 address and continue to deliver IPv4 services to subscribers.

The IPv4 Configuration Option described in this document can be used in various deployment scenarios, some of which are described in [RFC6346]

## 2. Requirements Language

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

## 3. DHCPv6 Option for IPv4 Configuration

### 3.1. Option Format

The IPv4 Configuration DHCPv6 Option consists of two types of sub-options, one for shared IPv4 address and the other for importing existing DHCPv4 options. The SUB\_OPT\_SHARRED\_ADDR MUST be conveyed by the IPv4 Configuration Option while the SUB\_OPT\_v4OPT MAY be conveyed if necessary. The format of IPv4 Configuration DHCPv6 Option is shown in Figure 1.

0										1										2										3									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1								

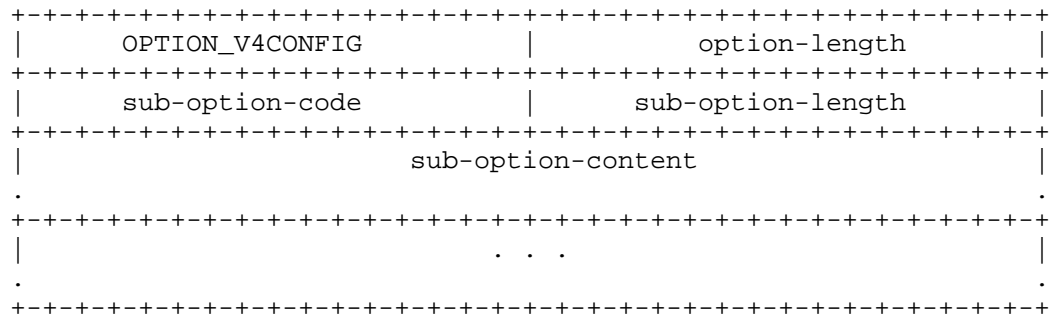


Figure 1 IPv4 Configuration Option Format

- o option-code: OPTION\_V4CONFIG (TBD)
- o option-length: This field indicating the length of the option excluding the 'Option Code' and the 'Option Length' fields. In this option, the option-length is variable, with value of no less than 12 octets.
- o sub-option-code: Specify the code of sub-option, which should be either SUB\_OPT\_SHARRED\_ADDR or SUB\_OPT\_v4OPT.
- o sub-option-length: Length of sub-option.
- o sub-option-content: The content of enclosed sub-option.

### 3.2. Shared IPv4 address Sub-Option

This sub-option is defined for a shared IPv4 address assignment. The sub-option format is shown in the following figure.

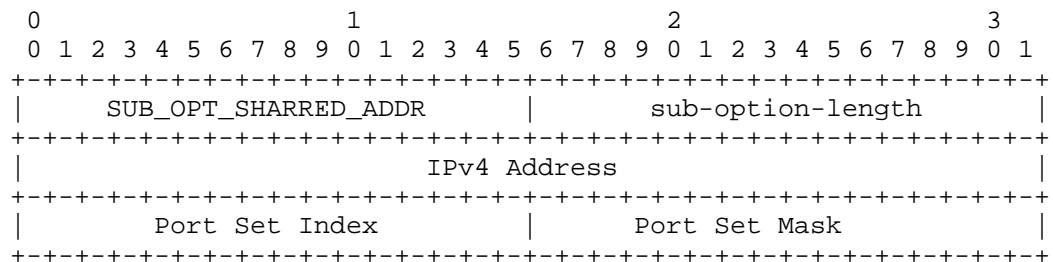


Figure 2 Shared-IPv4 Address Sub-Option Format

- o sub-option-code: SUB\_OPT\_SHARRED\_ADDR (TBD)

- o sub-option-length: The length this option is 8.
- o Port Set Index: Port Set Index identifies a set of ports assigned to a device. The first k bits on the left of the 2-octet field is the Port Set Index value, with the rest of the field right padding zeros.
- o Port Set Mask: Port Set Mask indicates the position of the bits used to build the mask. The first k bits on the left is padding ones while the remained (16-k) bits of the 2-octet field on the right is padding zeros.

In the context of this sub-option, the port number should consist of port set prefix and port number suffix. The port set prefix can be got from Port Set Index and Port Set Mask, while port number suffix can change continuously. The format of port number is shown in Figure 2.

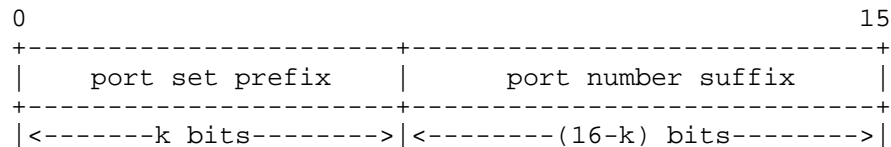


Figure 3 Bit Representation of a port number

In order to exclude the system ports ([I-D.ietf-tsvwg-iana-ports]) or ports saved by SPs, the former port-sets that contains well-known ports SHOULD NOT be assigned.

For example: If k is 10 (the left 10 bits of Port Set Mask is '1'), the first 16 port sets is located in well-known port space, which should not be allocated. Or,

For example: If k is 4 (the left 4 bits of Port Set Mask is '1'), the first port set (0 - 4095) contains the well-know port space. It should be perceived as well.

### 3.3. Sub-Option for Conveying Existing DHCPv4 Options

This sub-option is used for the cases that some of the existing DHCPv4 options are needed to be provisioned to the end users. The existing DHCPv4 options can be put in with original formats remained. This sub-option MUST NOT appear in OPTION\_V4CONFIG if SUB\_OPT\_SHARED\_ADDR is not conveyed. The sub-option format is as follows.

```

0                                     1                                     2                                     3

```

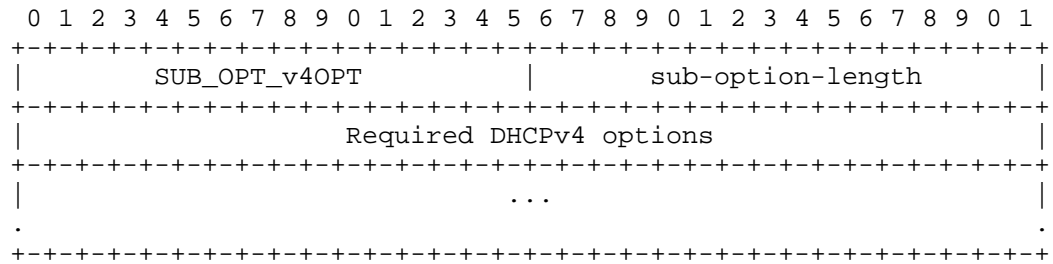


Figure 4 Format of Sub-Option Containing DHCPv4 Options

- o sub-option-code: SUB\_OPT\_v4OPT (TBD)
- o sub-option-length: The length is variable.
- o Required DHCPv4 options: Required DHCPv4 options can be put in this field one by one. The format of DHCPv4 options will not change.

#### 4. Server Behavior

TBD.

#### 5. Client Behavior

TBD

#### 6. Security Consideration

##### 6.1. Denial-of-Service

The solution is generally vulnerable to DoS when used in shared medium or when access network authentication is not a prerequisite to IP address assignment. The solution SHOULD only be used on point-to-point links, tunnels, and/or in environments where authentication at link layer is performed before IP address assignment, and not shared medium.

#### 7. IANA Consideration

IANA is kindly requested to allocate DHCPv6 option code to the OPTION\_V4CONFIG, DHCPv6 sub-option codes to the SUB\_OPT\_SHARRED\_ADDR and SUB\_OPT\_v4OPT. The code should be added to the DHCPv6 option code space.

#### 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.
- [RFC3046] Patrick, M., "DHCP Relay Agent Information Option", RFC 3046, January 2001.
- [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.
- [RFC3527] Kinnear, K., Stapp, M., Johnson, R., and J. Kumarasamy, "Link Selection sub-option for the Relay Agent Information Option for DHCPv4", RFC 3527, April 2003.
- [RFC4925] Li, X., Dawkins, S., Ward, D., and A. Durand, "Softwire Problem Statement", RFC 4925, July 2007.
- [RFC5961] Ramaiah, A., Stewart, R., and M. Dalal, "Improving TCP's Robustness to Blind In-Window Attacks", RFC 5961, August 2010.
- [RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-Protocol Port Randomization", BCP 156, RFC 6056, January 2011.
- [RFC6346] Bush, R., "The Address plus Port (A+P) Approach to the IPv4 Address Shortage", RFC 6346, August 2011.

## 8.2. Informative References

- [I-D.bajko-pripaddrassign] Bajko, G., Savolainen, T., Boucadair, M., and P. Levis, "Port Restricted IP Address Assignment", draft-bajko-pripaddrassign-04 (work in progress), April 2012.



- [I-D.ietf-dhc-dhcpv4-over-ipv6]      Cui, Y., Wu, P., Wu, J., and T. Lemon, "DHCPv4 over IPv6 Transport", draft-ietf-dhc-dhcpv4-over-ipv6-05 (work in progress), September 2012.
- [I-D.ietf-tsvwg-iana-ports]      Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", draft-ietf-tsvwg-iana-ports-10 (work in progress), February 2011.
- [I-D.sun-dhc-port-set-option]      Sun, Q., Lee, Y., Sun, Q., Bajko, G., and M. Boucadair, "Dynamic Host Configuration Protocol (DHCP) Option for Port Set Assignment", draft-sun-dhc-port-set-option-00 (work in progress), October 2012.
- [I-D.vixie-dnsext-dns0x20]      Vixie, P. and D. Dagon, "Use of Bit 0x20 in DNS Labels to Improve Transaction Identity", draft-vixie-dnsext-dns0x20-00 (work in progress), March 2008.

Authors' Addresses

Qi Sun  
Tsinghua University  
Department of Computer Science, Tsinghua University  
Beijing 100084  
P.R.China

Phone: +86-10-6278-5822  
EMail: sunqi@csnet1.cs.tsinghua.edu.cn

Yong Cui  
Tsinghua University  
Department of Computer Science, Tsinghua University  
Beijing 100084  
P.R.China

Phone: +86-10-62603059  
EMail: yong@csnet1.cs.tsinghua.edu.cn

