dhc   T. Lemon
Internet-Draft   Nominum, Inc.
Intended status: Standards Track   H. Deng
Expires: January 12, 2012   L. Huang
                                                         China Mobile
                                                         July 11, 2011

Relay Agent Encapsulation for DHCPv4
draft-ietf-dhc-dhcpv4-relay-encapsulation-01

Abstract

   This document describes a general mechanism whereby DHCP relay agents
   can encapsulate DHCP packets that they are forwarding in the
   direction of DHCP servers, and decapsulate packets that they are
   forwarding toward DHCP clients, so that more than one relay agent can
   insert relay agent suboptions into the forwarding chain.

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 12, 2012.

Table of Contents

1. Introduction

   In some networking environments, it is useful to be able to configure
   relay agents in a hierarchy, so that information from a relay agent
   close to the client can be combined with information from one or more
   relay agents closer to the server, particularly in cases where the
   relay agents may be in separate administrative domains.

   The current Relay Agent Information Option (RAIO) specification
   [RFC3046] specifies that when a relay agent receives a packet
   containing an RAIO, it must not add an RAIO. This prevents chaining
   of RAIOs, and hence prohibits the hierarchical use case.

   DHCP version 6 [RFC3315] provides a much cleaner technique for
   providing RAIO suboptions to the DHCP server. Rather than appending
   an information option to the DHCP client's message, the relay agent
   encapsulates the DHCP client message in a new DHCP message which it
   sends to the DHCP server along with any options it chooses to add to
   provide information to the DHCP server.

   This document specifies a mechanism for providing the same
   functionality in DHCPv4.

1.1. 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].

1.2. Terminology

   The following terms and acronyms are used in this document:

   BOOTREPLY message   Any DHCP or BOOTP message in which the 'op'
                          field is set to BOOTREPLY.

   BOOTREQUEST message   Any DHCP or BOOTP message in which the 'op'
                          field is set to BOOTREQUEST.

   DHCP   Dynamic Host Configuration Protocol Version 4
                          [RFC2131]

   decapsulate   the act of de-encapsulating DHCP packets being
                          relayed from DHCP servers or relay agents in
                          the direction of DHCP clients, according to
                          this specification.

encapsulate   the act of encapsulating DHCP packets that are
                       being relayed from DHCP clients or relay agents
                       toward DHCP servers, according to the method
                       described in this specification.

encapsulating relay agent a relay agent that implements this
                       specification and is configured to encapsulate.

L2RA   Layer 2 Relay Agent--a relay agent that doesn't
                       have an IP address reachable by the DHCP
                       server.

L3RA   Layer 3 Relay Agent--a relay agent that has an
                       IP address reachable by the DHCP server.

option buffer   the portion of the DHCP packet following the
                       magic cookie in the 'vend' field of the DHCP
                       Packet.

RAIO   Relay Agent Information Option [RFC3046]. Also
                       commonly referred to as "Option 82."

RAIO suboption   a DHCP suboption that has been defined for
                       encapsulation in the Relay Agent Information
                       Option

relay message   a RELAYFORWARD or RELAYREPLY message.

RELAYFORWARD message   Any DHCP or BOOTP message in which the 'op'
                       field is set to RELAYFORWARD.

RELAYREPLY message   Any DHCP or BOOTP message in which the 'op'
                       field is set to RELAYREPLY.

silently discard   in many places in this specification, the
                       implementation is required to silently discard
                       erroneous messages. What is meant by 'silently
                       discard' is that the implementation MUST NOT
                       send any ICMP message indicating that the
                       delivery was in error. It may be desirable for
                       the implementation to keep a count of messages
                       that have been discarded, either by message
                       type or by reason for discarding, or some
                       combination. Nothing in this specification
                       should be construed to forbid such data
                       collection.

2. **Protocol Summary**

   This document specifies two new BOOTP message types: the RELAYFORWARD
   message, and the RELAYREPLY message. These messages are analogous to
   the Relay Forward and Relay Reply messages in DHCPv6 [RFC3315].

   Although this specification is generally aimed at DHCP
   implementations, it is not specifically restricted to DHCP, and is
   applicable to BOOTP in cases where the BOOTP server is a DHCP server
   that implements this specification, or the less likely case that the
   BOOTP server only supports the BOOTP protocol, but still implements
   this specification.

   In general, when the term "DHCP" appears in this specification, the
   reader should not read this as intending to exclude BOOTP.

2.1. **RELAYFORWARD Message**

   Conforming relay agents encapsulate messages being sent toward DHCP
   servers in RELAYFORWARD messages.

2.2. **RELAYREPLY Message**

   A conforming DHCP server encapsulates any message being sent toward a
   DHCP client in a RELAYREPLY message, if the message being responded
   to was encapsulated.

   A conforming relay agent, when it receives a RELAYREPLY message,
   decapsulates the message contained in the RELAYREPLY message and
   sends it to the next relay or to the client.

2.3. **Layer Two Address suboption**

   In cases where the closest relay agent to the client is an L2RA, but
   where there is an L3RA on the path to the client, the DHCP server
   will encode the link layer address that would normally go in the
   chaddr field of the DHCP packet into a Layer Two Address suboption.

```
        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
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | SUBOPT_L2AS |   length    |    htype    |    chaddr ...
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   The Layer Two Address suboption has four fields:

SUBOPT_L2AS One octet: the suboption code for the Layer Two Address
   suboption (TBD).

length One octet: the length of the Layer Two Address suboption.

htype One octet: the type of the Layer Two Address suboption--
   corresponds to the 'htype' field in a non-relay DHCP or BOOTP
   message.

chaddr One or more octets: the layer two address of the client, from
   the 'chaddr' field of the DHCP or BOOTP message.

[3]. Encoding

RELAYFORWARD and RELAYREPLY messages are distinguished through the
use of the 'op' field of the DHCP packet.

In non-relay DHCP packets, the 'op' field either contains
BOOTREQUEST, for any DHCP message from the client to the server, or
BOOTREPLY, for any DHCP message from the server to the client.

This document defines two additional codes, RELAYFORWARD and
RELAYREPLY. Conforming DHCP servers and DHCP relay agents MUST
support these two new values for the 'op' field. DHCP clients should
never see either value.

```
            +------+--------------+
            | code | meaning      |
            +------+--------------+
            | 1    | BOOTREQUEST  |
            | 2    | BOOTREPLY    |
            | 3    | RELAYFORWARD |
            | 4    | RELAYREPLY   |
            +------+--------------+
```

                 Values for the 'op' field

RELAYFORWARD and RELAYREPLY messages share only the 'op' field in
common with other DHCP and BOOTP messages. The remainder of the
message consists of a series of fixed-length fields followed by two
variable-length fields: the relay segment, and the encapsulated
message.

```
+-----+-----+-----+-----+
|  op  |  ep  |   padlen  |
+-----+-----+-----+-----+
|   rslen   |   caplen  |
+-----+-----+-----+-----+
|         aiaddr        |
+-----+-----+-----+-----+
.                       .
.      relay segment    .
.                       .
+-----------------------+
.                       .
.  encapsulated message .
.                       .
+-----------------------+
```

3.1. The fixed-length header

The fixed-length header of the relay message contains a series of
fields that perform two purposes: to provide enough information that
the DHCP server can reconstruct the original packet sent by the DHCP
client, and to establish the lengths of the two variable-length
segments.

To satisfy the first of these requirements, two fields in the fixed-
length header report the amount of padding stripped from the client
message, if any, and whether or not an end option was stripped from
the client message. Except for a relay message that immediately
encapsulates a message from a DHCP client, these fields are always
zero. Using these two fields, the DHCP server can reconstruct the
client packet exactly, and this allows the DHCP server to validate
any signature [RFC3118] that may be present.

The fixed-length header consists of five fields:

op The BOOTP 'op' field, which, for a relay message, MUST contain the
   RELAYFORWARD or RELAYREPLY code.

ep If an End option was present in the option buffer prior to
   encapsulation, this field is set to 1; otherwise, it is set to 0.
   This field is a single byte.

padlen The length of any padding that was removed from the option
   buffer prior to encapsulation: two bytes in network byte order.

   rslen The length of the relay segment: two byte in network byte
      order.

   caplen The length of the encapsulation segment: two byte in network
      byte order.

   aiaddr Relay agent IP address.

3.2. Relay Segment

   The relay segment contains any RAIO suboptions that the encapsulating
   agent (the relay agent or the DHCP server) wishes to send. End and
   Pad options MUST NOT appear in the relay segment.

3.3. Encapsulation Segment

   The encapsulation segment contains the entire DHCP message being
   encapsulated, with four exceptions:

   o The encapsulating agent MUST omit the IP and UDP headers, as well
      as any layer two header, from the encapsulated message.

   o The encapsulating agent MUST omit any options following the first
      End option in the option buffer. These options are assumed to be
      garbage, and are not covered by any signature [RFC3118].

   o The encapsulating MUST omit any Pad options present either at the
      end of the option buffer, or prior to the first End option, that
      are followed only by other Pad options or a single End option.
      The encapsulating agent MUST record number of Pad options that
      were omitted in the 'padlen' field of the message header.

   o The encapsulating agent MUST omit the End option, if present. The
      encapsulating agent MUST set the 'ep' field in the message header
      to 1 if an End option was present in the option buffer, and to
      zero if no End option was present.

   These exceptions apply only to the option buffer. The encapsulating
   agent MUST NOT modify the contents of the 'file' and 'sname' fields.
   The encapsulating agent MUST NOT count End or Pad options that appear
   in these fields.

4. DHCP Relay Agent Behavior

   DHCP Relay agents implementing this specification MUST have a
   configuration parameter controlling relay encapsulation. By default,
   relay encapsulation MUST be disabled.

Relay agents with encapsulation disabled MUST NOT encapsulate. Relay
agents with encapsulation disabled MUST NOT decapsulate.

In any case where a relay agent implementing this specification does
not encapsulate or decapsulate, it MUST behave exactly as a relay
agent that does not implement this specification at all.

DHCP relay agents that are configured with encapsulation enabled, but
which have no agent-specific options to send to the DHCP server, MUST
encapsulate. Relay agents that are configured with encapsulation
enabled MUST decapsulate.

Layer two relay agents MUST silently discard any messages that
contains an IPsec authentication header [RFC4302]. This is because
they cannot modify such messages, but also cannot detect that a
message from the DHCP server is in response such messages, since the
response message might not contain an IPsec authentication header.

If a relay message would exceed the MTU of the outgoing interface, it
MUST be discarded, and an error condition SHOULD be logged.

4.1. Packet processing

Relay agents implementing this specification may receive packets
directed toward DHCP servers with a source port of 67 (BOOTPS).
Therefore, the source port cannot be used to determine whether the
packet is traveling toward a DHCP server or toward a DHCP client.

In order to determine whether a message is traveling toward a DHCP
client or toward a DHCP server, the relay agent must check the 'op'
field of the DHCP message. If the 'op' field is set to BOOTREQUEST
or RELAYFORWARD, the message is traveling toward a DHCP server. If
the 'op' field is set to BOOTREPLY or RELAYREPLY, the message is
traveling toward a DHCP client. At the time of the writing of this
specification, no other value is meaningful in the 'op' field.

Relay agents implementing this specification MUST NOT encapsulate or
decapsulate messages with other values in the 'op' field. It is
assumed that if meanings are defined for additional values, the
document that specifies the meaning of those values will update this
document; in the absence of such an update, the behavior specified
here will remain in effect.

Relay agents implementing this specification MAY differentiate
between DHCP and BOOTP messages. Under normal circumstances, BOOTP
and DHCP messages are forwarded to the same server, which should be
able to successfully decapsulate both DHCP and BOOTP messages.
However, some relay agents may send BOOTP and DHCP packets to

different servers; this document should not be construed to require
that such a relay agent should encapsulate all messages regardless of
protocol.

4.1.1. Packets traveling toward DHCP servers

Any DHCP or BOOTP packet with an 'op' value of BOOTREQUEST or
RELAYFORWARD is traveling toward a DHCP server. When a DHCP relay
agent that is configured to encapsulate receives such a packet, the
relay agent MUST encapsulate that packet into a RELAYFORWARD message
and send it to the address or addresses with which it is configured
to forward messages intended for DHCP servers.

4.1.2. Packets traveling toward DHCP clients

Any DHCP or BOOTP packet with an 'op' value of BOOTREPLY or
RELAYREPLY is traveling toward a DHCP client. When a DHCP relay
agent that is configured to encapsulate recieves a RELAYREPLY message
that is traveling toward a DHCP or BOOTP client, the relay agent MUST
decapsulate that message and forward the decapsulated message toward
the client.

4.1.3. Anti-spoofing

Because this specification allows for chaining of relay agent-
supplied information, it is now possible for a relay agent outside of
the trusted portion of a network to communicate relay agent
information to the DHCP server without preventing the legitimate
relay from communicating return path information to the DHCP server,
as is the case with RFC3046.

In order to prevent this sort of spoofing, relay agents implementing
this specification MUST be configurable to discard all RELAYFORWARD
messages that they receive. Administrators relying on a trusted
network architecture to control the flow of information to the DHCP
server SHOULD configure relay agents on the edge of their networks to
discard RELAYFORWARD messages.

4.2. Constructing RELAYFORWARD messages

4.2.1. Initializing the fixed-length header

The relay agent constructs the RELAYFORWARD message by constructing
the fixed-length header as specified in the earlier section titled
'Encoding'. The relay agent MUST set the 'op' field to a value of
RELAYFORWARD.

If the relay agent is not a layer two relay agent

[I-D.ietf-dhc-l2ra], it MUST store one of its own IP addresses in the
'aiaddr' field. This address MUST be a valid IP address that is
reachable by the next hop relay(s) or DHCP server(s) to which the
relay agent is configured to forward.

DHCP servers normally use the relay agent IP address to determine on
what link the DHCP client's IP address should be allocated. In some
cases, the value stored in the 'aiaddr' field will not be a valid IP
address on the link on which the source message was received. In
this case, the relay agent MUST include a link selection suboption
[RFC3527] that identifies that link in the relay segment.

If the relay agent is a layer two relay agent, it MAY include a link
selection suboption in the relay segment.

If the message being encapsulated is a BOOTREQUEST, L2RAs MUST store
a value of zero in the 'aiaddr' field. Otherwise, the L2RA MUST copy
the value of the 'aiaddr' field in the RELAYFORWARD message being
encapsulated into the 'aiaddr' field of the RELAYFORWARD message that
it generates.

The 'rslen' field depends on the length of the relay segment. The
'caplen', 'padlen' and 'ep' values in the fixed header are
initialized differently depending on whether the message being
encapsulated is a BOOTREQUEST or a RELAYFORWARD message.

4.2.2. Initializing the relay segment

Following the fixed header, the relay agent MUST append any RAIO
suboptions it wishes to send to the DHCP server; this is the 'relay
segment'. It MUST store the length of the relay segment in the
'rslen' field of the fixed header.

The relay agent SHOULD include a Relay Agent ID suboption
[I-D.ietf-dhc-relay-id-suboption] in the relay segment to identify
itself to the DHCP server.

4.2.3. Fixed header settings for RELAYFORWARD messages

If the message being encapsulated is a RELAYFORWARD message, the
relay agent MUST initialize the 'caplen' field of the fixed header to
the length of the source message, excluding any layer 2, IP and UDP
headers. It MUST append the contents of the message, again excluding
any layer 2, IP or UDP headers, to the new message. It MUST
initialize the 'ep' and 'padlen' fields in the fixed header of the
new message to zero.

4.2.4. **Fixed header settings for BOOTREQUEST messages**

   If the message being encapsulated is a BOOTREQUEST message, the relay
   agent determines the length of the encapsulation segment by scanning
   forward across the option buffer of the source message, beginning
   with the first option in the option buffer, until an End option is
   reached, or the end of the buffer is reached. The difference between
   the offset of this location in the message and the offset of the
   first location following the UDP header of the message is the length
   of the message to be relayed.

   If an End option terminated the scan, the relay agent MUST set the
   value of the 'ep' field in the fixed header to one. Otherwise, the
   relay agent MUST set the value of the 'ep' field to zero.

   The relay agent MUST count all of the Pad options that follow the
   last option in the option buffer that is neither a Pad nor an End
   option. The relay agent MUST store this count in the 'padlen' field
   of the fixed header.

   The relay agent MUST store the difference between the value stored in
   'padlen' and the length of the message to be relayed in the 'caplen'
   field of the fixed header.

4.2.5. **Initializing the encapsulation segment**

   The relay agent MUST copy the portion of the message being
   encapsulated that immediately follows the UDP header into the
   RELAYFORWARD message being generated. The length of the data being
   copied is the value that was stored in 'caplen'.

4.3. **Decapsulating RELAYREPLY messages**

   To decapsulate a RELAYREPLY message, the relay agent creates a
   decapsulated message, processes any RAIO suboptions it recognizes in
   the relay segment, and forwards the decapsulated message to its
   destination.

4.3.1. **Processing relay agent suboptions**

   The suboptions parsed from the relay segment are processed by the
   relay agent as specified in the Relay Agent Information Option
   specification [RFC3046], as well as in any documents that define
   suboptions to the Relay Agent Information Option. A current list of
   DHCP Relay Agent suboptions and the documents that define them can be
   located in the IANA protocol registry for Bootp and DHCP parameters,
   in the section titled "DHCP Relay Agent Sub-Option Codes."

4.3.2. Constructing the decapsulated message

   To construct a decapsulated message, the relay agent copies the
   portion of the RELAYREPLY message following the relay segment, with a
   length specified in the 'caplen' field of the fixed-length header,
   into the new message.

4.4. Retransmitting modified messages

   If the relay agent did not modify the message either by encapsulating
   or decapsulating it, it retransmits the message according to existing
   practice.

   Otherwise, how the modified message is transmitted to its next
   destination depends on two factors. First, is the relay agent that
   modified the message a layer two [I-D.ietf-dhc-l2ra] relay agent or a
   layer three [RFC1542] relay agent? Second, is the modified message a
   BOOTREPLY message, a RELAYREPLY message, or some other message?

4.4.1. Layer two relay agents

   There are two special aspects to the handling of relayed packets in
   Layer Two Relay Agents (L2RAs). The first is the construction of the
   layer two, IP and UDP headers on the packet. The second is how the
   L2RA makes the forwarding decision.

4.4.1.1. Constructing the headers

   The L2RA constructs the headers on the relayed packet by copying and,
   as necessary, modifying the headers from the original packet.

   If there is a layer two header, the L2RA MUST copy this header in
   accordance with the needs of the particular layer two implementation
   it is using. If the header contains a packet length field, the L2RA
   MUST adjust the value in the packet length field. If the header
   contains a non-secure integrity check such as a CRC or checksum that
   covers the entire packet, the L2RA MUST recompute this value.

   L2RA encapsulation in cases where the layer two contains a secure
   integrity check must either construct a new integrity signature, or
   else remove the integrity signature. If neither of these is
   possible, the L2RA MUST silently discard the packet.

   The L2RA MUST copy the IP header without modification except length
   and checksum field which should be recomputed. If the IP header
   contains any sort of secure integrity check on the packet, the L2RA
   MUST silently discard the packet.

The L2RA MUST copy the UDP header and adjust the 'Length' field
[RFC0768]. If the contents of the 'Checksum' field are not zero, the
L2RA MUST compute a new checksum according to the algorithm specified
in User Datagram Protocol. [RFC0768]

4.4.1.2. Forwarding the modified packet

Ordinarily when a layer two device forwards a packet, it simply
copies that packet from the interface on which it was received and
transmits it, unchanged, on one or more interfaces other than that
interface. The mechanism used to choose which interfaces it forwards
the packet to is beyond the scope of this document.

Once a DHCP packet has been modified by the L2RA either as an
encapsulation or a decapsulation, the L2RA must forward it either
toward the DHCP server or the DHCP client. The implementation has
two options: transmit the packet as it would transmit any other
packet, or use its configuration and/or information in the relay
header to direct the packet out a particular interface.

The details of ordinary packet forwarding in devices that implement
L2RA is beyond the scope of this document.

When processing a RELAYREPLY message, the L2RA MAY use information in
the relay segment of the RELAYREPLY to determine on which network
interface the RELAYREPLY should be forwarded.

When processing any other message, the L2RA MAY use configuration
information to direct the packet out a specific port or ports that
have been marked as reaching DHCP servers. The L2RA MUST NOT forward
any packet on the interface on which it was received, even if that
interface is so marked.

4.4.2. Layer three relay agents

4.4.2.1. Transmitting a decapsulated RELAYREPLY message

When the decapsulated message is itself a RELAYREPLY message, the
relay agent MUST forward the decapsulated message to the IP address
specified in the 'aiaddr' field of the fixed-length header.

If the relay segment of the packet that was decapsulated contains a
Link Layer Address suboption, the relay agent MUST transmit the
packet to that link layer address without attempting to use Address
Resolution Protocol (ARP) to translate the address contained in
'aiaddr' to a layer two address.

4.4.2.2. Transmitting a decapsulated BOOTREPLY message

When transmitting a decapsulated BOOTREPLY message, the relay agent transmits the message as specified in Bootstrap Protocol, Section 4 [RFC0951].

4.4.2.3. Transmitting other messages

When transmitting RELAYFORWARD and BOOTREQUEST messages, the relay agent simply sends the message to the IP address or addresses configured as DHCP servers for that relay agent.

5. DHCP Server Behavior

A DHCP server which receives a RELAYREPLY message MUST silently discard that message.

5.1. Receiving RELAYFORWARD messages

DHCP servers that implement this specification MUST decapsulate RELAYFORWARD messages.

5.1.1. Decapsulation

By design, this specification supports multiple layers of encapsulation. The DHCP server implementation therefore MUST decapsulate each layer and retain the information in that layer, organized so that the ordering of the encapsulation is available both during packet processing, and when constructing the reply.

Aside from the necessity of handling an RAIO differently than an encapsulation when constructing a RELAYREPLY message, DHCP servers MUST NOT, by default, treat relay suboptions received in an RAIO any differently than relay suboptions received in an encapsulation.

It is not the goal of this specification to define a particular implementation strategy or data structure for representing the encapsulation, so long as the ability to process the options and suboptions as required below is preserved. Implementations MAY provide means for addressing the contents of relay segments and of the inner encapsulation segment in any convenient form, as long as it conforms generally to the requirements of this document.

5.1.2. Processing of decapsulated suboptions

This section specifies requirements for the processing of decapsulated relay suboptions in the default case, where no custom

processing has been specified. This should not be construed to
forbid implementations for providing mechanisms for customizing the
processing of these suboptions.

This document does not specify special handling for DHCP options.
Only the inner encapsulation--the encapsulation of the original
message sent from the client, which is done by the first
encapsulating relay--can ever contain DHCP Options; hence, whatever
normal mechanisms a DHCP server provides for processing these options
should suffice.

Some relay agent suboptions, such as the Relay Agent Subnet Selection
suboption [RFC3527], are intended to be processed by DHCP servers.
Others, like the Circuit ID and Remote ID [RFC3046] suboptions, were
not intended to be processed by the DHCP server, but are often used
by the DHCP server for identification purposes.

In cases where more than one relay agent sends the same suboption,
the DHCP server must either factor all such suboptions into its
processing, or choose one such suboption and use that exclusively for
processing.

By default, DHCP servers MUST use the outermost suboption (the one
added by the relay agent closest to the DHCP server) for every
suboption that was sent by more than one relay agent.

5.1.3. Address allocation

During normal processing, DHCP servers use a variety of data to
determine where the DHCP client is in the network topology. These
data are provided by relay agents. In the absence of any relay-
agent-provided topology data, the DHCP server assumes that the client
is connected to the link on which the message was received.

One datum used by DHCP servers in locating the client is the value of
the 'giaddr' field of the BOOTP header. This specification
eliminates the use of giaddr; hence, it cannot be used in the address
allocation decision.

The functionality provided by giaddr is replaced in this
specification by the 'aiaddr' field in the fixed-length relay header.

5.1.3.1. Default link selection algorithm

DHCP servers MUST implement a default algorithm for determining the
link to which the client is attached. This algorithm is to first
search the client message for a subnet selection option [RFC3011].

The server next searches for link-identifying data in each
RELAYFORWARD encapsulation, starting from the inner most and ending
at the outermost, until a RELAYFORWARD is found that identifies the
client's location.

A RELAYFORWARD encapsulation contains link-identifying data if the
value of the 'aiaddr' field of the fixed-length header is not zero,
or if the relay segment contains a Link Selection suboption
[RFC3527].

If a Link Selection suboption is present in the innermost
RELAYFORWARD message containing link-identifying data, the DHCP
server MUST use the contents of that option to identify the link to
which the client is connected.

Otherwise, if a Subnet Selection option was found in the client
message, the DHCP server MUST use the contents of that option to
identify the link to which the client is connected.

Otherwise, the DHCP server MUST use the contents of the 'aiaddr'
field in the RELAYFORWARD encapsulation that was identified as being
the innermost RELAYFORWARD containing link-identifying data.

5.1.3.2. Other link selection algorithms

DHCP servers implementing this specification MAY implement link
selection algorithms other than the one described in the preceding
section. DHCP servers MUST NOT use any link selection algorithm
other than the one described in the preceding section unless
specially configured to do so.

5.2. Responding to RELAYFORWARD messages

Once the DHCP server has processed the encapsulated message from the
DHCP client and constructed a response to the DHCP client, it
constructs a RELAYREPLY message and sends it toward the client.

5.2.1. Constructing a RELAYREPLY encapsulation

The server MUST encapsulate any response to a client message
contained in one or more RELAYFORWARD encapsulations in a set of
corresponding RELAYREPLY encapsulations. Each RELAYREPLY is nested
in the same way that the corresponding RELAYFORWARD was nested, so
that the innermost RELAYREPLY corresponds to the innermost
RELAYFORWARD, and the outermost RELAYREPLY corresponds to the
outermost RELAYFORWARD.

5.2.1.1. Constructing the relay segments

   The server MUST copy every suboption that appeared in the relay
   segment of the RELAYFORWARD encapsulation into corresponding outgoing
   RELAYREPLY encapsulation's relay segment. The server SHOULD NOT
   preserve the order of options from the incoming relay segment to the
   outgoing relay segment.

   If the message encapsulated by a particular RELAYREPLY encapsulation
   is not a RELAYREPLY, or if the message encapsulated by that
   RELAYREPLY is a RELAYREPLY, but the 'aiaddr' field of the fixed-
   length header of the inner RELAYREPLY has a value of zero, the DHCP
   server MUST include a Layer Two Address suboption in the relay
   segment. The DHCP server MUST set the 'htype' field of the Layer Two
   Address suboption to the value of 'htype' in the client message. The
   DHCP server MUST set the 'chaddr' field in the Layer Two Address
   suboption to the value of the 'chaddr' field in the client message.

5.2.1.2. Constructing the fixed-length header

   The server MUST set the values of 'ep' and 'padlen' in the fixed-
   length header to zero. The server MUST set the value of 'caplen' to
   the length of the message encapsulated in the RELAYREPLY
   encapsulation being constructed. The server MUST set the value of
   'rslen' to the length of the relay segment of the RELAYREPLY
   encapsulation being constructed.

   If the message encapsulated in the RELAYREPLY being constructed is a
   RELAYREPLY, and the 'aiaddr' field of the RELAYFORWARD encapsulation
   corresponding to the inner RELAYREPLY is not zero, the DHCP server
   MUST copy the value from that 'aiaddr' field to the 'aiaddr' field of
   the RELAYREPLY being constructed.

   Otherwise, if the BROADCAST bit in the 'flags' field of the inner
   message is set to 1, or 'ciaddr' is zero and 'yiaddr' is also zero,
   the DHCP server MUST set the value of 'aiaddr' to 255.255.255.255.
   If 'ciaddr' is not zero, the DHCP server must copy that value to
   'aiaddr'. If 'ciaddr' is zero and 'yiaddr' is not, the DHCP server
   MUST copy the value of 'yiaddr' to 'aiaddr'.

5.2.2. Transmission of RELAYREPLY messages

   If the value of 'aiaddr' in the outermost RELAYFORWARD encapsulation
   to which the RELAYREPLY is a response is nonzero, the DHCP server
   MUST transmit the RELAYREPLY to that IP address.

   Otherwise, if 'ciaddr' and 'yiaddr' are both zero, or the BROADCAST
   bit in the 'flags' field is set to 1, or the DHCP server

implementation is not able to perform the ARP-cache preloading trick
described in Bootstrap Protocol, Section 4 [RFC0951], the DHCP server
MUST broadcast the RELAYREPLY message to the 255.255.255.255
broadcast address.

Otherwise, the DHCP server MUST use the value of 'ciaddr', if
nonzero, or 'yiaddr', otherwise, as the destination address for the
message, and MUST preload its ARP cache (or otherwise arrange to
transmit the message without using ARP) to the layer two address
provided by the client in 'htype' and 'chaddr' and 'hlen'.

5.3. Responding to messages other than RELAYFORWARD

When a DHCP server constructs a response to a DHCP client message
that did not arrive encapsulated in a RELAYFORWARD message, the DHCP
server MUST NOT encapsulate the response in a RELAYREPLY message.
DHCP server implementors should be careful that their servers do not
respond to an incoming packet with RAIO from a non-conforming relay
agent with a RELAYREPLY message.

6. DHCP Client Behavior

A DHCP client that receives either a RELAYFORWARD message or a
RELAYREPLY message MUST silently discard that message.

7. Security Considerations

DHCP Relay Information Option [RFC3046] limits relay agent
information to a single relay agent, and provides some minimal anti-
spoofing. By supporting relay agent chaining, it is now possible for
a relay agent downstream of the trusted portion of a provider network
to communicate relay agent information options to a DHCP server.

This specification defines a much more robust spoofing rejection
mechanism, by allowing the administrator to configure the relay to
discard RELAYFORWARD messages originating from outside of the trusted
portion of the network. Administrators of networks that rely on this
trusted/untrusted configuration are encouraged to configure edge
relays to discard RELAYFORWARD messages.

In networks where trusted relay agents may be separated from the DHCP
server by transit networks that are not trusted, it is possible to
modify the message in transit. This possibility exists with normal
DHCP relays as well. Administrators of such networks should consider
using relay agent authentication [RFC4030] to prevent modification of
relay agent communications in transit.

8. IANA Considerations

   We request that IANA assign one new suboption code from the registry
   of DHCP Agent Sub-Option Codes maintained in
   http://www.iana.org/assignments/bootp-dhcp-parameters. This
   suboption code will be assigned to the Layer Two Address Suboption.

9. References

9.1. Normative References

   [I-D.ietf-dhc-relay-id-suboption]
              Stapp, M., "The DHCPv4 Relay Agent Identifier Suboption",
              draft-ietf-dhc-relay-id-suboption-07 (work in progress),
              July 2009.

   [RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768,
              August 1980.

   [RFC0951] Croft, B. and J. Gilmore, "Bootstrap Protocol", RFC 951,
              September 1985.

   [RFC1542] Wimer, W., "Clarifications and Extensions for the
              Bootstrap Protocol", RFC 1542, October 1993.

   [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.

   [RFC3011] Waters, G., "The IPv4 Subnet Selection Option for DHCP",
              RFC 3011, November 2000.

   [RFC3046] Patrick, M., "DHCP Relay Agent Information Option",
              RFC 3046, January 2001.

   [RFC3118] Droms, R. and W. Arbaugh, "Authentication for DHCP
              Messages", RFC 3118, June 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.

   [RFC4030] Stapp, M. and T. Lemon, "The Authentication Suboption for
              the Dynamic Host Configuration Protocol (DHCP) Relay Agent
              Option", RFC 4030, March 2005.

   [RFC4302]   Kent, S., "IP Authentication Header", RFC 4302,
               December 2005.

9.2. Informative References

   [I-D.ietf-dhc-l2ra]
               Joshi, B. and P. Kurapati, "Layer 2 Relay Agent
               Information", draft-ietf-dhc-l2ra-04 (work in progress),
               April 2009.

   [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.

Authors' Addresses

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

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

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

   Email: denghui@chinamobile.com

   Lu Huang
   China Mobile
   53A, Xibianmennei Ave.
   Xunwu District, Beijing 100053
   China

   Email: huanglu@chinamobile.com