Internet Engineering Task Force
INTERNET DRAFT
DHC Working Group
Obsoletes: draft-ietf-dhc-dhcpv6-11.txt

J. Bound
Digital Equipment Corp.
C. Perkins
Sun Microsystems

21 November 1997

Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
draft-ietf-dhc-dhcpv6-11.txt

Status of This Memo

Abstract

The Dynamic Host Configuration Protocol (DHCPv6) provides a framework for passing configuration information, via extensions, to IPv6 nodes. It offers the capability of automatic allocation of reusable network addresses and additional configuration flexibility. This protocol should be considered a stateful counterpart to the IPv6 Stateless Address Autoconfiguration protocol specification, and can be used separately or together with the latter to obtain configuration information.

Contents

1. Introduction

The Dynamic Host Configuration Protocol (DHCPv6, or in this document usually DHCP) provides configuration parameters to Internet nodes. DHCP consists of a protocol for delivering node-specific configuration parameters from a DHCP server to a client, and a mechanism for allocation of network addresses and other related parameters to IPv6 [6] nodes.

DHCP is built on a client-server model, where designated DHCP servers allocate network addresses and automatically deliver configuration parameters to dynamically configurable clients. Throughout the remainder of this document, the term "server" refers to a node providing initialization parameters by way of the DHCP protocol, and the term "client" refers to a node requesting initialization parameters from a DHCP server.

DHCPv6 uses Request and Reply messages to support a client/server processing model whereby both client and server are assured that requested configuration parameters have been received and accepted by the client. DHCP supports optional configuration parameters and processing for nodes through extensions described in its companion document ``Extensions for the Dynamic Host Configuration Protocol for IPv6'' [12].

The IPv6 Addressing Architecture [8] and IPv6 Stateless Address Autoconfiguration [16] specifications provide new features not available in IP version 4 (IPv4) [15], which are used to simplify and generalize the operation of DHCP clients. This document is intended to complement those specifications for clients attached to the kinds of Internet media for which those specifications apply. In particular, the specification in this document does not necessarily apply to nodes which do not enjoy a full duplex link to the Internet.

Section 2 provides definitions for terminology used throughout this document. Section 3 provides an overview of the protocol design model that guided the design choices in the specification; section 3.2 briefly describes the protocol messages and their semantics. Section 4 provides the message formats and field definitions used for each message. Sections 5, 6, and 7 specify how clients, servers, and relays interact. The timeout and retransmission guidelines and configuration variables are discussed in Section 8. Appendix A summarizes related work in IPv6 that will provide helpful context; it is not part of this specification, but included for informational purposes. Appendix B discusses the differences between DHCPv4 and DHCPv6.

2. Terminology and Definitions

Relevant terminology from the IPv6 Protocol [6], IPv6 Addressing Architecture [8], and IPv6 Stateless Address Autoconfiguration [16] will be provided, and then the DHCPv6 terminology.

2.1. IPv6 Terminology

address — An IP layer identifier for an interface or a set of interfaces.

unicast address
An identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address.

multicast address
An identifier for a set of interfaces (typically belonging to different nodes). A packet sent to a multicast address is delivered to all interfaces identified by that address.

host — Any node that is not a router.

IP — Internet Protocol Version 6 (IPv6). The terms IPv4 and IPv6 are used only in contexts where it is necessary to avoid ambiguity.

interface
A node's attachment to a link.

link — A communication facility or medium over which nodes can communicate at the link layer, i.e., the layer immediately below IP. Examples are Ethernet (simple or bridged); Token Ring; PPP links, X.25, Frame Relay, or ATM networks; and internet (or higher) layer "tunnels", such as tunnels over IPv4 or IPv6 itself.

link-layer identifier
a link-layer identifier for an interface. Examples include IEEE 802 addresses for Ethernet or Token Ring network interfaces, and E.164 addresses for ISDN links.

link-local address
An IP address having link-only scope, indicated by having the routing prefix FE80::0000/64), that can be used to reach neighboring nodes attached to the same link. Every interface has a link-local address.

message   A unit of data carried in a packet, exchanged between DHCP agents and clients.

neighbor   A node attached to the same link.

node   A device that implements IP.

packet   An IP header plus payload.

prefix   A bit string that consists of some number of initial bits of an address.

router   A node that forwards IP packets not explicitly addressed to itself.

2.2. DHCPv6 Terminology

Agent Address

The address of a neighboring DHCP Agent on the same link as the DHCP client.

binding   A binding (or, client binding) in DHCP contains the data which a DHCP server maintains for each of its clients (see Section 6).

resource-server association

An association between a resource and a DHCP server maintained by the client which received that resource from that DHCP server.

configuration parameter

Any parameter that can be used by a node to configure its network subsystem and enable communication on a link or internetwork.

DHCP agent (or agent)

Either a DHCP server or a DHCP relay.

DHCP client (or client)

A node that initiates requests on a link to obtain configuration parameters.

DHCP relay (or relay)

A node that acts as an intermediary to deliver DHCP messages between clients and servers.

DHCP server (or server)

A server is a node that responds to requests from

clients to provide: addresses, prefix lengths, or other configuration parameters.

transaction-ID
The transaction-ID is a monotonically increasing unsigned integer identifier specified by the client or server, and used to match a response to a pending message.

2.3. Specification Language

In this document, several words are used to signify the requirements of the specification, in accordance with RFC 2119 [2]. These words are often capitalized.

MUST
This word, or the adjective "required", means that the definition is an absolute requirement of the specification.

MUST NOT
This phrase means that the definition is an absolute prohibition of the specification.

SHOULD
This word, or the adjective "recommended", means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications must be understood and carefully weighed before choosing a different course. Unexpected results may result otherwise.

MAY
This word, or the adjective "optional", means that this item is one of an allowed set of alternatives. An implementation which does not include this option MUST be prepared to interoperate with another implementation which does include the option.

silently discard
The implementation discards the packet without further processing, and without indicating an error to the sender. The implementation SHOULD provide the capability of logging the error, including the contents of the discarded packet, and SHOULD record the event in a statistics counter.

3. Protocol Design Model

This section is provided for implementors to understand the DHCPv6 protocol design model from an architectural perspective. Goals and conceptual models are presented in this section.

3.1. Design Goals

The following list gives general design goals for this DHCP specification.

- DHCP should be a mechanism rather than a policy. DHCP MUST allow local system administrators control over configuration parameters where desired; e.g., local system administrators should be able to enforce local policies concerning allocation and access to local resources where desired.

- DHCP MUST NOT introduce any requirement for manual configuration of DHCP clients, except when security requirements need authentication or encryption keys. Each node should be able to obtain appropriate local configuration parameters without user intervention, and incorporate those parameters into its own configuration.

- DHCP MUST NOT require a server on each link. To allow for scale and economy, DHCP MUST work across DHCP relays.

- A DHCP client MUST be prepared to receive multiple (possibly different) responses to solicitations for DHCP servers. Some installations may include multiple, overlapping DHCP servers to enhance reliability and/or to increase performance.

- DHCP MUST coexist with statically configured, non-participating nodes and with existing network protocol implementations.

- DHCPv6 MUST be compatible with IPv6 Stateless Address Autoconfiguration [16].

- A DHCPv6 Client implementation MAY be started in the absence of any IPv6 routers on the client's link.

- DHCP architecture MUST support the requirements of automated renumbering of IP addresses [3].

- DHCP servers SHOULD be able to support Dynamic Updates to DNS [18].

- DHCP servers MUST be able to support multiple IPv6 addresses for each client.

- DHCP MUST work on network links that do not have routers, as long as a DHCP server is present on the link.

- A DHCP server to server protocol is NOT part of this specification.

- It is NOT a design goal of DHCP to specify how a server configuration parameter database is maintained or determined. Methods for configuring DHCP servers are outside the scope of this document.

3.2. DHCP Messages

Each DHCP message contains a type, which defines its function within the protocol. Processing details for these DHCP messages are specified in Sections 5, 6, and 7. The specified message types are as follows:

01 DHCP Solicit

The DHCP Solicit message is an IP multicast message sent by a DHCP client to one or more DHCP agents.

02 DHCP Advertise

The DHCP Advertise is an IP unicast message sent by a DHCP Agent in response to a DHCP client's DHCP Solicit message.

03 DHCP Request

The DHCP Request is an IP unicast message sent by a DHCP client to a DHCP server to request configuration parameters on a network.

04 DHCP Reply

The DHCP Reply is an IP unicast message sent by a DHCP server in response to a client's DHCP Request, or by the DHCP relay that relayed that client's DHCP Request. Extensions [12] to the DHCP Reply describe the resources that the DHCP server has committed and allocated to this client, and may contain other information for use by this client.

05 DHCP Release

The DHCP Release is an IP unicast message sent by a DHCP client to inform the DHCP server that the client is releasing resources.

06 DHCP Reconfigure

The DHCP Reconfigure is an IP unicast or multicast message sent by a DHCP server to inform one or more clients that the server has new configuration information of importance. Each client is expected to initiate a new Request/Reply transaction.

DHCP message types not defined here (msg-types 0 and 7-255) are reserved and SHOULD be silently ignored.

3.3. Request/Response Processing Model

The request/response processing for DHCPv6 is transaction based and uses a set of best-effort messages to complete the transaction.

To find a server, a client sends a DHCP Solicit message from the interface which it wishes to configure. The client then awaits a DHCP Advertise message, which will provide an IP address of a DHCP server. Transactions are started by a client with a DHCP Request, which may be issued after the client knows the server's address. The response (DHCP Reply) is sent from the server (possibly via a DHCP Relay). At this point in the flow all data has been transmitted and is presumed to have been received. To provide a method of recovery if either the client or server do not receive the messages to complete the transaction, the client retransmits each DHCP Request message until it elicits the corresponding DHCP Reply, or until it can be reasonably certain that the desired DHCP server is unavailable, or it determines that it does not want a response (i.e., it MAY abort the transaction). The timeout and retransmission guidelines and configuration variables are discussed in Section 8.

DHCP uses the UDP [14] protocol to communicate between clients and servers. UDP is not reliable, but the DHCP retransmission scheme in the referenced section provides reliability between clients and servers. The following well-known multicast addresses are used by DHCP agents and clients:

FF02:0:0:0:0:0:1:2
All DHCP Agents (Servers and Relays) MUST join the link-local All-DHCP-Agents multicast group at the address FF02:0:0:0:0:0:1:2.

FF05:0:0:0:0:0:1:3

All DHCP servers MUST join the site-local All-DHCP-Servers multicast group at the address FF05:0:0:0:0:0:1:3.

FF05:0:0:0:0:0:1:4

All DHCP Relays MUST join the site-local All-DHCP-Relays multicast group at the address FF05:0:0:0:0:0:1:4.

A DHCP server or agent MUST transmit all messages to DHCP clients on UDP port 546. A DHCP client MUST transmit all messages to a DHCP agent over UDP using port 547. A DHCP server MUST transmit all messages to DHCP Relays over UDP on port 546. The source port for DHCP messages is arbitrary.

For the proper operation of the DHCP protocol to operate within a network where one or more firewalls [4] are used, DHCP transactions using UDP destination ports 546 and 547 will need to be permitted.

4. DHCP Message Formats and Field Definitions

All fields in DHCP messages MUST be initialized to binary zeroes by both the client and server unless otherwise noted. All reserved fields in a message MUST be ignored by the receiver of the message.

4.1. DHCP Solicit Message Format

A DHCP client transmits a DHCP Solicit message over the interface it is trying to configure, to obtain one or more DHCP server addresses. In the event that there is no DHCP server on this link, such a request MAY be forwarded by a DHCP relay attached to this link (if such a relay exists) on behalf of a client to a DHCP server.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    msg-type   |C|                   reserved                  |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                   client's link-local address                 |
   |                          (16 octets)                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         relay address                         |
   |                          (16 octets)                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

msg-type   1

C — If set, the client requests that all servers receiving the message deallocate the resources associated with the client.

reserved — 0

client's link-local address — The IP link-local address of the client interface from which the client issued the DHCP Request message

relay address — If nonzero, the IP address of the interface on which the relay received the client's DHCP Solicit message

To obtain a DHCP Agent address a DHCP client SHOULD send a DHCP Solicit message to the All-DHCP-Agents multicast address (see section 3.3). Any DHCP Relay receiving the solicitation MUST forward it to the All-DHCP-Servers multicast address, to instruct DHCP servers to send their advertisements to the prospective client. In that case, the relay MUST copy a non-link-local address of its interface from which the client's solicitation was received into the relay address field.

4.2. DHCP Advertise Message Format

A DHCP agent sends a DHCP Advertise message to inform a prospective client about the IP address of a DHCP Agent to which a DHCP Request message may be sent. When the client and server are on different links, the server sends the advertisement back through the DHCP Relay whence the solicitation came.

```
      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
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |    msg-type   |S|P|                 reserved                  |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                 server preference (if present)                |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                   client's link-local address                 |
     |                          (16 octets)                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                         agent address                         |
     |                          (16 octets)                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                         server address                        |
     |                    (16 octets, if present)                    |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |              extensions (variable number and length) ...
```

```
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

msg-type   2

S   If set, the server address is present.

P   If set, the server preference is present.

reserved   0

server preference
A 32-bit unsigned integer indicating a server's willingness to provide service to the client.

client's link-local address
The IP link-local address of the client interface from which the client issued the DHCP Request message

agent address
The IP address of a DHCP Agent interface on the same link as the client.

server address
If present, the IP address of the DHCP server

extensions   See [12].

Suppose that a DHCP server on the same link as a client issues the DHCP Advertise in response to a DHCP Solicit message sent to the All-DHCP-Agents multicast address. Then the agent address will be an IP address of one of the server's interfaces on the same link as the client, and the 'S' bit will be set to zero. No server address will be present in the DHCP Advertise message.

If the 'P' bit is set, the server preference field is present. If the 'P' bit is not set, the server preference field is not present, but implicitly has the value of 0xffffffff (in other words, the highest possible value).

The DHCP server MUST copy the client's link-local address into the advertisement which is sent in response to a DHCP Solicit. Both agent address and server address (if present) of the DHCP Advertise message MUST have sufficient scope to be reachable by the DHCP client. Moreover, the Agent address of any DHCP Advertise message sent by a DHCP relay MUST NOT be a link-local address. In situations where there are no routers sending Router Advertisements, then a DHCP server MUST be configured on the same link as prospective clients. The DHCPv6 protocol design does not apply to situations where the client has no way to route messages to a server not on the same link.

See section 5.3 for information about how clients handle the server preference field.

4.3. DHCP Request Message Format

In order to request configuration parameters from a DHCP server, a client sends a DHCP Request message, and MAY append extensions [12]. If the client does not know any DHCP server address, it MUST first obtain a server address by multicasting a DHCP Solicit message (see Section 4.1). If the client does not have a valid IP address of sufficient scope for the DHCP server to communicate with the client, the client MUST send the message to the local DHCP relay and insert the DHCP relay address as the agent address in the message header. In this case, the client cannot send the message directly to the DHCP server because the server could not return any response to the client. Otherwise, the client MAY omit the server address in the DHCP Request message; in this case, the client MUST clear the S-bit in the DHCP Request message and send it directly to to the server, using the server address as the IP destination address in the IP header.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    msg-type   |C|S|R|N|  rsvd |        transaction-ID         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   client's link-local address                 |
|                          (16 octets)                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         agent address                         |
|                          (16 octets)                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         server address                        |
|                    (16 octets, if present)                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  extensions (variable number and length)   ....
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

msg-type   3

C   If set, the client requests the server to remove all resources associated with the client binding, except those resources provided as extensions.

S   If set, the server address is present

R If set, the client has rebooted and requests that all of its previous transaction IDs be expunged and made available for re-use.

N If set, the client has begun operating in stateless address autoconfiguration.

rsvd 0

transaction-ID
A monotonically increasing unsigned integer used to identify this Request, and copied into the Reply.

client's link-local address
The IP link-local address of the client interface from which the client issued the DHCP Request message

agent address
The IP address of an agent's interface, copied from a DHCP Advertisement message.

server address
If present, the IP address of the DHCP server which should receive the client's DHCP Request message.

extensions See [12].

4.4. DHCP Reply Message Format

The server sends one DHCP Reply message in response to every DHCP Request or DHCP Release received. If the request comes with the 'S' bit set, the client could not directly send the Request to the server and had to use a neighboring relay agent. In that case, the server sends back the DHCP Reply with the 'L' bit set, and the DHCP Reply is addressed to the agent address found in the DHCP Request message. If the 'L' bit is set, then the client's link-local address will also be present.

```
   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
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |    msg-type   |L|   status    |        transaction-ID         |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |                   client's link-local address                 |
  |                     (16 octets, if present)                   |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  extensions (variable number and length)   ....
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

msg-type 4

L If set, the client's link-local address is present

status One of the following decimal values:

0	Success
16	Failure, reason unspecified
17	Authentication failed or nonexistent
18	Poorly formed Request or Release
19	Resources unavailable
20	Client record unavailable
21	Invalid client IP address in Release
23	Relay cannot find Server Address
24	Cannot understand selected Character Set
64	Server unreachable (ICMP error)

transaction-ID
A monotonically increasing unsigned integer used to identify this Reply, and copied from the client's Request.

client's link-local address
If present, the IP address of the client interface which issued the corresponding DHCP Request message.

extensions
See [12].

If the 'L' bit is set, and thus the link-local address is present in the Reply message, the Reply is sent by the server to the relay's address which was specified as the agent address in the DHCP Request message, and the relay uses the link-local address to deliver the Reply message to the client.

4.5. DHCP Release Message Format

The DHCP Release message is sent without the assistance of any DHCP relay. When a client sends a Release message, it is assumed to have a valid IP address with sufficient scope to allow access to the target server. If parameters are specified in the extensions, only those parameters are released. The DHCP server acknowledges the Release message by sending a DHCP Reply (Sections 4.4, 6.3).

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    msg-type   |D|  reserved   |        transaction-ID         |
```

```
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                   client's link-local address                    |
 |                          (16 octets)                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                         agent address                            |
 |                          (16 octets)                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                        client address                            |
 |                    (16 octets, if present)                       |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  extensions (variable number and length)   ....
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

msg-type 5

D If the 'D' ("Direct") flag is set, the client instructs the server to send the DHCP Reply directly back to the client, instead of using the given agent address and link-local address to relay the Reply message.

reserved 0

transaction-ID
A monotonically increasing unsigned integer used to identify this Release, and copied into the Reply.

client's link-local address
The IP link-local address of the client interface from which the the client issued the DHCP Release message

agent address
The IP address of the agent interface to which the client issued a previous DHCP Request message

client address
The IP address of the client interface from which the the client issued the DHCP Release message. The client address field is present whenever the 'D' bit is set, even if it is equal to the link-local address.

extensions See [12]

Suppose that the client has an IP address that will still be valid after the server performs the operations requested in the extensions to the DHCP Release message, and which has sufficient scope to be reachable from the server. In that case, and only then, the client SHOULD set the 'D' flag. When the 'D' flag is set, the server MUST send the DHCP Reply back to the client using the client address field of the Release message. Otherwise, when the 'D' bit is not set, the

server MUST send its DHCP Reply message to the agent address in the Release message, so that the relay agent can subsequently forward the Reply back to the releasing client at the client's link-local address indicated in the Reply message. Note that it is an error (status code 21) to include within the DHCP Release message both the 'D' bit and an IP address extension which has the IP address used as the client IP address field of the DHCP Release message header. If the clients link-local address and agent address do not match a client binding (see section 6) an error (status code 20) will be returned to the client.

4.6. DHCP Reconfigure Message Format

Reconfigure messages can only be sent to clients which have established an IP address which routes to the link at which they are reachable, hence the DHCP Reconfigure message is sent without the assistance of any DHCP relay. When a server sends a Reconfigure message, the client to which it is sent is assumed to have a valid IP address with sufficient scope to be accessible by the server. Reconfigure messages can only be sent to clients which are reachable by the DHCP server. Hence, the DHCP Reconfigure message is sent without the assistance of any DHCP relay. Only the parameters which are specified in the extensions to the Reconfigure message need be requested again by the client. A Reconfigure message can either be unicast or multicast by the server.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    msg-type   |N|  reserved   |         transaction-ID        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        server address                         |
|                          (16 octets)                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  extensions (variable number and length)   ....
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

msg-type 6

N The 'N' flag indicates that the client should not expect a DHCP Reply in response to the DHCP Request it sends as a result of the DHCP Reconfigure message.

reserved 0

transaction-ID
A monotonically increasing unsigned integer used to

identify this Reconfigure message, and copied into the client's Request.

server address
The IP address of the DHCP server issuing the DHCP Reconfigure message.

extensions See [12]

5. DHCP Client Considerations

A node which is not a DHCP agent MUST silently discard any DHCP Solicit, DHCP Request, or DHCP Release message it receives.

5.1. Verifying Resource Allocations After Restarts

A DHCP client MAY retain its configured parameters and resources across client system reboots and DHCP client program restarts. However, in these circumstances a DHCP client MUST also formulate a DHCP Request message to verify that its configured parameters and resources are still valid. This Request message MUST have the 'C' bit set, to clean up stale client binding information at the server which may no longer be in use by the client; stale information is that which the client does not include in extensions to such request messages.

If the server does not respond to the DHCP Request message after REPLY_MSG_MIN_RETRANS (see section 8), the client may still use any resources whose lifetimes have not yet expired. In such cases, however, the client MUST begin to search for another server by multicasting a new DHCP Solicit message, again with the 'C' bit set.

This also handles the case wherein a client restarts on a new network, where its IP address is no longer valid. In this situation, when the client receives a new IP address and the old IP address is no longer needed, the client MUST release its old IP address by issuing a DHCP Release message with the appropriate extension if it can communicate with its previous server.

5.2. Sending DHCP Solicit Messages

A DHCP client MUST have the address of a DHCP server to send a Request message. The client SHOULD locate a DHCP server by multicasting a DHCP Solicit message to the All-DHCP-Agents link-local multicast address, setting the Hop Limit == 1 (see Section 3.3). If there are no DHCP servers on the same link as the node, then a

DHCP relay MUST be present if solicitations sent from a client's link-local address are to be handled. The prospective client SHOULD wait for ADV_CLIENT_WAIT to get all the DHCP Advertisement messages which may be sent in response to the solicitation.

When sending a DHCP Solicit message, a client MUST set the Relay Address field to 16 octets of zeros.

If a DHCP client reboots and does not have a valid IP address, it MUST set the 'C' bit in the DHCP Solicit message it sends when restarting. By setting the 'C' bit in the solicitation, a DHCP client requests that all the DHCP servers that receive the solicitation should clean up their client records that match its link-local address. The server MUST take the Relay Address Field and use it as the agent address prefix to locate the client binding (see section 6.0 for server bindings).

If a client sends a DHCP Solicit message after it reboots, the solicitation SHOULD be delayed after reception of the first Router Advertisement [11] message, by at least some random amount of time between MIN_SOLICIT_DELAY and MAX_SOLICIT_DELAY (see section 8 seconds. This delay is intended to help stagger requests to DHCP servers (and avoid link-layer collisions) after a power outage causes many nodes to reboot all at once. Each subsequent DHCP Solicit message that is issued before receiving an advertisement MUST be delayed by twice the amount by which the previous DHCP Solicit message was delayed, plus a small random delay between MIN_SOLICIT_DELAY and MAX_SOLICIT_DELAY seconds.

5.3. Receiving DHCP Advertise Messages

After a DHCP client has received a DHCP Advertise message, it has the address of a DHCP server for subsequent DHCP Request messages. If the 'S' bit is zero, the DHCP Advertise message was transmitted by a DHCP server on the same link as the client, and the client uses the Agent address as the address of a DHCP server; otherwise, the DHCP server address is located in the server address field. If the server's address is shown as a Multicast address, the advertisement MUST be silently discarded.

A DHCP server MAY append extensions to its Advertisements; this might allow the DHCP client to select the configuration that best meets its needs from among several prospective servers.

Unless a DHCP Advertisement is received with a "server preference" field of 0xffffffff, a DHCP client must wait ADV_CLIENT_WAIT seconds after issuing the DHCP Solicit message in order to receive the Advertisement with the highest preference. After waiting for that

period of time, a client MUST select the highest preference DHCP server as the target of its DHCP request. If a DHCP client receives an advertised preference of 0xffffffff, it does not have to wait for any more advertisements.

If a DHCP client sends a DHCP Request to a highly preferred DHCP server but fails to receive a DHCP reply from that server after following the retransmission algorithm in section 8, the client may subsequently attempt to send a DHCP Request to a less preferred server.

A DHCP client is free to cache the result of any DHCP Advertisement it hears. However, it should be noted that this is purely a potential performance enhancement as the results need not be constant over time, hence it may not get a response if it uses the address obtained from this message and may have to emit its own DHCP Solicit message subsequently.

5.4. Sending DHCP Request Messages

A DHCP client obtains configuration information from a DHCP server by sending a DHCP Request message. The client MUST know the server's address before sending the Request message, and the client MUST have acquired a (possibly identical) DHCP agent address. If the client and server are on the same link, the agent address used by the client MUST be the same as the DHCP server's address. A DHCP Request message MUST NOT be sent to any multicast address, since otherwise multiple DHCP agents would possibly allocate resources to the client in response to the same Request, and the client would have no way to know which servers had made the allocations, if any packets were lost due to collisions, etc.

If the DHCP server is off-link, and the client has no valid IP address of sufficient scope, then the client MUST include the server address in the appropriate field and set the 'S' bit in the DHCP Request message. In this case, the IP destination address in the IP header will be a DHCP relay address.

Otherwise, if the client already has a valid IP address of sufficient scope and knows the IP address of a candidate DHCP server, it MUST send the Request message directly to the DHCP server without requiring the services of the local DHCP relay.

If a client wishes to instruct a DHCP server to deallocate all unknown previous resources, configuration information, and bindings associated with its agent address and link-local address, it sets the 'C' bit in the DHCP Request. A client MAY send in such a Request even when it is no longer attached to the link on which the relay

address is attached. The 'C' bit allows better reclamation of available resources, since otherwise a client might not be able to release resources that it has no record of using.

In any case, after choosing a transaction-ID which is numerically greater than its previous transaction-ID, and filling in the appropriate fields of the DHCP Request message, the client MAY append various DHCP Extensions to the message. These extensions denote specific requests by the client; for example, a client may request a particular IP address, or request that the server send an update containing the client's new IP address to a Domain Name Server. When all desired extensions have been applied, the DHCP client sends the DHCP Request to the appropriate DHCP Agent.

For each pending DHCP Request message, a client MUST maintain the following information:

- The transaction-ID of the request message,
- The server address,
- The agent address (which can be the same as the server address),
- The time at which the next retransmission will be attempted, and
- All extensions appended to the request message.

If a client does not receive a DHCP Reply message (Section 5.5) with the same transaction-ID as a pending DHCP Request message within REPLY_MSG_TIMEOUT (see section 8) seconds, or if the received DHCP Reply message contains a DHCP Authentication extension which fails to provide the correct authentication information, the client MUST retransmit the Request with the same transaction-ID and continue to retransmit according to the rules in Section 8. If (after following those rules) the client never receives a Reply message, it naturally SHOULD start over again by sending a new DHCP Solicit message to find a different server.

If the client receives an ICMP error message in response to such a DHCP Request, it likewise naturally SHOULD start over again by sending a new DHCP Solicit message, to find a different server.

If the client transmits a DHCP Request in response to a DHCP Reconfigure message (see Section 5.7), the client can continue to operate with its existing configuration information and resources until it receives the corresponding DHCP Reply from the server. The same retransmission rules apply as for any other DHCP Request message from the client. When the 'N' bit is set, a DHCP Request sent in response to a DHCP Reconfigure message will not elicit a DHCP Reply message from the server.

5.5. Receiving DHCP Reply Messages

When a client receives a DHCP Reply message, it MUST check whether the transaction-ID in the Reply message matches the transaction-ID of a pending DHCP Request message. If no match is found, the Reply message MUST be silently discarded.

If the Reply message is acceptable, the client processes each Extension [12], extracting the relevant configuration information and parameters for its network operation. The client can determine when all extensions in the Reply have been processed by using the UDP Length field of the Reply. Some extensions in the Reply may have status codes, which indicate to the client the reason for failure when the server was unable to honor the request. If the server is unable to honor the request in an extension included by the client, that extension may simply be omitted from the Reply. The server MAY also provide the client with configuration parameters the client did not specifically request.

Some configuration information extracted from the extensions to the DHCP Reply message MUST remain associated with the DHCP server that sent the message. The particular extensions that require this extra measure of association with the server are indicated in the DHCP Extensions document [12]. These "resource-server" associations are used when sending DHCP Release messages.

5.6. Sending DHCP Release Messages

If a client wishes to ask the server to release all information and resources relevant to the client, the client SHOULD send a DHCP Release message without any extensions; this is preferable to sending a DHCP Request message with the 'C' bit set and no extensions. If a DHCP client wishes to retain some of its network configuration parameters, but determines that others are no longer needed, it SHOULD enable the DHCP server to release allocated resources which are no longer in use by sending a DHCP Release message to the server, and including extensions to identify the unneeded items. The client consults its list of resource-server associations in order to determine which server should receive the desired Release message.

Suppose a client wishes to release resources which were granted to it on another link, and the client has an IP address with enough scope so that the DHCP server can reach it. In that case, the client MUST instruct the server to send the DHCP Reply directly back to the client at that address, instead of performing the default processing of sending the DHCP Reply back through the agent-address included in the DHCP Release. This is done by setting the 'D' bit in the DHCP

Release message (see section 4.5).

5.7. Receiving DHCP Reconfigure Messages

Each DHCP client implementation MUST support listening at UDP port 546 to receive possible DHCP Reconfigure messages; in cases where the client knows that no Reconfigure message will ever be issued, the client MAY be configured to avoid executing this supported feature. In some cases, the IP address at which the client listens will be a multicast address sent to the client by the DHCP server in an extension to an earlier DHCP Reply message. If the client does not listen for DHCP Reconfigure messages, it is possible that the client will not receive notification that its DHCP server has deallocated the client's IP address and/or other resources allocated to the client. See discussion in 6.5. The client MAY receive a prefix update for one or more of their addresses and then MUST use that prefix for those addresses.

If a DHCP client receives a DHCP Reconfigure message, it is a request for the client to initiate a new DHCP Request/Reply transaction with the server which sent the Reconfigure message. The server sending the Reconfigure message MAY be different than the server which sent a DHCP Reply message containing the original configuration information.

For each Extension which is present in the Reconfigure message, the client MUST append a matching Extension to its Request message, which it formulates to send to the server specified in the server address field of the message. The client also copies a transaction-ID from the Reconfigure message into the Request message. If the 'N' bit is not set, processing from then on is the same as specified above in Section 5.4.

Resources held by the client which are not identified by Extensions in the server's Reconfigure message are not affected.

Note that a server may ask its client to join a multicast group for the purpose of receiving DHCP Reconfigure messages. When a Reconfigure message is delivered to the client by way of the selected multicast address, the client MUST delay its further response for a random amount of time uniformly distributed within the interval between RECONF_MSG_MIN_RESP and RECONF_MSG_MAX_RESP seconds (see section 8). This will minimize the likelihood that the server will be flooded with DHCP Request messages.

5.8. Interaction with Stateless Address Autoconfiguration

If a client begins to receive Router Advertisements [11] with the 'M' bit no longer set (so that the link is no longer a "managed" link), then the client SHOULD send one or more DHCP Release messages to deallocate the IP addresses it has received. The client sends the

DHCP Release messages to the servers recorded in the resource-server associations maintained for the IP addresses.

If a client is receiving Router Advertisements with the 'M' bit equal to 0, then it is using IP addresses obtained by way of Stateless Address Autoconfiguration [16]. In this case, if the client receives a DHCP Reconfigure message, the client MUST return a DHCP Request message to the DHCP server with the 'N' bit set.

Suppose a client, after using Stateless Address Autoconfiguration to configure its stateless IP address, begins to receive Router Advertisements with the 'M' bit set. Then the client MAY continue to use its existing IP addresses, but it MUST send a DHCP Solicit message to discover a DHCP server. If the solicitation results in the receipt of of a DHCP Advertisement message from a DHCP server, the client MUST subsequently transact a DHCP Request message with at least one such DHCP server.

6. DHCP Server Considerations

A node which is not a DHCP client or DHCP relay MUST ignore any DHCP Advertise, DHCP Reply, or DHCP Reconfigure message it receives.

A server maintains a collection of client records, called ``bindings''. Each binding is uniquely identifiable by the ordered pair <link-local address, agent address prefix>, since the link-local address is guaranteed to be unique [16] on the link identified by the agent address. An implementation MUST support bindings consisting of at least a client's link-local address, agent address prefix, preferred lifetime and valid lifetime [16] for each client address, and the transaction-ID. A client binding may be used to store any other information, resources, and configuration data which will be associated with the client. A DHCP server MUST retain its clients' bindings across server reboots, and, whenever possible, a DHCP client should be assigned the same configuration parameters despite server system reboots and DHCP server program restarts. A DHCP server MUST support fixed or permanent allocation of configuration parameters to specific clients.

6.1. Receiving DHCP Solicit Messages

If the DHCP Solicit message was received at the All-DHCP-Servers multicast address, the DHCP server MUST check to make sure that the relay address is present, and not a link-local address. If the relay address is not present, or if it is a link-local address, the server MUST silently discard the packet. Note that if the client sends a DHCP Solicit message from a link-local address, the

multicast destination will be the All-DHCP-Agents address, not the All-DHCP-Servers address.

As an optimization, a server processing a Solicit message from relays MAY check the prefix of the IP source address in the IP header to determine whether the server has received the Solicit from multiple relays on the same link.

6.2. Sending DHCP Advertise Messages

Upon receiving and verifying the correctness of a DHCP Solicit message, a server constructs a DHCP Advertise message and transmits it on the same link as the solicitation was received from. When the solicitation is received at the DHCP Servers multicast address, the server SHOULD delay the transmission of its advertisement for a random amount of time between SERVER_MIN_ADV_DELAY and SERVER_MAX_ADV_DELAY (see section 8).

If the relay address is nonzero, the server MUST put the relay address in the agent address field of the advertisement message, and MUST send the advertisement message to the relay address; otherwise, the server MUST send the advertisement to the client's link-local address. An IP address of the interface on which the server received the Solicit message MUST appear in the server address field of the corresponding advertisement.

The DHCP server MAY append extensions to the Advertisement, in order to offer the soliciting node the best possible information about the services and resources which the server may be able to make available.

6.3. DHCP Request and Reply Message Processing

The DHCP server MUST check to ensure that the client's link-local address field of the Request message contains a link-local address. If not, the message MUST be silently discarded. If the 'S' bit is set, the server MUST check that the server address matches the destination IP address at which the Request message was received by the server. If the server address does not match, the Request message MUST be silently discarded.

If the received agent address and link-local address do not correspond to any binding known to the server, then the server MAY create a new binding for the previously unknown client, and send a DHCP Reply with any resources allocated to the new binding. Otherwise, if the server cannot create a new binding, it SHOULD return a DHCP Reply with a status of 20. If the client is updating

its resources but the database is temporarily unavailable, the server SHOULD return a DHCP Reply with a status of 19.

While processing the Request, the server MUST first determine whether or not the Request is a retransmission of an earlier DHCP Request from the same client. This is done by comparing the transaction-ID to all those transaction-IDs received from the same client during the previous XID_TIMEOUT seconds. If the transaction-ID is the same as one received during that time, the server MUST take the same action (e.g., retransmit the same DHCP Reply to the client) as it did after processing the previous DHCP Request with the same transaction-ID.

Otherwise, if the server has no record of a message from the client with the same transaction-ID, the server identifies and allocates all the relevant information, resources, and configuration data that is associated with the client. Then it sends that information to its DHCP client by constructing a DHCP Reply message and including the client's information in DHCP Extensions to the Reply message. The DHCP Reply message uses the same transaction-ID as found in the received DHCP Request message. Note that the reply message MAY contain information not specifically requested by the client.

If the DHCP Request message has the 'S' bit set in the message header, the DHCP server MUST send the corresponding DHCP Reply message to the agent address found in the Request (see section 7.2). Otherwise, the server SHOULD send the corresponding DHCP Reply message to the IP source address in the IP header received from the client Request message.

6.3.1. Processing for Extensions to DHCP Request and Reply Messages

The DHCP Request may contain extensions [12], which are interpreted (by default) as advisory information from the client about its configuration preferences. For instance, if the IP Address Extension is present, the DHCP server SHOULD attempt to allocate or extend the lifetime of the address indicated by the extension. Some extensions may be marked by the client as required.

The DHCP server may accept some extensions for successful processing and allocation, while still rejecting others, or the server may reject various extensions for different reasons. The server sets the status appropriately for those extensions which return status to the client. The DHCP server sends a single Reply message in response to each DHCP Request, with the same transaction-ID as the Request.

Whenever it is able to, the server includes an extension in the Reply message for every extension sent by the client in the Request message. If the client requests some extensions that cannot be

supplied by the server, the server can simply fail to provide them, not including them in the Reply. Other extensions can be rejected by including them in the Reply with an appropriate status indicating failure. The server can include extensions in the reply that were not requested by the client.

6.3.2. Client Requests to Deallocate Unknown Resources

When a client DHCP Request is received that has the 'C' bit set, the server should check to find out whether the extensions listed in the Request message match those which it has associated with the client's binding. Any resources which are not indicated by the client are presumed to be unknown by the client, and thus possible candidates for reallocation to satisfy requests from other clients. The DHCP server MUST deallocate all resources associated with the client upon reception of a DHCP Request with the 'C' bit set, except for those which the server is willing to reallocate in response to the client's request. It may be more efficient to avoid deallocating any resources until after the list of extensions to the request have been inspected.

6.4. Receiving DHCP Release Messages

If the server receives a DHCP Release Message, it MUST verify that the link-local address field of the message contains an address which could be a valid link-local address (see Section 2.1). If not, the message MUST be silently discarded.

In response to a DHCP Release Message with a valid client's link-local address and agent address, the DHCP server formulates a DHCP Reply message that will be sent back to the releasing client by way of the client's link-local address. A DHCP Reply message sent in response to a DHCP Release message MUST be sent to the client's link-local address via the agent address in the Release message with the 'L' bit set in the Reply, unless the 'D' bit is set in the Release message.

If the received agent address and link-local address do not correspond to any binding known to the server, then the server SHOULD return a DHCP Reply, indicating the error by setting the status code to 20.

Otherwise, if the agent address and link-local address indicate a binding known to the server, then the server continues processing the Release message. If there are any extensions, the server releases the particular configuration items specified in the extensions.

If there are no extensions, the server releases all configuration information in the client's binding.

After performing the operations indicated in the DHCP Release message and its extensions, the DHCP server formulates a DHCP Reply message, copying the transaction-ID from the DHCP Release message. For each Extension in the DHCP Release message successfully processed by the server, a matching Extension is appended to the DHCP Reply message. For extensions in the DHCP Release message which cannot be successfully processed by the server, a DHCP Reply message containing extensions with the appropriate status MUST be returned by the server. If the Release message contains no extensions, the server does not include any extensions in the corresponding DHCP Reply message to the client.

6.5. Sending DHCP Reconfigure Messages

If a DHCP server needs to change the configuration associated with any of its clients, it constructs a DHCP Reconfigure message and sends it to each such client. The Reconfigure MAY be sent to a multicast address chosen by the server and previously sent to each of its clients in an extension to a previous DHCP Reply message.

Often, it happens that a client does not send DHCP Request messages after the DHCP Reconfigure message has been issued and retransmitted according to the algorithm specified in Section 8. This can happen when the cleint is not listening for the Reconfigure message, possibly because the client is a mobile node disconnected from the network, or because the client node has sustained a power outage or operating system crash. In such cases, the DHCP server SHOULD reserve any resources issued to the client until the client responds at some future time, or until administrative intervention causes the resources to be manually returned to use.

7. DHCP Relay Considerations

The DHCP protocol is constructed so that a relay does not have to maintain any state in order to mediate DHCP client/server interactions.

All relays MUST send DHCP Request messages using the source IP address from the interface where the DHCP request was received.

The purpose of the DHCP relay is to enable clients and servers to carry out DHCP protocol transactions. DHCP Solicit messages are issued by the relay when initiated by prospective DHCP clients. By default, the relay locates DHCP servers by use of multicasting DHCP

solicitations to the All-DHCP-Servers multicast address, but relays SHOULD allow this behavior to be configurable. The relay SHOULD NOT send such a multicast solicitation on the interface from which it received the solicitation from the client. The source address must be a site-local or global-scope address belonging to the relay's interface on which the client's original solicitation was received.

7.1. DHCP Solicit and DHCP Advertise Message Processing

Upon receiving a DHCP Solicit message from a prospective client, a relay, by default, forwards the message to all DHCP servers at a site according to the following procedure:

- copying the prospective client's solicitation message fields into the appropriate fields of the outgoing solicitation,

- copying a non-link-local address of its interface from which the solicitation was received from the client into the DHCP relay address field, and

- by default, setting the TTL field in the solicitation to the value DEFAULT_SOLICIT_TTL (see section 8).

- finally, sending the resulting message to the All-DHCP-Servers multicast address, FF05:0:0:0:0:0:1:3.

When the relay receives a DHCP advertisement, it relays the advertisement to the client at the client's link-local address by way of the interface indicated in the agent's address field.

7.2. DHCP Request Message Processing

When a relay receives a DHCP Request message, it SHOULD check that the IP source address in the IP header is a link-local address, that the link-local address matches the link-local address field in the Request message header, and that the agent address field of the message matches an IP address associated with the interface from which the DHCP Request message was received. If any of these checks fail, the relay MUST silently discard the Request message.

The relay MUST check whether the 'S' bit is set in the message header. If not, the packet is discarded, and the relay SHOULD return a DHCP Reply message to the address contained in the client's link-local address field of the Request message, with status 18.

If the received request message is acceptable, the relay then transmits the DHCP Request message to the address of the DHCP server

found in the Server IP Address field of the received DHCP Request message. All of the fields of DHCP Request message transmitted by the relay are copied over unchanged from the DHCP Request received from the client. Only the fields in the IP header will differ from the packet received from the client. If the Relay receives an ICMP error, the Relay SHOULD return a DHCP Reply message to the client address (which can be found in the payload of the ICMP message [5]), with status 64.

7.3. DHCP Reply Message Processing

When the relay receives a DHCP Reply, it MUST check that the message has the 'L' bit set. It MUST check that the link-local address field contains a link-local address. If either check fails, the packet MUST be silently discarded. If both checks are satisfied, the relay MUST send a DHCP Reply message to the link-local address listed in the received Reply message. Only the fields in the IP header will differ from the packet received from the server, not the payload.

8. Retransmission and Configuration Variables

When a DHCP client does not receive a DHCP Reply in response to a pending DHCP Request, the client MUST retransmit the identical DHCP Request, with the same transaction-ID, to the same server again until it can be reasonably sure that the DHCP server is unavailable and an alternative can be chosen. The DHCP server assumes that the client has received the configuration information included with the extensions to the DHCP Reply message, and it is up to the client to continue to try for a reasonable amount of time to complete the transaction. All the actions specified for DHCP Request in this section hold also for DHCP Release messages, that do not have the 'N' bit set, sent by the DHCP client.

Similarly, when a client sends a DHCP Request message in response to a Reconfigure message from the server, the client assumes that the DHCP server has received the Request. The server MUST retransmit the identical DHCP Reconfigure to the client for a reasonable amount of time, to try to elicit the Request message from the client. If no corresponding DHCP Request is received by the server within this time, the server MAY erase or deallocate information as needed from the client's binding.

When a client reboots and loses its previous state, the server should no longer keep track of the transaction IDs associated with that previous state. In order to inform the server that the client no longer wishes any association to be maintained between used

transaction IDs and previous transactions, the client should set the 'R' bit in its DHCP Request.

Retransmissions occur using the following configuration variables for a DHCP implementation. These configuration variables MUST be configurable by a client or server:

ADV_CLIENT_WAIT

The minimum amount of time a client waits to receive DHCP Advertisements after transmitting a DHCP Solicit to the All-DHCP Agents multicast address.

Default: 2 seconds

DEFAULT_SOLICIT_TTL

The default TTL value used by DHCP relays when sending DHCP Solicit messages on behalf of a client.

Default: 4

SERVER_MIN_ADV_DELAY

The minimum amount of time a server waits to transmit a DHCP Advertisement after receiving a DHCP Solicit at the All-DHCP Servers or All-DHCP Agents multicast address.

Default: 100 milliseconds

SERVER_MAX_ADV_DELAY

The maximum amount of time a server waits to transmit a DHCP Advertisement after receiving a DHCP Solicit at the All-DHCP Agents multicast address.

Default: 1 second

REPLY_MSG_TIMEOUT

The time in seconds that a DHCP client waits to receive a server's DHCP Reply before retransmitting a DHCP Request.

Default: 2 seconds.

REPLY_MSG_MIN_RETRANS

The minimum number of DHCP Request transmissions that a DHCP client should retransmit, before aborting the request, possibly

retrying the Request with another Server, and logging a DHCP System Error.

Default: 10 retransmissions.

REPLY_MSG_RETRANS_INTERVAL

The time between successive retransmissions of DHCP Request messages.

Default: 2 seconds.

RECONF_MSG_TIMEOUT

The time in seconds that a DHCP server waits to receive a client's DHCP Request before retransmitting its DHCP Reconfigure.

Default: 12 seconds.

RECONF_MSG_MIN_RETRANS

The minimum number of DHCP Reconfigure messages that a DHCP server should retransmit, before assuming the the client is unavailable and that the server can proceed with the needed reconfiguration of that client's resources, and logging a DHCP System Error.

Default: 10 retransmissions.

RECONF_MSG_RETRANS_INTERVAL

The least time between successive retransmissions of DHCP Reconfigure messages.

Default: RECONF_MSG_TIMEOUT

RECONF_MSG_MIN_RESP

The minimum amount of time before a client can respond to a DHCP Reconfigure message sent to a multicast address.

Default: 2 seconds.

RECONF_MSG_MAX_RESP

The maximum amount of time before a client MUST respond to a DHCP Reconfigure message sent to a multicast address.

Default: 10 seconds.

MIN_SOLICIT_DELAY

The maximum amount of time a prospective client is required to wait, after determining from a Router Advertisement message that the client should perform stateful address configuration, before sending a DHCP Solicit to a DHCP server.

Default: 1 second

MAX_SOLICIT_DELAY

The maximum amount of time a prospective client is required to wait, after determining from a Router Advertisement message that the client should perform stateful address configuration, before sending a DHCP Solicit to a DHCP server.

Default: 5 seconds

XID_TIMEOUT

The amount of time a DHCP server has to keep track of client transaction-IDs in order to make sure that client retransmissions using the same transaction-ID are idempotent.

Default: 600 seconds

Note that, if a client receives a DHCP message which fails authentication, it should continue to wait for another message which might be correctly authenticated just as if the failed message had never arrived; however, receiving such failed messages SHOULD be logged.

9. Security Considerations

DHCP clients and servers often have to authenticate the messages they exchange. For instance, a DHCP server may wish to be certain that a DHCP Request originated from the client identified by the <link-local address, agent address> fields included within the Request message header. Conversely, it is quite often essential for a DHCP client to be certain that the configuration parameters and addresses it has received were sent to it by an authoritative DHCP server. Similarly, a DHCP server should only accept a DHCP Release message which seems to be from one of its clients, if it has some assurance that the client actually did transmit the Release message. Again, a client might wish to only accept DHCP Reconfigure messages that are certain to have originated from a server with authority to issue them.

The IPv6 Authentication Header can provide security for DHCPv6 messages when both endpoints have a suitable IP address. However, a client often has only a link-local address, and such an address is not sufficient for a DHCP server which is off-link. In those circumstances the DHCP relay is involved, so that the DHCP message MUST have the relay's address in the IP destination address field, even though the client aims to deliver the message to the DHCP server. The DHCP Client-Server Authentication Extension [12] is intended to be used in these circumstances.

10. Year 2000 considerations

Since all times are relative to the current time of the transaction, there is no problem within the DHCPv6 protocol related to any hardcoded dates or two-digit representation of the current year.

11. Acknowledgements

Thanks to the DHC Working Group for their time and input into the specification. Ralph Droms and Thomas Narten have had a major role in shaping the continued improvement of the protocol by their careful reviews. Many thanks to Matt Crawford and Erik Nordmark for their studied review as part of the Last Call process. Thanks also for the consistent input, ideas, and review by (in alphabetical order) Mike Carney, Brian Carpenter, Gerald Maguire, Jack McCann, Yakov Rekhter, Matt Thomas, Sue Thomson, and Phil Wells.

Thanks to Steve Deering and Bob Hinden, who have consistently taken the time to discuss the more complex parts of the IPv6 specifications.

A. Related Work in IPv6

The related work in IPv6 that would best serve an implementor to study is the IPv6 Specification [6], the IPv6 Addressing Architecture [8], IPv6 Stateless Address Autoconfiguration [16], IPv6 Neighbor Discovery Processing [11], and Dynamic Updates to DNS [18]. These specifications enable DHCP to build upon the IPv6 work to provide both robust stateful autoconfiguration and autoregistration of DNS Host Names.

The IPv6 Specification provides the base architecture and design of IPv6. A key point for DHCP implementors to understand is that IPv6 requires that every link in the internet have an MTU of 1500 octets or greater (in IPv4 the requirement is 68 octets). This means that a UDP packet of 536 octets will always pass through an internet

(less 40 octets for the IPv6 header), as long as there are no IP options prior to the UDP header in the packet. But, IPv6 does not support fragmentation at routers, so that fragmentation takes place end-to-end between hosts. If a DHCP implementation needs to send a packet greater than 1500 octets it can either fragment the UDP packet into fragments of 1500 octets or less, or use Path MTU Discovery [10] to determine the size of the packet that will traverse a network path. It is implementation dependent how this is accomplished in DHCP. Path MTU Discovery for IPv6 is supported for both UDP and TCP and can cause end-to-end fragmentation when the PMTU changes for a destination.

The IPv6 Addressing Architecture specification [8] defines the address scope that can be used in an IPv6 implementation, and the various configuration architecture guidelines for network designers of the IPv6 address space. Two advantages of IPv6 are that support for multicast is required, and nodes can create link-local addresses during initialization. This means that a client can immediately use its link-local address and a well-known multicast address to begin communications to discover neighbors on the link, or to send a DHCP Solicit and locate a DHCP server or relay.

IPv6 Stateless Address Autoconfiguration [16] (addrconf) specifies procedures by which a node may autoconfigure addresses based on router advertisements [11], and the use of a valid lifetime to support renumbering of addresses on the Internet. In addition the protocol interaction by which a node begins stateless or stateful autoconfiguration is specified. DHCP is one vehicle to perform stateful autoconfiguration. Compatibility with addrconf is a design requirement of DHCP (see Section 3.1).

IPv6 Neighbor Discovery [11] is the node discovery protocol in IPv6 which replaces and enhances functions of ARP [13]. To understand IPv6 and addrconf it is strongly recommended that implementors understand IPv6 Neighbor Discovery.

Dynamic Updates to DNS [18] is a specification that supports the dynamic update of DNS records for both IPv4 and IPv6. DHCP can use the dynamic updates to DNS to integrate addresses and name space to not only support autoconfiguration, but also autoregistration in IPv6. The security model to be used with DHCPv6 should conform as closely as possible to the authentication model outlined in [9].

B. Comparison between DHCPv4 and DHCPv6

This appendix is provided for readers who will find it useful to see a model and architecture comparison between DHCPv4 [7, 1] and DHCPv6.

There are three key reasons for the differences:

- o IPv6 inherently supports a new model and architecture for communications and autoconfiguration of addresses.

- o DHCPv6 in its design was able to take advantage of the inherent benefits of IPv6.

- o New features were added to support the expected evolution and the existence of more complicated Internet network service requirements.

IPv6 Architecture/Model Changes:

- o The link-local address permits a node to have an address immediately when the node boots, which means all clients have a source IP address at all times to locate a server or relay agent on the local link.

- o The need for bootp compatibility and broadcast flags are removed, which permitted a great deal of freedom in designing the new packet formats for the client and server interaction.

- o Multicast and the scoping methods in IPv6 permitted the design of discovery packets that would inherently define their range by the multicast address for the function required.

- o Stateful autoconfiguration has to coexist and integrate with stateless autoconfiguration supporting Duplicate Address Detection and the two IPv6 lifetimes, to facilitate the dynamic renumbering of addresses and the management of those addresses.

- o Multiple addresses per interface are inherently supported in IPv6.

- o Most DHCPv4 options are unnecessary now because the configuration parameters are either obtained through IPv6 Neighbor Discovery or the Service Location protocol [17].

DHCPv6 Architecture/Model Changes:

- o The message type is the first byte in the packet.

- o IPv6 Address allocations are now handled in a message extension as opposed to the main header.

- o Client/Server bindings are now mandatory and take advantage of the client's link-local address to always permit communications either directly from an on-link server, or from a remote server through an on-link relay-agent.

- Servers are now discovered by a client solicit and server or relay-agent advertisement model.

- The client will know if the server is on-link or off-link.

- The client after a solicit will be returned the addresses of available servers either from an on-link server or from an on-link relay-agent as agents providing the advertisements.

- The on-link relay-agent will obtain the location of remote server addresses from system configuration or by the use of a site wide DHCPv6 Multicast packet.

- The protocol is optimized and removes the use of ACKs and NAKs once the client and server set-up is complete.

- The server assumes the client receives its responses unless it receives a retransmission of the same client request. This permits recovery in the case where the network has faulted.

- The function of DHCPINFORM is inherent in the new packet design; a client can request configuration parameters other than IPv6 addresses in the optional extension headers.

- Clients MUST listen to their UDP port for the new Reconfigure message from servers.

- Dynamic Updates to DNS are supported in the IPv6 Address extension.

- New extensions have been defined.

New Internet User Features:

- Configuration of Dynamic Updates to DNS to support different requirements.

- Configuration of what policy is enforced when addresses are deprecated for dynamic renumbering can be implemented.

- Configuration of how relay-agents locate remote servers for a link can be implemented.

- An Authentication extension has been added.

- Configuration of additional addresses for server applications can be requested by a client in an implementation.

- Reclaiming addresses allocated with very long lifetimes can be implemented using the Reconfigure message type.

- Configuration of tightly coupled integration between stateless and stateful address autoconfiguration can be implemented.

References

[1] S. Alexander and R. Droms. DHCP Options and BOOTP Vendor Extensions. RFC 2132, March 1997.

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

[3] S. Bradner and A. Mankin. The Recommendation for the IP Next Generation Protocol. RFC 1752, January 1995.

[4] William R. Cheswick and Steven Bellovin. Firewalls and Internet Security. Addison-Wesley, Reading, Massachusetts, 1994. (ISBN: 0-201-63357-4).

[5] A. Conta and S. Deering. Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6). RFC 1885, December 1995.

[6] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. RFC 1883, December 1995.

[7] R. Droms. Dynamic Host Configuration Protocol. RFC 2131, March 1997.

[8] R. Hinden and S. Deering. IP Version 6 Addressing Architecture. RFC 1884, December 1995.

[9] Stephen Kent and Randall Atkinson. IP Authentication Header. draft-ietf-ipsec-auth-header-01.txt, July 1997. (work in progress).

[10] J. McCann, S. Deering, and J. Mogul. Path MTU Discovery for IP version 6. RFC 1981, August 1996.

[11] T. Narten, E. Nordmark, and W. Simpson. Neighbor Discovery for IP version 6 (IPv6). RFC 1970, August 1996.

[12] C. Perkins. Extensions for the Dynamic Host Configuration Protocol for IPv6. draft-ietf-dhc-dhcpv6ext-09.txt, October 1997. (work in progress).

[13] David C. Plummer. An Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Addresses for Transmission on Ethernet Hardware. RFC 826, November 1982.

[14] J. B. Postel. User Datagram Protocol. RFC 768, August 1980.

[15] J. B. Postel, Editor. Internet Protocol. RFC 791, September 1981.

[16] S. Thomson and T. Narten. IPv6 stateless address autoconfiguration. RFC 1971, August 1996.

[17] J. Veizades, E. Guttman, C. Perkins, and S. Kaplan. Service Location Protocol. RFC 2165, July 1997.

[18] P. Vixie, S. Thomson, Y. Rekhter, and J. Bound. Dynamic Updates in the Domain Name System (DNS). RFC 2136, April 1997.

Chair's Address

The working group can be contacted via the current chair:

Ralph Droms
Computer Science Department
323 Dana Engineering
Bucknell University
Lewisburg, PA 17837

Phone: (717) 524-1145
E-mail: droms@bucknell.edu

Author's Address

Questions about this memo can be directed to:

Jim Bound
Digital Equipment Corporation
110 Spitbrook Road, ZKO3-3/U14
Nashua, NH 03062

Phone: +1-603-884-0400
Fax:
E-mail: bound@zk3.dec.com

Charles Perkins
Technology Development
Sun Microsystems, Inc.
901 San Antonio Rd.
Palo Alto, CA 94303
+1-650-786-6464
+1-650-786-6445
charles.perkins@sun.com