

DHC Working Group
Internet-Draft
Intended status: Standards Track
Expires: 26 October 2025

C. Lin
New H3C Technologies
24 April 2025

DHCPv6 Active Leasequery over QUIC
draft-lin-dhc-dhcpv6-active-leasequery-quic-00

Abstract

RFC7653 allows for actively transferring the real-time binding information data of Dynamic Host Configuration Protocol for IPv6 (DHCPv6) via TCP, which satisfies the need of continuous update of an external requestor with Leasequery data. QUIC could provide useful, reliable and secure semantics for transferring active leasequery of DHCPv6, which enabling much better efficiency and performance by reducing connect time. This document describes how to use the QUIC transport protocol to deliver DHCPv6 Active Leasequery messages, named DHCP6oQUIC.

Status of This Memo

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

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

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

This Internet-Draft will expire on 26 October 2025.

Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components

extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
2. Terminology and Definitions	3
3. Connection Management	4
3.1. Connection Establishment	4
3.2. Connection Termination	4
3.2.1. QUIC Connection Termination Process	4
3.2.2. DHCP6oQUIC Considerations for Connection Termination	4
4. Stream Mapping and Usage	5
4.1. Bidirectional Stream from Requestor to Server	5
5. Endpoint Authentication	5
6. Operational Considerations	6
7. IANA Considerations	6
8. Security Considerations	6
9. References	6
9.1. Normative References	6
9.2. Informative References	7
Author's Address	7

1. Introduction

In order to allow an entity indirectly involved in DHCPv6 client-server transactions to keep current with the state of the DHCPv6 lease state information in real time, the Active Leasequery capability is proposed in [RFC7653].

An Active Leasequery requestor creates a TCP connection to a DHCPv6 server by using the DHCPv6 port 547. After the TCP connection is established, the messages of Active Leasequery mechanism will delivered between requestor and server.

Active Leasequery allows the external entity to lose its connection with the DHCPv6 server, and then reconnect and receive the latest information regarding any IPv6 bindings changed when it was disconnected. When using TCP for transmission of Active Leasequery protocol, it allows the server to send one or more DHCPv6 messages in response to a single Active Leasequery request.

QUIC [RFC9000] is UDP-based multiplexed and secure transport protocol that provides connection-oriented and stateful interaction between a client and server. It can provide low latency and encrypted transport with resilient connections. QUIC uses multiple

simultaneous streams to carry data in one direction. Each stream is a separate unidirectional or bidirectional channel consisting of an ordered stream of bytes. In Addition, each stream has its own flow control, which limit bytes sent on a stream, together with flow control of the connection. Moreover, QUIC does not have the TCP shortcomings such as head of line blocking.

Compared to the TCP transport protocols [RFC9293] supported by DHCPv6 [RFC7653], QUIC offers the following advantages:

- * QUIC provides reliability and congestion control similar to TCP [RFC9293]. It is more universally used and supported across the global Internet environment because QUIC is a UDP-based protocol.
- * QUIC has built-in TLS encryption (typically TLS 1.3), offering end-to-end security to ensure data confidentiality and integrity. In contrast, with TCP [RFC9293], such encryption requires additional protocols like TLS.
- * QUIC supports multi-streaming, allowing multiple data streams to be multiplexed within a single connection. Each stream transmits data independently, avoiding the head-of-line blocking issue found in TCP [RFC9293].
- * QUIC has lower latency. Since QUIC combines connection establishment and encryption handshake, it only requires 1 Round-Trip Time (RTT), while TCP typically needs 1-2 RTTs.

Therefore, QUIC is a proper secure and reliable transport protocol for the Active Leasequery mechanism of DHCPv6. This document specifies how to use QUIC as the transport protocol for Active Leasequery Protocol.

2. Terminology and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

In this document, the terms "client" and "server" are used to refer to the two ends of the QUIC connection. The client actively initiates the QUIC connection. The terms "Requestor" is used to refer to the initiating end of the DHCPv6 Active Leasequery Protocol session. Generally, a "Requestor" is a "Client".

- * Client: The endpoint that initiates a QUIC connection, the DHCPv6 Requestor.

3. Connection Management

3.1. Connection Establishment

QUIC connection establishment is described in [RFC9000]. During establishing connection, DHCP6oQUIC support is indicated by selecting the Application-Layer Protocol Negotiation (ALPN) [RFC7301] token as listed in the IANA Section 7 in the TLS handshake.

The DHCPv6 Requestor MUST act as the client, and The DHCPv6 Requestor should be the initiator of the QUIC connection to the DHCPv6 server meanwhile the DHCPv6 server acts as the connection acceptor.

The DHCPv6 Requestor MAY record an alarm if the underlying QUIC connection establishment time out; this timeout should be configurable on the DHCPv6 Requestor.

If the DHCPv6 Server does not acknowledge an attempt by the DHCPv6 Requestor to establish a connection, QUIC will automatically retry connection establishment using exponential backoff.

3.2. Connection Termination

3.2.1. QUIC Connection Termination Process

The typical QUIC connection termination process is described in [RFC9000].

3.2.2. DHCP6oQUIC Considerations for Connection Termination

When the Active Leasequery protocol is implemented based on a QUIC connection, the idle timeout should be disabled or the QUIC `max_idle_timeout` should be set appropriately in order to keep the QUIC connection persistent even if the Active Leasequery session is idle.

When a DHCPv6 Requestor is shut down, it SHOULD shut down the QUIC connection.

The requestor or DHCPv6 Leasequery server MAY close its end of the QUIC connection at any time. The DHCPv6 server might decide to close the connection based on its own configuration.

When a DHCPv6 Requestor is detecting the abnormal interruption of the QUIC connection to the DHCPv6 Leasequery server, it SHOULD try to re-establish the connection. Connection timeouts and retry schedules SHOULD be configurable. The default configuration is that a DHCPv6 Requestor MUST NOT attempt to establish a connection more frequently than one every ACTIVE_LQ_IDLE_TIMEOUT, according to the section 6.4 of [RFC7653].

4. Stream Mapping and Usage

QUIC [RFC9000] uses multiple simultaneous streams to carry data in one direction. QUIC Streams provide a lightweight, ordered byte-stream abstraction to an application. Streams can be unidirectional or bidirectional meanwhile streams can be initiated by either the client or the server. Unidirectional streams carry data in one direction: from the initiator of the stream to its peer. Bidirectional streams allow for data to be sent in both directions.

QUIC uses Stream ID to identify the stream. The least significant bit (0x1) of the stream ID identifies the initiator of the stream (client with the bit set to 0). The second least significant bit (0x2) of the stream ID distinguishes between bidirectional streams (with the bit set to 0) and unidirectional streams [RFC9000].

4.1. Bidirectional Stream from Requestor to Server

According to the Active Leasequery mechanism defined in [RFC7653], after the requestor sends an ACTIVELEASEQUERY message over the connection, the server sends updates to the requestor using LEASEQUERY-REPLY and LEASEQUERY-DATA messages in response to the request. Therefore, the Active Leasequery connection is bidirectional from requestor to server.

Based on the above description, The Active Leasequery messages are initiated by the requestor and reply is needed from the server. So the Active Leasequery messages MAY be mapped into one bidirectional stream whose stream type is 0x00 according to section 2.1 of [RFC9000].

5. Endpoint Authentication

DHCP6oQUIC uses QUIC which uses TLS version 1.3 or greater. Therefore, the TLS handshake process can be used for DHCP6oQUIC endpoint authentication. A third-party authentication mechanism can also be applied for DHCP6oQUIC endpoint authentication, such as a TLS client certificate.

6. Operational Considerations

The decision to use DHCP6oQUIC instead of the TCP-based mechanism in [RFC7653] is an operational decision, and an implementation **MUST** provide a configuration mechanism to enable DHCP6oQUIC on the Active Leasequery session.

Some connectivity problems (such as blocking UDP [RFC768]) could result in a failure to establish a QUIC connection. When this happens, the requestor **SHOULD** attempt to establish an Active Leasequery session via TCP.

7. IANA Considerations

This document creates a new registration for the identification of DHCP6oQUIC in the "Application Layer Protocol Negotiation (ALPN) Protocol IDs" registry established in [RFC7301].

The "DHCP6oQ" string identifies DHCP6oQUIC:

- * Protocol: DHCP6oQUIC
- * Identification Sequence: 0x44 0x48 0x43 0x50 0x36 0x6f 0x51 ("DHCP6oQ")
- * Specification: This document

In addition, it is requested for IANA to reserve a UDP port TBD for 'DHCPv6 over QUIC'.

8. Security Considerations

This document replaces the transport protocol layer of DHCPv6 Active Leasequery from TCP to QUIC. The basic protocol specification of DHCPv6 Active Leasequery is not modified, and therefore the new security risks are not introduced to the basic DHCPv6 Active Leasequery protocol. DHCP6oQUIC enhances transport-layer security for DHCPv6 Active Leasequery session according to [RFC9000].

This document does not require to support third-party authentication (e.g., backend Authentication) due to the fact that TLS does not specify this way of authentication. If third-party authentication is needed, TLS client certificates are recommended to be used here.

9. References

9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC7653] Raghuvanshi, D., Kinnear, K., and D. Kukrety, "DHCPv6 Active Leasequery", RFC 7653, DOI 10.17487/RFC7653, October 2015, <<https://www.rfc-editor.org/info/rfc7653>>.
- [RFC768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC0768, August 1980, <<https://www.rfc-editor.org/info/rfc768>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <<https://www.rfc-editor.org/info/rfc9000>>.
- [RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <<https://www.rfc-editor.org/info/rfc9293>>.

9.2. Informative References

- [RFC7301] Friedl, S., Popov, A., Langley, A., and E. Stephan, "Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension", RFC 7301, DOI 10.17487/RFC7301, July 2014, <<https://www.rfc-editor.org/info/rfc7301>>.

Author's Address

Changwang Lin
New H3C Technologies
8 Yongjia North Road
Beijing
Haidian District, 100094
China
Email: linchangwang.04414@h3c.com