Stream:	Internet Engineering Task Force (IETF)
RFC:	9443
Updates:	5764, 7983
Category:	Standards Track
Published:	July 2023
ISSN:	2070-1721
Authors:	B. Aboba	G. Salgueiro	C. Perkins
---	---	---	---
	Microsoft Corporation	*Cisco Systems*	*University of Glasgow*

RFC 9443
Multiplexing Scheme Updates for QUIC

Abstract

RFC 7983 defines a scheme for a Real-time Transport Protocol (RTP) receiver to demultiplex Datagram Transport Layer Security (DTLS), Session Traversal Utilities for NAT (STUN), Secure Real-time Transport Protocol (SRTP) / Secure Real-time Transport Control Protocol (SRTCP), ZRTP, and Traversal Using Relays around NAT (TURN) channel packets arriving on a single port. This document updates RFC 7983 and RFC 5764 to also allow QUIC packets to be multiplexed on a single receiving socket.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9443.

Copyright Notice

Table of Contents

1. Introduction

"Multiplexing Scheme Updates for Secure Real-time Transport Protocol (SRTP) Extension for Datagram Transport Layer Security (DTLS)" [RFC7983] defines a scheme for a Real-time Transport Protocol (RTP) [RFC3550] receiver to demultiplex DTLS [RFC9147], Session Traversal Utilities for NAT (STUN) [RFC8489], Secure Real-time Transport Protocol (SRTP) / Secure Real-time Transport Control Protocol (SRTCP) [RFC3711], ZRTP [RFC6189], and Traversal Using Relays around NAT (TURN) channel packets arriving on a single port. This document updates [RFC7983] and "Datagram Transport Layer Security (DTLS) Extension to Establish Keys for the Secure Real-time Transport Protocol (SRTP)" [RFC5764] to also allow QUIC [RFC9000] to be multiplexed on the same port.

The multiplexing scheme described in this document supports multiple use cases. In the WebRTC scenarios described in [P2P-QUIC] and [P2P-QUIC-TRIAL], SRTP transports audio and video while peer-to-peer QUIC is used for data exchange. For this use case, SRTP [RFC3711] is keyed using DTLS-SRTP [RFC5764]; therefore, SRTP/SRTCP [RFC3550], STUN, TURN, DTLS, and QUIC need to be multiplexed on the same port. Were SRTP to be keyed using QUIC-SRTP (not yet specified), SRTP/SRTCP, STUN, TURN, and QUIC would need to be multiplexed on the same port. Where QUIC is used for peer-to-peer transport of data as well as RTP/RTCP [RTP-OVER-QUIC], STUN, TURN, and QUIC need to be multiplexed on the same port.

While the scheme described in this document is compatible with QUIC version 2 [RFC9369], it is not compatible with QUIC bit greasing [RFC9287]. As a result, endpoints that wish to use multiplexing on their socket **MUST NOT** send the grease_quic_bit transport parameter.

1.1. Terminology

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.

2. Multiplexing of TURN Channels

TURN channels are an optimization where data packets are exchanged with a 4-byte prefix instead of the standard 36-byte STUN overhead (see Section 3.5 of [RFC8656]). [RFC7983] allocates the values from 64 to 79 in order to allow TURN channels to be demultiplexed when the TURN client does the channel binding request in combination with the demultiplexing scheme described in [RFC7983].

In the absence of QUIC bit greasing, the first octet of a QUIC packet (e.g. a short header packet in QUIC v1 or v2) may fall in the range 64 to 127, thereby overlapping with the allocated range for TURN channels of 64 to 79. However, in practice this overlap does not represent a problem. TURN channel packets will only be received from a TURN server to which TURN allocation and channel-binding requests have been sent. Therefore, a TURN client receiving packets from the source IP address and port of a TURN server only needs to disambiguate STUN (i.e., regular TURN) packets from TURN channel packets; (S)RTP, (S)RTCP, ZRTP, DTLS, or QUIC packets will not be sent from a source IP address and port that had previously responded to TURN allocation or channel-binding requests.

As a result, if the source IP address and port of a packet do not match that of a responding TURN server, a packet with a first octet of 64 to 127 can be unambiguously demultiplexed as QUIC.

3. Updates to RFC 7983

This document updates the text in Section 7 of [RFC7983] (which in turn updates [RFC5764]) as follows:

OLD TEXT

> The process for demultiplexing a packet is as follows. The receiver looks at the first byte of the packet. If the value of this byte is in between 0 and 3 (inclusive), then the packet is STUN. If the value is between 16 and 19 (inclusive), then the packet is ZRTP. If the value is between 20 and 63 (inclusive), then the packet is DTLS. If the value is between 64 and 79 (inclusive), then the packet is TURN Channel. If the value is in between 128 and 191 (inclusive), then the packet is RTP (or RTCP, if both RTCP and RTP are being multiplexed

over the same destination port). If the value does not match any known range, then the packet **MUST** be dropped and an alert **MAY** be logged. This process is summarized in Figure 3.

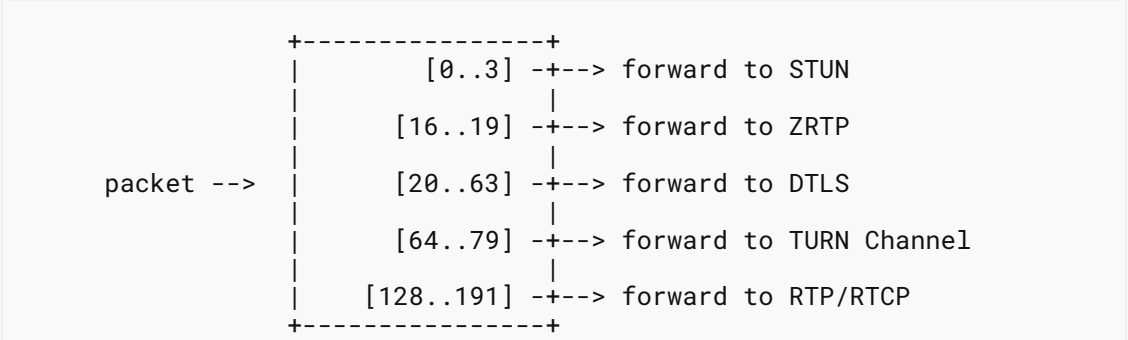

Figure 3: The DTLS-SRTP receiver's packet demultiplexing algorithm.

END OLD TEXT

NEW TEXT

The process for demultiplexing a packet is as follows. The receiver looks at the first byte of the packet. If the value of this byte is between 0 and 3 (inclusive), then the packet is STUN. If the value is between 16 and 19 (inclusive), then the packet is ZRTP. If the value is between 20 and 63 (inclusive), then the packet is DTLS. If the value is between 128 and 191 (inclusive), then the packet is RTP (or RTCP, if both RTCP and RTP are being multiplexed over the same destination port). If the value is between 80 and 127 (inclusive) or between 192 and 255 (inclusive), then the packet is QUIC. If the value is between 64 and 79 (inclusive) and the packet has a source IP address and port of a responding TURN server, then the packet is TURN channel; if the source IP address and port are not that of a responding TURN server, then the packet is QUIC.

If the value does not match any known range, then the packet **MUST** be dropped and an alert **MAY** be logged. This process is summarized in Figure 3.

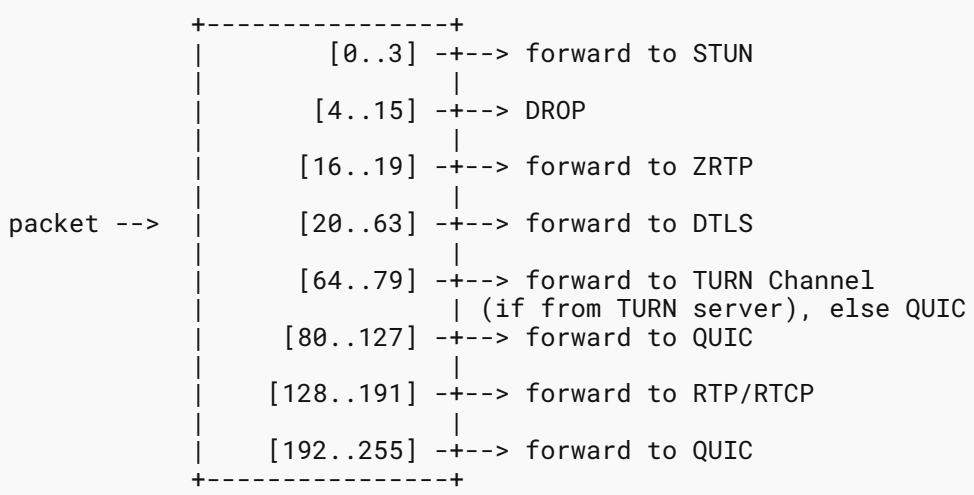

Figure 3: The receiver's packet demultiplexing algorithm.

Note: Endpoints that wish to demultiplex QUIC **MUST NOT** send the grease_quic_bit transport parameter, as described in [RFC9287].

END NEW TEXT

4. Security Considerations

The solution discussed in this document could potentially introduce some additional security issues beyond those described in [RFC7983]. These additional concerns are described below.

In order to support multiplexing of QUIC, this document adds logic to the scheme defined in [RFC7983]. If misimplemented, the logic could potentially misclassify packets, exposing protocol handlers to unexpected input.

When QUIC is used solely for data exchange, the TLS-within-QUIC exchange [RFC9001] derives keys used solely to protect QUIC data packets. If properly implemented, this should not affect the transport of SRTP or the derivation of SRTP keys via DTLS-SRTP. However, if a future specification were to define use of the TLS- within-QUIC exchange to derive SRTP keys, both transport and SRTP key derivation could be adversely impacted by a vulnerability in the QUIC implementation.

5. IANA Considerations

In the "TLS ContentType" registry, IANA replaced references to [RFC7983] with references to this document.

6. References

6.1. 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>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/info/rfc3550>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <https://www.rfc-editor.org/info/rfc3711>.

[RFC5764] McGrew, D. and E. Rescorla, "Datagram Transport Layer Security (DTLS) Extension to Establish Keys for the Secure Real-time Transport Protocol (SRTP)", RFC 5764, DOI 10.17487/RFC5764, May 2010, <https://www.rfc-editor.org/info/rfc5764>.

[RFC7983] Petit-Huguenin, M. and G. Salgueiro, "Multiplexing Scheme Updates for Secure Real-time Transport Protocol (SRTP) Extension for Datagram Transport Layer Security (DTLS)", RFC 7983, DOI 10.17487/RFC7983, September 2016, <https://www.rfc-editor.org/info/rfc7983>.

[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>.

[RFC8489] Petit-Huguenin, M., Salgueiro, G., Rosenberg, J., Wing, D., Mahy, R., and P. Matthews, "Session Traversal Utilities for NAT (STUN)", RFC 8489, DOI 10.17487/RFC8489, February 2020, <https://www.rfc-editor.org/info/rfc8489>.

[RFC8656] Reddy, T., Ed., Johnston, A., Ed., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC 8656, DOI 10.17487/RFC8656, February 2020, <https://www.rfc-editor.org/info/rfc8656>.

[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>.

[RFC9001] Thomson, M., Ed. and S. Turner, Ed., "Using TLS to Secure QUIC", RFC 9001, DOI 10.17487/RFC9001, May 2021, <https://www.rfc-editor.org/info/rfc9001>.

[RFC9147] Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", RFC 9147, DOI 10.17487/RFC9147, April 2022, <https://www.rfc-editor.org/info/rfc9147>.

[RFC9287] Thomson, M., "Greasing the QUIC Bit", RFC 9287, DOI 10.17487/RFC9287, August 2022, <https://www.rfc-editor.org/info/rfc9287>.

6.2. Informative References

[P2P-QUIC] Thatcher, P., Aboba, B., and R. Raymond, "QUIC API For Peer-to-Peer Connections", W3C Community Group Draft Report, commit 50d79c0, 20 May 2023, <https://www.w3.org/p2p-webtransport/>.

[P2P-QUIC-TRIAL] Hampson, S., "RTCQuicTransport Coming to an Origin Trial Near You (Chrome 73)", January 2019, <https://developer.chrome.com/blog/rtcquictransport-api/>.

[RFC6189] Zimmermann, P., Johnston, A., Ed., and J. Callas, "ZRTP: Media Path Key Agreement for Unicast Secure RTP", RFC 6189, DOI 10.17487/RFC6189, April 2011, <https://www.rfc-editor.org/info/rfc6189>.

[RFC9369] Duke, M., "QUIC Version 2", RFC 9369, DOI 10.17487/RFC9369, May 2023, <https://www.rfc-editor.org/info/rfc9369>.

[RTP-OVER-QUIC] Ott, J., Engelbart, M., and S. Dawkins, "RTP over QUIC", Work in Progress, Internet-Draft, draft-ietf-avtcore-rtp-over-quic-04, 10 July 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-rtp-over-quic-04>.

Acknowledgments

We would like to thank Martin Thomson, Roni Even, Jonathan Lennox, and other participants in the IETF QUIC and AVTCORE Working Groups for their discussion of the QUIC multiplexing issue, and their input relating to potential solutions.

Authors' Addresses

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
United States of America
Email: bernard.aboba@gmail.com

Gonzalo Salgueiro
Cisco Systems
7200-12 Kit Creek Road
Research Triangle Park, NC 27709
United States of America
Email: gsalguei@cisco.com

Colin Perkins
School of Computing Science
University of Glasgow
Glasgow
G12 8QQ
United Kingdom
Email: csp@csperkins.org