QUIC
Internet Draft
Intended status: Standards Track
Expires: March 2023

J. Chen
T. Liu
J. Jing
Y. Yu
Bytedance Inc.

September 19, 2022

An Unreliable Extension to QUIC
draft-chen-quic-quicu-01.txt

Status of this Memo

Copyright Notice

Abstract

QUIC Unreliable (QUICU) is an extension of the QUIC protocol for unreliable data transmission, mainly through the definition and use of three new types of frames, namely the Expire Offset Frame, Boundary Frame, and Correlation Frame. The main purpose of this extension is to reduce data delivery delay. Through controlling the timing and scope of packet losses, QUICU reduces useless transmission to improve transmission efficiency, reduces head-of-line blocking to improve transmission timeliness, which are beneficial to delay-sensitive applications, especially audio and video applications.

This document describes the motivation, the operations, and the wire formats of the three new types of frames.

Table of Contents

1. Introduction

QUIC Unreliable (QUICU) is an extension of the QUIC protocol for unreliable data transmission, mainly through the definition and use of three new types of frames, namely the Expire Offset Frame, Boundary Frame, and Correlation Frame. The main purpose of this extension is to reduce data delivery delay. Through controlling the timing and scope of packet losses, QUICU reduces useless transmission to improve transmission efficiency, reduces head-of-line blocking to improve transmission timeliness, which are beneficial to delay-sensitive applications, especially audio and video applications.

This document describes the motivation, the operations, and the wire formats of the three new types of frames.

2. Conventions used in this document

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

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying significance described in RFC 2119.

3. Motivation

The QUIC protocol has a variety of features, such as 0RTT connection establishment, congestion control, link migration, secure and reliable transmission, and so on. Most UDP applications can be migrated to QUIC. RFC 9221 extends QUIC by adding support for sending and receiving unreliable datagrams over a QUIC connection.

In real-time transmission scenarios with high bit rates, such as audio and video transmission, including virtual reality type of applications, packet losses are sometimes unavoidable. On one hand, after detecting packet losses, QUIC's reliable transmission algorithm will continue to retransmit the lost data until the data is successfully received. Although this ensures the reliability of the data, it will increase the data delivery delay. On the other hand, there will also be issues when not applying any retransmission after packet losses, e.g., by using the new datagram QUIC frame types specified in RFC 9221. Although no retransmission is good for delay, the quality of data delivered to the application layer can become uncontrollable. For example, the audio can then have frequent and sharp bursts, and the video picture may be frequently blurred or even cannot be played. Therefore, it would be desirable to have a mechanism that allows to control the amount and time range of data that is retransmitted, thus allowing optimal tradeoff between delay and media quality.

This document proposes such a mechanism. The proposed mechanism involves the definition of three new types of frames, namely the Expire Offset Frame, Boundary Frame, and Correlation Frame. Each of these frames carries a type field indicating the frame type and the stream ID of the target stream for which certain information is provided by these frames. In addition, each of these frames carries a special field, summarized as follows:

1) The Expire Offset Frame (See Section 5.1.1) carries an offset of the target stream, for notifying the receiving end that data before the offset in the target stream will not be retransmitted. Upon reception of this notification, the receiving end shall not wait for the arrival of any data preceding the offset in the target stream.

2) The Boundary Frame (see Section 5.1.2) also carries an offset of the target stream, for indicating the end of an independent block of data in the target stream. All data pieces of an independent block of data should be coupled in transmission, in the sense that a data piece is meaningfully useful only when all of the preceding data pieces of the same independent block have been successfully received. This information can be used to save transmission, retransmission, or request for retransmission of certain data pieces when it is known that some preceding data pieces of the same independent block have been lost and will not be retransmitted. It can also be used to allocate the same priority for all data pieces of an independent block of data. For example, if a coded picture or a part thereof is lost and will not be retransmitted, then all subsequent pictures in the same independent group of pictures would typically be useless even received as they typically depend on the lost (or partially lost) preceding picture for inter prediction reference, therefore in this case transmission, retransmission, or request for retransmission of the data of these pictures can be skipped.

3) The Correlation Frame (see Section 5.1.3) carries the correlation stream ID, which is the stream ID of the stream that contains the request based on which the current stream was created, for associating multiple streams created based on the same request. For backward compatibility reasons, streaming systems may store multiplexed media contents, e.g., in the FLV (Flash Video) format, in the origin and edge servers, while transmitting the media components in separate streams using QUIC to take advantage of different priorities and other properties of the different media components. In such streaming systems, based on one request (either a download request or an upload request), multiple streams can be created. However, without additional information, the entity that receives these multiple streams won't be able to determine which of the received streams were originated from the same request and thus should be presented together. The Correlation Frames in these received streams provide such additional information; those received streams having the same correlation stream ID value were originated from the same request and should be presented together.

4. Operations

4.1. Transport Negotiation

Text When establishing a connection, it is necessary to confirm whether the receiver supports QUICU, using a QUIC transport parameter (name=support_quic_unreliable, value=0x21). The support_quic_unreliable transport parameter is an integer value (represented as a variable-length integer), for which the value 0 indicates that QUICU is not supported and the value 1 indicates that QUICU is supported. The default value for this parameter is 0.

When the connection is established, a bidirectional stream can be created like normal QUIC. Before receiving the Boundary Frame, the receiver can respond to the data like a normal QUIC stream, while after receiving the Boundary Frame, it starts to enter the QUICU mode.

When the peer's support_quic_unreliable transport parameter is not received during the handshake negotiation, the Boundary Frame, Correlation Frame and Expire Offset Frame cannot be sent to the peer. If one side receives a QUICU-specific frame from the peer while it itself does not carry the support_quic_unreliable transport parameter, it should disconnect the connection and return the error code "QUICU mode not supported".

Once the QUICU mode is established for a stream, the QUICU mode is used for the entire lifetime of the stream.

The QUICU mode can be unidirectional. Receiving a Boundary Frame indicates that the incoming flow for an entity is in the QUICU mode. However, at the same time, the outgoing flow of this entity can still be in the normal QUIC mode.

4.2. Data Sending

When different components of the data do not need to be treated differently in transmission, e.g., with different priorities, they can be transmitted in the same stream, and this transmission mode is referred to as the single-stream mode. In this case, the data is encapsulated in Stream Frames as usual.

When different components of the data do need to be treated differently in transmission, e.g., with different priorities, they should be transmitted in different streams, and this transmission mode is referred to as the multi-stream mode. In this case, Correlation Frames need to be sent to associate streams created based on the same request.

In both the single-stream and multi-stream modes, and Boundary Frames can be used to indicate the boundaries between independent blocks of data in a stream.

4.3. Data Expiration

A piece of sent data of a stream can be considered lost if its reception is not confirmed for a certain period of time; this period of time is referred to as the timeout period. After the timeout period expires, an Expire Offset Frame can be sent to notify the other side that certain data of the stream has been lost and will not be retransmitted. The Expire Offset Frame needs to be sent out as soon as possible after the timeout period expires. Note that this operation is RFC 9221 compliant in the sense that the data transmission is not guaranteed to be reliable.

The timeout period can be a fixed value; or configured by users through the application.

4.4. Data Receiving

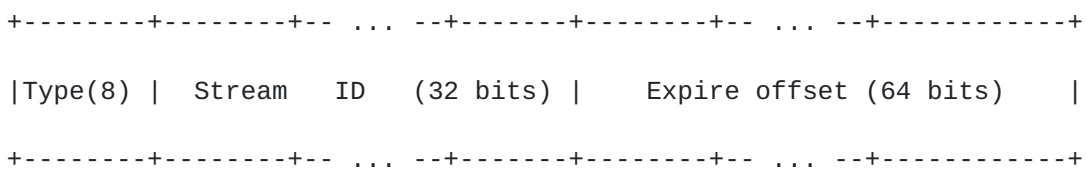
In a stream, if a Boundary Frame is not received, the data of the stream will be delivered to the application same as in the normal QUIC mode; otherwise, the QUICU mode will be in use.

In the QUICU mode, the data in independent block is only delivered to the application when the entire independent block of data has been received and placed in the right order, after the delivery of the previous independent block of data of the same stream. However, when some data of an independent block expires as indicated by an Expire Offset Frame, the content of this independent block is discarded, and the next independent block of data can be delivered to the application when the entire independent block has been received and placed in the right order.

5. The Wire Format

Based on the basic standard of QUIC, RFC 9000, three new types of frames are specified, namely, the Boundary Frame, Correlation Frame, and Expire Offset Frame.

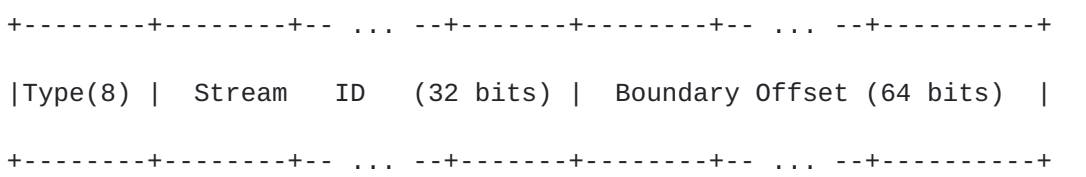
5.1. Expire Offset Frame



Stream ID: Indicates the stream ID of the target stream for which information on which pieces of data will not to be retransmitted is indicated by the Expire Offset field.

Expire Offset: Indicating the offset, in units of bytes, of such a position of the target stream, that no data before that position in the target stream will be retransmitted.

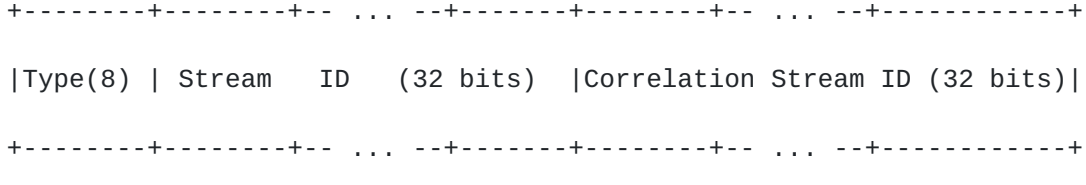
5.2. Boundary Frame



Stream ID: Indicates the stream ID of the target stream for which the end position of an independent block of data is indicated by the Boundary Offset field.

Boundary Offset: Indicates the end position of an independent block of data in the target stream. Let the value of the Boundary Offset field of the i-th Boundary Frame be bo[i] with bo[0] being the value of the very first Boundary Frame. The first independent block of data in the target stream consists of bytes 0 to bo[0], inclusive. For any value of i greater than 0, the i-th independent block of data in the target stream consists of bytes bo[i-1] + 1 to bo[i], inclusive. The Boundary Frame of the last independent block of data in the target stream can be but does not have to be sent, as it is known that the end position is the end of the target stream itself.

5.3. Correlation Frame



Stream ID: Indicates the stream ID of the target stream that is one of the streams created based on a request contained in a stream identified by the Correlation Stream ID field.

Correlation Stream ID: Indicates the stream ID of the stream containing the request based on which the target stream was created.

6. Security Considerations

The three new types of frames share the same security properties as the rest of the data transmitted within a QUIC connection, and the security considerations of [RFC9000] apply accordingly. All new frame types, like the Expire Offset Frame, the Boundary Frame and the Boundary Frame, MUST be protected either by 0-RTT or 1-RTT keys.

7. IANA Considerations

This document registers a new value in the QUIC Transport Parameter Registry:

Value:0x0021 (if this document is approved)

Parameter Name: support_quic_unreliable

Specification: Section 4.1 of this document.

This document also registers three new values in the QUIC Frame Type Registry:

Value:0x32 (if this document is approved)

Frame Name: Expire Offset

Specification: Section 5.1.1 of this document

Value:0x33 (if this document is approved)

Frame Name: Boundary

Specification: Section 5.1.2 of this document

Value:0x34 (if this document is approved)

Frame Name: Correlation

Specification: Section 5.1.3 of this document

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[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/rfc/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/rfc/rfc9000>.

[RFC9221] Pauly, T., Kinnear, E., and D. Schinazi, "An Unreliable Datagram Extension to QUIC", RFC 9221, DOI 10.17487/RFC9221, March 2022, <https://www.rfc-editor.org/info/rfc9221>.

Authors' Addresses

Jianping Chen
Bytedance Inc.
No. 27, North Third Ring West Road, Haidian District,
Beijing
China
Email: chenjianping.ireader@bytedance.com

Tianyi Liu
Bytedance Inc.
No. 27, North Third Ring West Road, Haidian District,
Beijing
China
Email: liutianyi.lty@bytedance.com

Junxian Jing
Bytedance Inc.
No. 27, North Third Ring West Road, Haidian District,
Beijing
China
Email: jingjunxian@bytedance.com

Yongyi Yu
Bytedance Inc.
No. 27, North Third Ring West Road, Haidian District,
Beijing
China
Email: yuyongyi.yyy@bytedance.com