Workgroup: QUIC
Internet-Draft: draft-banks-quic-cibir-00
Published: 1 March 2022
Intended Status: Experimental
Expires: 2 September 2022
Authors: N. Banks
Microsoft Corporation

QUIC Connection ID Based Initial Routing

Abstract

This document defines an extension to the QUIC transport protocol to consistently route all packets from a client to the appropriate server on a shared UDP port.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Several scenarios exist where multiple independent or isolated servers need to run in the same environment, but cannot use independent local UDP ports. For instance, in server deployments that have hundreds or thousands of machines, each with tens or hundreds of different QUIC servers running on them, the server infrastructure may not be able to support the number of local UDP ports it would require to give each server a unique one. Additionally, because of infrastructure requirements additional IP addresses may not be able to be used as a solution either.

In these scenarios, the server infrastructure needs a way to essentially NAT QUIC packets on a shared local UDP port between all servers using that port. This document defines a mechanism for using QUIC connection IDs to encode the necessary information for all client to server QUIC packets to be correctly routed to the appropriate server. A cooperating client can then use this to specifically target a server on a shared port.

1.1. Terms and Definitions

The keywords "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. Specification

2.1. Transport Parameter

Support for encoding CIBIR information is negotiated by means of a QUIC Transport Parameter (name=cibir_encoding, value=0x30). The cibir_encoding transport parameter consists of two integer values (represented as variable-length integers) that represent the length and offset to the well-known identifier encoded into the client's source connection ID.

Servers that share a local UDP port using the CIBIR extension unconditionally route received packets according to the CIBIR extension's protocol. The cibir_encoding transport parameter is used on the server side after the routing has already happened to validate the intent of the client. Servers MUST validate the client sent the cibir_encoding transport parameter with the matching offset and length that has been configured locally. If the transport parameter is missing or contains incorrect values the server MUST terminate the connection with an error of type CONNECTION_REFUSED.

No special routing is done on the client side, but client MUST also validate the server sent the cibir_encoding transport parameter with the matching offset and length so as to verify the server is cooperating in the expected routing scheme. If the transport parameter is missing or contains incorrect values the client MUST terminate the connection with an error of type TRANSPORT_PARAMETER_ERROR.

2.2. Packet Encoding and Routing

The base QUIC transport protocol provides no way to consistently route long header packets to the correct server in a shared UDP environment. The only possibly way a server's infrastructure has to identify which server the client is trying to connect to is the ALPN or SNI, but these are not included in all long header packets. Additionally, the destination connection ID in packets sent to the server cannot be used because there is no stateless way determine if the CID is client or server chosen, not to mention the complexities around server chosen CIDs in a load balanced environment (which the client does not necessarily know anything about).

To achieve consistent routing for these long header packets, the client encodes a well-known identifier into its source connection ID. The length and offset of the well-known ID must be pre-agreed upon between the client and server, and is validated via the cibir_encoding transport parameter as described above. When the server infrastructure receives a QUIC long header packet on the shared UDP port it uses the well-known identifier to route the packet to the correct server.

No special routing is necessary for short header packets. These packets always use server chosen destination connection IDs, and the logic by which these CIDs are chosen, created and interpreted is purely up to the server and server infrastructure. The client doesn't need to be involved in this logic beyond the normal use of destination connection IDs.

3. Security Considerations

The client encodes well-known IDs in the QUIC connection ID that may expose information to an observer.

4. IANA Considerations

4.1. QUIC Transport Parameter

This document registers a new value in the QUIC Transport Parameter Registry maintained at https://www.iana.org/assignments/quic/quic.xhtml#quic-transport.

Value: 0x30

Parameter Name: cibir_encoding

Status: permanent

Specification: This document

5. 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>.

[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>.

Author's Address

Nick Banks
Microsoft Corporation

Email: nibanks@microsoft.com