Workgroup: MASQUE
Internet-Draft:
draft-schinazi-masque-connect-udp-ecn-02
Published: 28 March 2022
Intended Status: Standards Track
Expires: 29 September 2022
Authors: D. Schinazi
Google LLC

An ECN Extension to CONNECT-UDP

Abstract

CONNECT-UDP allows proxying UDP packets over HTTP. This document describes an extension to CONNECT-UDP that allows conveying ECN information on proxied UDP packets.

Discussion Venues

This note is to be removed before publishing as an RFC.

Source for this draft and an issue tracker can be found at https://github.com/DavidSchinazi/draft-connect-udp-ecn.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction
  1.1. Conventions and Definitions
2. Context Identifiers
3. ECN Header Definition
4. Encoding of ECN bits
5. A Note about Future Extensions
6. Security Considerations
7. IANA Considerations
8. Normative References
Acknowledgments
Author's Address

1. Introduction

CONNECT-UDP [CONNECT-UDP] allows proxying UDP packets over HTTP. This document describes an extension to CONNECT-UDP that allows conveying ECN [ECN] information on proxied UDP packets.

1.1. Conventions 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.

2. Context Identifiers

The "Context Identifiers" section of [CONNECT-UDP] defines the concept of context IDs and how they can be used to extend CONNECT-UDP. When a client wishes to use ECN with CONNECT-UDP, it generates an unique client-allocated context ID. In this document, we'll refer to that context ID as the "chosen context ID". Note that, by definition of client-allocated context IDs, the chosen context ID will always be a non-zero even number. We also add the restriction that the chosen context ID **MUST** be strictly less than 10^{15}. If the client has run out of available context ID values that match this requirement for this CONNECT-UDP request, it **MUST NOT** use the ECN extension with this CONNECT-UDP request.

3. ECN Header Definition

The "ECN" header field is an Item Structured Field, see Section 3.3 of [STRUCT-FIELD]; its value **MUST** be a Integer; any other value type **MUST** be handled as if the field were not present by recipients (for example, if this field is included multiple times, its type will become a List and the field will therefore be ignored). This document does not define any parameters for the "ECN" header field value, but future documents might define parameters. Receivers **MUST** ignore unknown parameters.

When present, the "ECN" header indicates that the sender supports this extension, and communicates the chosen context ID as the "ECN" field value.

For example, if the client chosen context ID is 42, it would send the following:

```
ECN: 42; foo=bar
```

Figure 1: Example Client ECN Field

Clients **MUST NOT** indicate support for this extension unless they know that the protocol running over UDP that is being proxied supports ECN, and will react appropriately to Congestion Experienced (CE) markings.

Proxies **MUST NOT** indicate support for this extension unless they know they have the ability to read and write the IP ECN bits on its target-bound UDP sockets.

This extension is said to have been negotiated when both client and proxy indicated support for it in their CONNECT-UDP request and response. When indicating support for this extension, the proxy send the client's chosen context ID as the "ECN" field value.

For example, the proxy could reply with:

```
ECN: 42
```

Figure 2: Example Proxy ECN Field

4. Encoding of ECN bits

When an HTTP Datagram [HTTP-DGRAM] associated with a CONNECT-UDP stream uses the chosen context ID as its context ID, its "Payload" field contains the following format (using the notation from the "Notational Conventions" section of [QUIC]):

```
CONNECT-UDP Payload with ECN {
  Must be Zero (6),
  ECN Bits (2),
  UDP Payload (..),
}
```

Figure 3: CONNECT-UDP Payload with ECN

Must be Zero: 6 bits that **MUST** be sent as zero. Receivers **MUST** validate that these bits are zero and **MUST** silently drop the HTTP Datagram if they have any other value. Extensions to this mechanism **MAY** relax this requirement.

ECN Bits: The ECN bits, sent in the same order as they appear in the IP header.

UDP Payload: The UDP Payload, as defined in the "HTTP Datagram Payload Format" section of [CONNECT-UDP].

When the proxy receives a datagram with the chosen context ID, it sets the IP packet's ECN bits accordingly on the UDP packet it sends to the target. Similarly, in the other direction the ECN Bits field represents which ECN bits were seen on the UDP packets received from the target.

5. A Note about Future Extensions

This CONNECT-UDP extension uses an HTTP field to register its chosen context ID. Future extensions to CONNECT-UDP can use the same strategy to register their chosen context ID(s) via another HTTP field. This strategy is best for CONNECT-UDP extensions that only need to register context IDs during the HTTP request and response.

Some extensions may need to register context IDs after the request and response have been exchanged, for example an extension that wishes to compress QUIC connection IDs [QUIC] is not aware of all connection IDs at request time. In such cases, extensions can use new Capsule Types (see [HTTP-DGRAM]) to perform context ID registration.

6. Security Considerations

This document does not have additional security considerations beyond those defined in [CONNECT-UDP].

7. IANA Considerations

This document will request IANA to register the following entry in the "HTTP Field Name" registry:

Field Name:
ECN

Template: None

Status: provisional (permanent if this document is approved)

Reference: This document

Comments: None

8. Normative References

[CONNECT-UDP] Schinazi, D., "UDP Proxying Support for HTTP", Work in Progress, Internet-Draft, draft-ietf-masque-connect-udp-08, 21 March 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-masque-connect-udp-08>.

[ECN] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/rfc/rfc3168>.

[HTTP-DGRAM] Schinazi, D. and L. Pardue, "HTTP Datagrams and the Capsule Protocol", Work in Progress, Internet-Draft, draft-ietf-masque-h3-datagram-08, 28 March 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-masque-h3-datagram-08>.

[QUIC] 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>.

[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/rfc/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/rfc/rfc8174>.

[STRUCT-FIELD] Nottingham, M. and P-H. Kamp, "Structured Field Values for HTTP", RFC 8941, DOI 10.17487/RFC8941, February 2021, <https://www.rfc-editor.org/rfc/rfc8941>.

Acknowledgments

This proposal was inspired directly or indirectly by prior work from many people. The author would like to thank contributors the MASQUE working group.

Author's Address

David Schinazi
Google LLC
1600 Amphitheatre Parkway
Mountain View, California 94043,
United States of America

Email: dschinazi.ietf@gmail.com