Network Working Group                                    H. Alvestrand
Internet-Draft                                                    Google
Intended status: Standards Track                        January 22, 2014
Expires: July 26, 2014

# Transports for RTCWEB
# draft-ietf-rtcweb-transports-02

## Abstract

This document describes the data transport protocols used by RTCWEB, including the protocols used for interaction with intermediate boxes such as firewalls, relays and NAT boxes.

## Status of this Memo



## Copyright Notice

Table of Contents

# 1. Introduction

The IETF RTCWEB effort, part of the WebRTC effort carried out in cooperation between the IETF and the W3C, is aimed at specifying a protocol suite that is useful for real time multimedia exchange between browsers.

The overall effort is described in the RTCWEB overview document, [I-D.ietf-rtcweb-overview]. This document focuses on the data transport protocos that are used by conforming implementations.

This protocol suite is designed for WebRTC, and intends to satisfy the security considerations described in the WebRTC security documents, [I-D.ietf-rtcweb-security] and [I-D.ietf-rtcweb-security-arch].

# 2. Requirements language

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].

# 3. Transport and Middlebox specification

## 3.1. System-provided interfaces

The protocol specifications used here assume that the following protocols are available to the implementations of the RTCWEB protocols:

- o UDP. This is the protocol assumed by most protocol elements described.

- o TCP. This is used for HTTP/WebSockets, as well as for TURN/SSL and ICE-TCP.

For both protocols, IPv4 and IPv6 support is assumed; applications MUST be able to utilize both IPv4 and IPv6 where available.

For UDP, this specification assumes the ability to set the DSCP code point of the sockets opened on a per-packet basis, in order to achieve the prioritizations described in [I-D.dhesikan-tsvwg-rtcweb-qos] when multiple media types are multiplexed. It does not assume that the DSCP codepoints will be honored, and does assume that they may be zeroed or changed, since this is a local configuration issue.

This specification does not assume that the implementation will have access to ICMP or raw IP.

### 3.2. Usage of Quality of Service functions

WebRTC implementations SHOULD attempt to set QoS on the packets sent, according to the guidelines in [I-D.dhesikan-tsvwg-rtcweb-qos]. It is appropriate to depart from this recommendation when running on platforms where QoS marking is not implemented.

### 3.3. Support for multiplexing

RTCWEB implementations MUST support the ability to send and receive multiple SSRCs on the same transport, and MUST support the ability to send and receive multiple SSRCs on multiple simultaneous transports, including the ability to send and receive audio and video on the same transport. The choice of configuration is done at higher layers (above transport), using mechanisms like BUNDLE [I-D.ietf-mmusic-sdp-bundle-negotiation]. Further information on RTP usage is found in [I-D.ietf-rtcweb-rtp-usage].

When different content types according to [I-D.dhesikan-tsvwg-rtcweb-qos] are used on the same transport, appropriate per-packet DSCP marking SHOULD be used.

DISCUSSION: Minimizing the number of transports has advantages in traversing NATs and firewalls, due to the reduced chance of negotiation failure. However, some network prioritization mechanisms (in particular active queue management techniques and flow-recognizing deep packet inspection boxes) will perform better when flows with different characteristics are separated on different 5-tuples. Since the optimum for this tradeoff is unknown, and may be variable, it is inappropriate to embed this choice in the protocol layer, and this is therefore left to the control of the application.

### 3.4. Middle box related functions

The primary mechanism to deal with middle boxes is ICE, which is an appropriate way to deal with NAT boxes and firewalls that accept traffic from the inside, but only from the outside if it's in response to inside traffic (simple stateful firewalls).

ICE [RFC5245] MUST be supported. The implementation MUST be a full ICE implementation, not ICE-Lite.

In order to deal with situations where both parties are behind NATs which perform endpoint-dependent mapping (as defined in [RFC5128] section 2.4), TURN [RFC5766] MUST be supported.

In order to deal with firewalls that block all UDP traffic, TURN using TCP between the client and the server MUST be supported, and TURN using TLS between the client and the server MUST be supported. See [RFC5766] section 2.1 for details.

In order to deal with situations where one party is on an IPv4 network and the other party is on an IPv6 network, TURN extensions for IPv6 [RFC6156] MUST be supported.

TURN TCP candidates [RFC6062] SHOULD be supported; this allows applications to achieve peer-to-peer communication when both parties are behind UDP-blocking firewalls using a single TURN server. (In this case, one can also achieve communication using two TURN servers that use TCP between the server and the client, and UDP between the TURN servers.)

ICE-TCP candidates [RFC6544] MAY be supported; this may allow applications to communicate to peers with public IP addresses across UDP-blocking firewalls without using a TURN server.

The ALTERNATE-SERVER mechanism specified in [RFC5389] (STUN) section 11 (300 Try Alternate) MUST be supported.

Further discussion of the interaction of RTCWEB with firewalls is contained in [I-D.hutton-rtcweb-nat-firewall-considerations]. This document makes no requirements on interacting with HTTP proxies or HTTP proxy configuration methods.

### 3.5. Transport protocols implemented

For transport of media, secure RTP is used. The details of the profile of RTP used are described in "RTP Usage" [I-D.ietf-rtcweb-rtp-usage].

For data transport over the RTCWEB data channel [I-D.ietf-rtcweb-data-channel], RTCWEB implementations MUST support SCTP over DTLS over ICE. This encapsulation is specified in [I-D.ietf-tsvwg-sctp-dtls-encaps]. Negotiation of this transport in SDP is defined in [I-D.ietf-mmusic-sctp-sdp].

The setup protocol for RTCWEB data channels is described in [I-D.jesup-rtcweb-data-protocol].

RTCWEB implementations MUST support multiplexing of DTLS and RTP over the same port pair, as described in the DTLS_SRTP specification [RFC5764], section 5.1.2. All application layer protocol payloads over this DTLS connection are SCTP packets.

## 4. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

## 5. Security Considerations

Security considerations are enumerated in [I-D.ietf-rtcweb-security].

## 6. Acknowledgements

This document is based on earlier versions embedded in [I-D.ietf-rtcweb-overview], which were the results of contributions from many RTCWEB WG members.

Special thanks for reviews of earlier versions of this draft go to Magnus Westerlund, Markus Isomaki and Dan Wing; the contributions from Andrew Hutton also deserve special mention.

## 7. References

### 7.1. Normative References

[I-D.dhesikan-tsvwg-rtcweb-qos]
Dhesikan, S., Druta, D., Jones, P., and J. Polk, "DSCP and other packet markings for RTCWeb QoS", draft-dhesikan-tsvwg-rtcweb-qos-03 (work in progress), December 2013.

[I-D.ietf-mmusic-sctp-sdp]
Loreto, S. and G. Camarillo, "Stream Control Transmission Protocol (SCTP)-Based Media Transport in the Session Description Protocol (SDP)", draft-ietf-mmusic-sctp-sdp-05 (work in progress), October 2013.

[I-D.ietf-rtcweb-data-channel]
Jesup, R., Loreto, S., and M. Tuexen, "RTCWeb Data Channels", draft-ietf-rtcweb-data-channel-06 (work in progress), October 2013.

[I-D.ietf-rtcweb-rtp-usage]
Perkins, C., Westerlund, M., and J. Ott, "Web Real-Time Communication (WebRTC): Media Transport and Use of RTP",

draft-ietf-rtcweb-rtp-usage-11 (work in progress),
December 2013.

[I-D.ietf-rtcweb-security]
Rescorla, E., "Security Considerations for WebRTC",
draft-ietf-rtcweb-security-05 (work in progress),
July 2013.

[I-D.ietf-rtcweb-security-arch]
Rescorla, E., "WebRTC Security Architecture",
draft-ietf-rtcweb-security-arch-07 (work in progress),
July 2013.

[I-D.ietf-tsvwg-sctp-dtls-encaps]
Tuexen, M., Stewart, R., Jesup, R., and S. Loreto, "DTLS
Encapsulation of SCTP Packets",
draft-ietf-tsvwg-sctp-dtls-encaps-02 (work in progress),
October 2013.

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

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment
(ICE): A Protocol for Network Address Translator (NAT)
Traversal for Offer/Answer Protocols", RFC 5245,
April 2010.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing,
"Session Traversal Utilities for NAT (STUN)", RFC 5389,
October 2008.

[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, May 2010.

[RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using
Relays around NAT (TURN): Relay Extensions to Session
Traversal Utilities for NAT (STUN)", RFC 5766, April 2010.

[RFC6062] Perreault, S. and J. Rosenberg, "Traversal Using Relays
around NAT (TURN) Extensions for TCP Allocations",
RFC 6062, November 2010.

[RFC6156] Camarillo, G., Novo, O., and S. Perreault, "Traversal
Using Relays around NAT (TURN) Extension for IPv6",
RFC 6156, April 2011.

[RFC6544] Rosenberg, J., Keranen, A., Lowekamp, B., and A. Roach,

"TCP Candidates with Interactive Connectivity Establishment (ICE)", RFC 6544, March 2012.

### 7.2. Informative References

[I-D.hutton-rtcweb-nat-firewall-considerations]
Stach, T., Hutton, A., and J. Uberti, "RTCWEB Considerations for NATs, Firewalls and HTTP proxies", draft-hutton-rtcweb-nat-firewall-considerations-02 (work in progress), September 2013.

[I-D.ietf-mmusic-sdp-bundle-negotiation]
Holmberg, C., Alvestrand, H., and C. Jennings, "Multiplexing Negotiation Using Session Description Protocol (SDP) Port Numbers", draft-ietf-mmusic-sdp-bundle-negotiation-05 (work in progress), October 2013.

[I-D.ietf-rtcweb-overview]
Alvestrand, H., "Overview: Real Time Protocols for Brower-based Applications", draft-ietf-rtcweb-overview-08 (work in progress), September 2013.

[I-D.jesup-rtcweb-data-protocol]
Jesup, R., Loreto, S., and M. Tuexen, "WebRTC Data Channel Protocol", draft-jesup-rtcweb-data-protocol-04 (work in progress), February 2013.

[RFC5128] Srisuresh, P., Ford, B., and D. Kegel, "State of Peer-to-Peer (P2P) Communication across Network Address Translators (NATs)", RFC 5128, March 2008.

## Appendix A. Change log

### A.1. Changes from -00 to -01

- Clarified DSCP requirements, with reference to -qos-
- Clarified "symmetric NAT" -> "NATs which perform endpoint-dependent mapping"
- Made support of TURN over TCP mandatory
- Made support of TURN over TLS a MAY, and added open question
- Added an informative reference to -firewalls-

- Called out that we don't make requirements on HTTP proxy interaction (yet

### A.2. Changes from -01 to -02

- Required support for 300 Alternate Server from STUN.

- Separated the ICE-TCP candidate requirement from the TURN-TCP requirement.

- Added new sections on using QoS functions, and on multiplexing considerations.

- Removed all mention of RTP profiles. Those are the business of the RTP usage draft, not this one.

- Required support for TURN IPv6 extensions.

- Removed reference to the TURN URI scheme, as it was unnecessary.

- Made an explicit statement that multiplexing (or not) is an application matter.

.

## Author's Address

Harald Alvestrand
Google

Email: harald@alvestrand.no