RTCWEB
Internet-Draft
Intended status: Standards Track
Expires: November 6, 2016

M. Thomson
Mozilla

May 5, 2016

Application Layer Protocol Negotiation for Web Real-Time Communications (WebRTC)

draft-ietf-rtcweb-alpn-04

Abstract

This document specifies two Application Layer Protocol Negotiation (ALPN) labels for use with Web Real-Time Communications (WebRTC). The "webrtc" label identifies regular WebRTC communications: a DTLS session that is used establish keys for Secure Real-time Transport Protocol (SRTP) or to establish data channels using SCTP over DTLS. The "c-webrtc" label describes the same protocol, but the peers also agree to maintain the confidentiality of the media by not sharing it with other applications.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Web Real-Time Communications (WebRTC) [I-D.ietf-rtcweb-overview] uses Datagram Transport Layer Security (DTLS) [RFC6347] to secure all peer-to-peer communications.

Identifying WebRTC protocol usage with Application Layer Protocol Negotiation (ALPN) [RFC7301] enables an endpoint to positively identify WebRTC uses and distinguish them from other DTLS uses.

Different WebRTC uses can be advertised and behavior can be constrained to what is appropriate to a given use. In particular, this allows for the identification of sessions that require confidentiality protection from the application that manages the signaling for the session.

1.1. Conventions and 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 [RFC2119].

2. ALPN Labels for WebRTC

The following identifiers are defined for use in ALPN:

webrtc: The DTLS session is used to establish keys for Secure Real-time Transport Protocol (SRTP) - known as DTLS-SRTP - as described

in [RFC5764]. The DTLS record layer is used for WebRTC data channels [I-D.ietf-rtcweb-data-channel].

c-webrtc: The DTLS session is used for confidential WebRTC communications, where peers agree to maintain the confidentiality of the media, as described in Section 3. The confidentiality protections ensure that media is protected from other applications, but the confidentiality protections do not extend to messages on data channels.

Both identifiers describe the same basic protocol: a DTLS session that is used to provide keys for an SRTP session in combination with WebRTC data channels. Either SRTP or data channels could be absent. The data channels send Stream Control Transmission Protocol (SCTP) [RFC4960] over the DTLS record layer, which can be multiplexed with SRTP on the same UDP flow. WebRTC requires the use of Interactive Communication Establishment (ICE) [RFC5245] to establish the UDP flow, but this is not covered by the identifier.

A more thorough definition of what WebRTC communications entail is included in [I-D.ietf-rtcweb-transports].

There is no functional difference between the identifiers except that an endpoint negotiating "c-webrtc" makes a promise to preserve the confidentiality of the media it receives.

A peer that is not aware of whether it needs to request confidentiality can use either identifier. A peer in the client role MUST offer both identifiers if it is not aware of a need for confidentiality. A peer in the server role SHOULD select "webrtc" if it does not prefer either.

An endpoint that requires media confidentiality might negotiate a session with a peer that does not support this specification. Endpoint MUST abort a session if it requires confidentiality but does not successfully negotiate "c-webrtc". A peer that is willing to accept "webrtc" SHOULD assume that a peer that does not support this specification has negotiated "webrtc" unless signaling provides other information; however, a peer MUST NOT assume that "c-webrtc" has been negotiated unless explicitly negotiated.

3. Media Confidentiality

Private communications in WebRTC depend on separating control (i.e., signaling) capabilities and access to media [I-D.ietf-rtcweb-security-arch]. In this way, an application can establish a session that is end-to-end confidential, where the ends in question are user agents (or browsers) and not the signaling

application. This allows an application to manage signaling for a session, without having access to the media that is exchanged in the session.

Without some form of indication that is securely bound to the session, a WebRTC endpoint is unable to properly distinguish between a session that requires this confidentiality protection and one that does not. The ALPN identifier provides that signal.

A browser is required to enforce this confidentiality protection using isolation controls similar to those used in content cross-origin protections (see Section 5.3 [1] of [HTML5]). These protections ensure that media is protected from applications. Applications are not able to read or modify the contents of a protected flow of media. Media that is produced from a session using the "c-webrtc" identifier MUST only be displayed to users.

The promise to apply confidentiality protections do not apply to data that is sent using data channels. Confidential data depends on having both data sources and consumers that are exclusively browser- or user-based. No mechanisms currently exist to take advantage of data confidentiality, though some use cases suggest that this could be useful, for example, confidential peer-to-peer file transfer. Alternative labels might be provided in future to support these use cases.

This mechanism explicitly does not define a specific authentication method; a WebRTC endpoint that accepts a session with this ALPN identifier MUST respect confidentiality no matter what identity is attributed to a peer.

RTP middleboxes and entities that forward media or data cannot promise to maintain confidentiality. Any entity that forwards content, or records content for later access by entities other than the authenticated peer, MUST NOT offer or accept a session with the "c-webrtc" identifier.

4. Security Considerations

Confidential communications depends on more than just an agreement from browsers.

Information is not confidential if it is displayed to those other than to whom it is intended. Peer authentication [I-D.ietf-rtcweb-security-arch] is necessary to ensure that data is only sent to the intended peer.

This is not a digital rights management mechanism. A user is not prevented from using other mechanisms to record or forward media. This means that (for example) screen recording devices, tape recorders, portable cameras, or a cunning arrangement of mirrors could variously be used to record or redistribute media once delivered. Similarly, if media is visible or audible (or otherwise accessible) to others in the vicinity, there are no technical measures that protect the confidentiality of that media.

The only guarantee provided by this mechanism and the browser that implements it is that the media was delivered to the user that was authenticated. Individual users will still need to make a judgment about how their peer intends to respect the confidentiality of any information provided.

On a shared computing platform like a browser, other entities with access to that platform (i.e., web applications), might be able to access information that would compromise the confidentiality of communications. Implementations MAY choose to limit concurrent access to input devices during confidential communications sessions.

For instance, another application that is able to access a microphone might be able to sample confidential audio that is playing through speakers. This is true even if acoustic echo cancellation, which attempts to prevent this from happening, is used. Similarly, an application with access to a video camera might be able to use reflections to obtain all or part of a confidential video stream.

5. IANA Considerations

The following two entries are added to the "Application Layer Protocol Negotiation (ALPN) Protocol IDs" registry established by [RFC7301]:

webrtc:

The "webrtc" label identifies mixed media and data communications using SRTP and data channels:

Protocol: WebRTC Media and Data

Identification Sequence: 0x77 0x65 0x62 0x72 0x74 0x63 ("webrtc")

Specification: This document (RFCXXXX)

c-webrtc:

The "c-webrtc" label identifies WebRTC communications with a promise to protect media confidentiality:

Protocol: Confidential WebRTC Media and Data

Identification Sequence: 0x63 0x2d 0x77 0x65 0x62 0x72 0x74 0x63 ("c-webrtc")

Specification: This document (RFCXXXX)

6. References

6.1. Normative References

[I-D.ietf-rtcweb-data-channel]

Jesup, R., Loreto, S., and M. Tuexen, "WebRTC Data Channels", draft-ietf-rtcweb-data-channel-13 (work in progress), January 2015.

[I-D.ietf-rtcweb-security-arch]
Rescorla, E., "WebRTC Security Architecture", draft-ietf-rtcweb-security-arch-11 (work in progress), March 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[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, <http://www.rfc-editor.org/info/rfc5764>.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <http://www.rfc-editor.org/info/rfc6347>.

[RFC7301] Friedl, S., Popov, A., Langley, A., and E. Stephan, "Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension", RFC 7301, DOI 10.17487/RFC7301, July 2014, <http://www.rfc-editor.org/info/rfc7301>.

6.2. Informative References

[HTML5] Berjon, R., Leithead, T., Doyle Navara, E., O'Connor, E., and S. Pfeiffer, "HTML 5", CR CR-html5-20121217, August 2010, <http://www.w3.org/TR/2012/CR-html5-20121217/>.

[I-D.ietf-rtcweb-overview]
Alvestrand, H., "Overview: Real Time Protocols for Browser-based Applications", draft-ietf-rtcweb-overview-15 (work in progress), January 2016.

[I-D.ietf-rtcweb-transports]
Alvestrand, H., "Transports for WebRTC", draft-ietf-rtcweb-transports-12 (work in progress), March 2016.

[RFC4960] Stewart, R., Ed., "Stream Control Transmission Protocol", RFC 4960, DOI 10.17487/RFC4960, September 2007, <http://www.rfc-editor.org/info/rfc4960>.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, DOI 10.17487/RFC5245, April 2010, <http://www.rfc-editor.org/info/rfc5245>.

6.3. URIs

[1] http://www.w3.org/TR/2012/CR-html5-20121217/browsers.html#origin

Author's Address

Martin Thomson
Mozilla
331 E Evelyn Street
Mountain View, CA 94041
US

Email: martin.thomson@gmail.com