Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 22, 2016

P. Jones
Cisco Systems

March 21, 2016

DTLS Tunnel between Media Distribution Device and Key Management Function to Facilitate Key Exchange
draft-jones-perc-dtls-tunnel-02

Abstract

This document defines a DTLS tunneling protocol for use in multimedia conferences that enables a Media Distribution Device (MDD) to facilitate key exchange between an endpoint in a conference and the Key Management Function (KMF) responsible for key distribution. The protocol is designed to ensure that the keying material used for hop-by-hop encryption and authentication is accessible to the MDD, while the keying material used for end-to-end encryption and authentication is inaccessible to the MDD.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 22, 2016.

Table of Contents

1. Introduction

An objective of the work in the Privacy-Enhanced RTP Conferencing (PERC) working group is to ensure that endpoints in a multimedia conference have access to the end-to-end (E2E) and hop-by-hop (HBH) keying material used to encrypt and authenticate Real-time Transport Protocol (RTP) [RFC3550] packets, while the Media Distribution Device (MDD) has access only to the hop-by-hop (HBH) keying material for encryption and authentication.

This specification defines a tunneling protocol that enables the MDD to tunnel DTLS [RFC6347] messages between an endpoint and the KMF, thus allowing an endpoint to use DTLS-SRTP [RFC5764] for establishing encryption and authentication keys with the KMF.

The tunnel established between the MDD and KMF is a DTLS association that is established before any messages are forwarded on behalf of the endpoint by the MDD. DTLS packets received from the endpoint are encapsulated by the MDD inside this tunnel as data to be sent to the KMF. Likewise, when the MDD receives data from the KMF over the tunnel, it extracts the DTLS message inside and forwards that to the endpoint. In this way, the DTLS association for the DTLS-SRTP

procedures is established between the endpoint and the KMF, with the MDD simply forwarding packets between the two entities and having no visibility into the confidential information exchanged or derived.

Following the existing DTLS-SRTP procedures, the endpoint and KMF will arrive at a selected cipher and keying material, which are used for HBH encryption and authentication by both the endpoint and the MDD. However, since the MDD would not have direct access to this information, the KMF explicitly shares the HBH key information with the MDD via the tunneling protocol defined in this document.

By establishing this DTLS tunnel between the MDD and KMF and implementing the protocol defined in this document, it is possible for the MDD to facilitate the establishment of a secure DTLS association between an endpoint and the KMF in order for the endpoint to receive E2E and HBH keying material. At the same time, the KMF can securely provide the HBH keying material to the MDD.

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 [RFC2119] when they appear in ALL CAPS. These words may also appear in this document in lower case as plain English words, absent their normative meanings.

3. Tunneling Concept

A DTLS association (tunnel) is established between the MDD and the KMF. This tunnel is used to relay DTLS messages between the endpoint and KMF, as depicted in Figure 1:

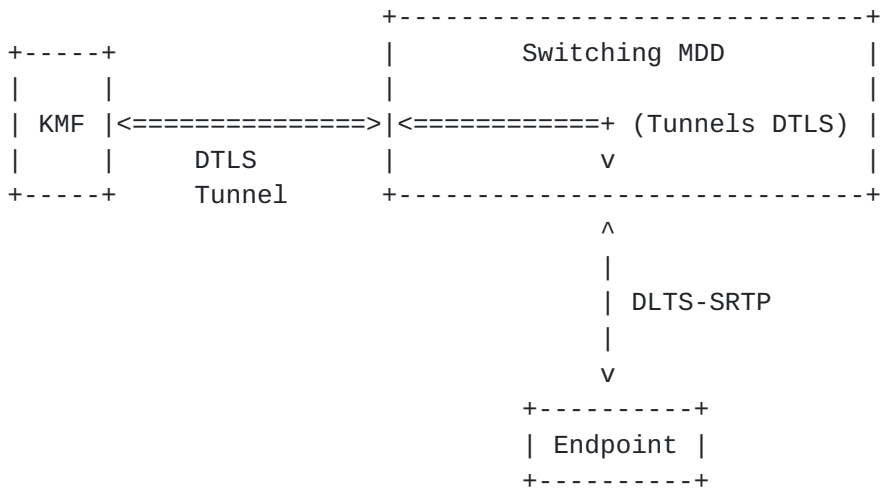


Figure 1: DTLS Tunnel to KMF

The three entities involved in this communication flow are the endpoint, the MDD, and the KMF. The behavior of each entity is described in Section 5.

The KMF is a logical function that might might be co-resident with a key management server operated by an enterprise, reside in one of the endpoints participating in the conference, or elsewhere that is trusted with E2E keying material. This document does not preclude any location, only requiring that the KMF not allow the MDD to gain access to the E2E keying material by following the operations defined.

4. Example Message Flows

This section provides an example message flow to help clarify the procedures described later in this document. Note that it is assumed that a mutually authenticated DTLS association is already established between the MDD and KMF for the purpose of sending tunneled messages.

Once the tunnel is established, it is possible for the MDD to relay the DTLS messages between the endpoint and the KMF. Figure 2 shows a message flow wherein the endpoint uses DTLS-SRTP to establish an association with the KMF. In the process, the MDD shares its supported SRTP protection profile information (see [RFC5764]) and the KMF shares HBH keying material and selected cipher with the MDD. The message used to tunnel the DTLS messages is named "Tunnel" and can include Profiles or Key Info data.

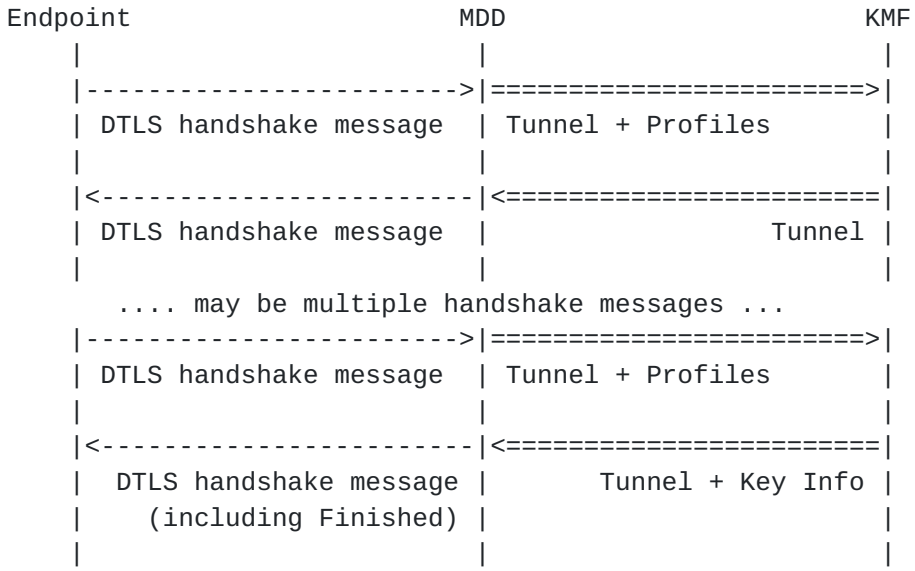


Figure 2: Sample DTLS-SRTP Exchange via the Tunnel

Each of these tunneled messages on the right-hand side of Figure 2 is a message of type "Tunnel" (see Section 6). Each message contains the following information:

- Protocol version
- Association ID
- DTLS message being tunneled

Additionally, all messages sent by the MDD will contain MDD-supported SRTP protection profiles at the end of the Tunnel message. The KMF will need to select a common profile supported by both the endpoint and the MDD to ensure that hop-by-hop operations can successfully be performed.

Further, the KMF will provide the SRTP [RFC3711] keying material for HBH operations at the time it sends a DTLS Finished message to the endpoint via the tunnel. The MDD would extract this Key Info when received and use it for hop-by-hop encryption and authentication. The delivery of the keying information along with the completion of the DTLS handshake ensures the delivery of the keying information is fate shared with completion of the DTLS handshake so that the MDD is guaranteed to have the HBH keying information before it receives any media that is encrypted or authenticated with that key.

5. Tunneling Procedures

The following sub-sections explain in detail the expected behavior of the endpoint, the media distribution device (MDD), and the key management function (KMF).

It is important to note that the tunneling protocol described in this document is not an extension to TLS [RFC5246] or DTLS [RFC6347]. Rather, it is a protocol that transports endpoint or KMF-generated DTLS messages as data inside of the DTLS association established between the MDD and KMF.

5.1. Endpoint Procedures

The endpoint follows the procedures outlined for DTLS-SRTP [RFC5764] in order to establish the keys used for encryption and authentication. The endpoint uses the normal procedures to establish a DTLS-SRTP association with the KMF.

5.2. Media Distribution Device Tunneling Procedures

The MDD, acting as a client, establishes a mutually authenticated DTLS association between itself and a KMF to relay DTLS messages from any number of endpoints to that KMF. This MDD initiated DTLS

association is called a "tunnel" to differentiate it from the DTLS associations initiated by endpoints.

The MDD MUST establish a tunnel with the KMF in advance of, or no later than the point, when an endpoint attempts to establish a DTLS association with the KMF.

The MDD MUST forward all messages received from an endpoint for a given DTLS association through the same tunnel if more than one tunnel has been established between it and a KMF. An MDD is not precluded from establishing more than one tunnel to a given KMF.

The MDD MUST assign a tunnel-unique "association identifier" for each endpoint-initiated DTLS association and include it in all messages forwarded to the KMF. The KMF will subsequently include in this identifier in all messages it sends so that the MDD can map messages received via a given tunnel and forward those messages to the correct endpoint.

The tunnel protocol enables the KMF to separately provide HBH keying material to the MDD for each of the individual endpoint DTLS associations, though the MDD cannot decrypt messages between the KMF and endpoints.

When a DTLS message is received by the MDD from an endpoint, it forwards the UDP payload portion of that message to the KMF encapsulated in a Tunnel + Profiles message (see Section 6). The Tunnel + Profiles message allows the MDD to signal which SRTP protection profiles it supports for HBH operations.

The MDD MUST support the same list of protection profiles for the life of a given endpoint's DTLS association, which is represented by the association identifier.

When a message from the KMF includes "Key Info," the MDD MUST extract the cipher and keying material conveyed in order to subsequently perform HBH encryption and authentication operations for RTP and RTCP packets sent between it and an endpoint. Since the HBH keying material will be different for each endpoint, the MDD uses the tunnel-unique association identifier included by the KMF to ensure that the HBH keying material is used with the correct endpoint.

The MDD MUST forward all messages received from either the endpoint or the KMF to ensure proper communication between those two entities.

5.3. Key Management Function Tunneling Procedures

The KMF MUST be prepared to establish one or more tunnels (DTLS associations) with the MDD for the purpose of relaying DTLS messages between an endpoint and the KMF. The KMF acts as a server and the MDD acts as a client to establish a tunnel.

When the MDD relays a DTLS message from an endpoint, the MDD will include an association identifier that is unique per endpoint-originated DTLS association and is relayed via the tunnel. The association identifier remains constant for the life of the DTLS association. The KMF identifies each distinct endpoint-originated DTLS association by the association identifier and the tunnel over which the DTLS association was established.

The KMF MUST encapsulate the DTLS message inside a Tunnel message (see Section 6) when sending a message to an endpoint.

The KMF MUST use the same association identifier in messages sent to an endpoint and MUST send all messages for a given endpoint DTLS association via the same tunnel. This ensures the MDD can forward the messages to the correct endpoint.

The KMF extracts tunneled DTLS messages from an endpoint and acts on those messages as if that endpoint had established the DTLS association directly with the KMF, which is acting as the server and the endpoint as the client. The handling of the messages and certificates is exactly the same as normal DTLS-SRTP procedures between endpoints.

The KMF MUST send a DTLS Finished message to the endpoint at the point the the DTLS handshake completes. This is accomplished by utilizing the Tunnel + Key Info message. The Key Info includes the selected cipher (i.e. protection profile), , MKI [RFC3711] value (if any), SRTP master keys, and SRTP master salt values.

The KMF MUST select a cipher that is supported by both the endpoint and the MDD for proper operations.

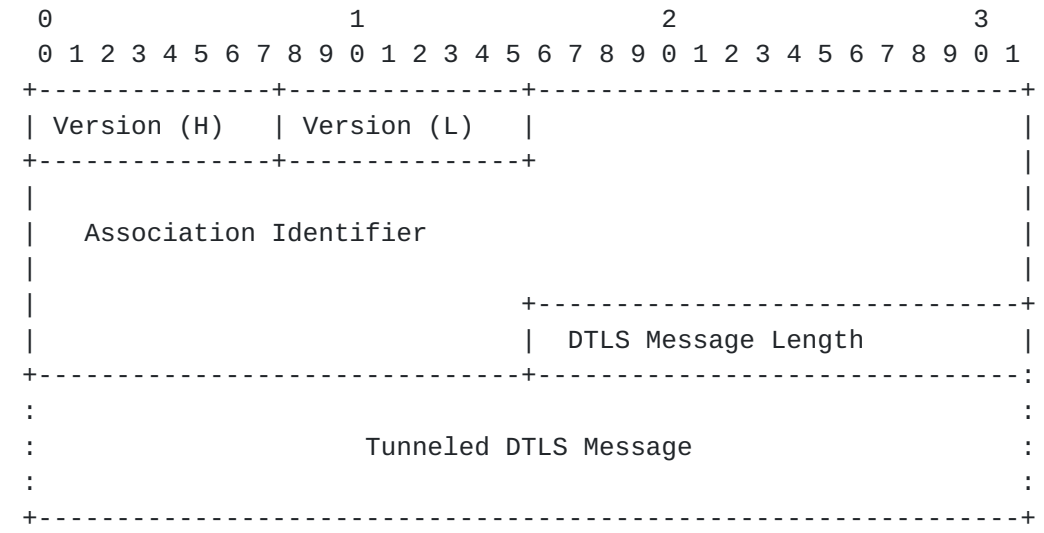
6. Tunneling Protocol

The tunneling protocol is transmitted over the DTLS association established between the MDD and KMF as application data. The basic message is referred to as the Tunnel message. The MDD will append supported SRTP protection profiles to all Tunnel messages it sends, forming the Tunnel + Profiles message. The KMF will append information necessary for the MDD to perform HBH encryption and authentication as it transmits the DTLS Finished message to the

endpoint, forming the Tunnel + Key Info message. The Tunnel, Tunnel + Profiles, and Tunnel + Key Info messages are detailed in the following sub-sections.

6.1. Tunnel Message

Tunneled DTLS messages are transported via the "Tunnel" message as application data between the MDD and the KMF. The "Tunnel" Message has the following format:



Version (H): This is the protocol major version number (set to 0x01).

Version (L): This is the protocol minor version number (set to 0x00).

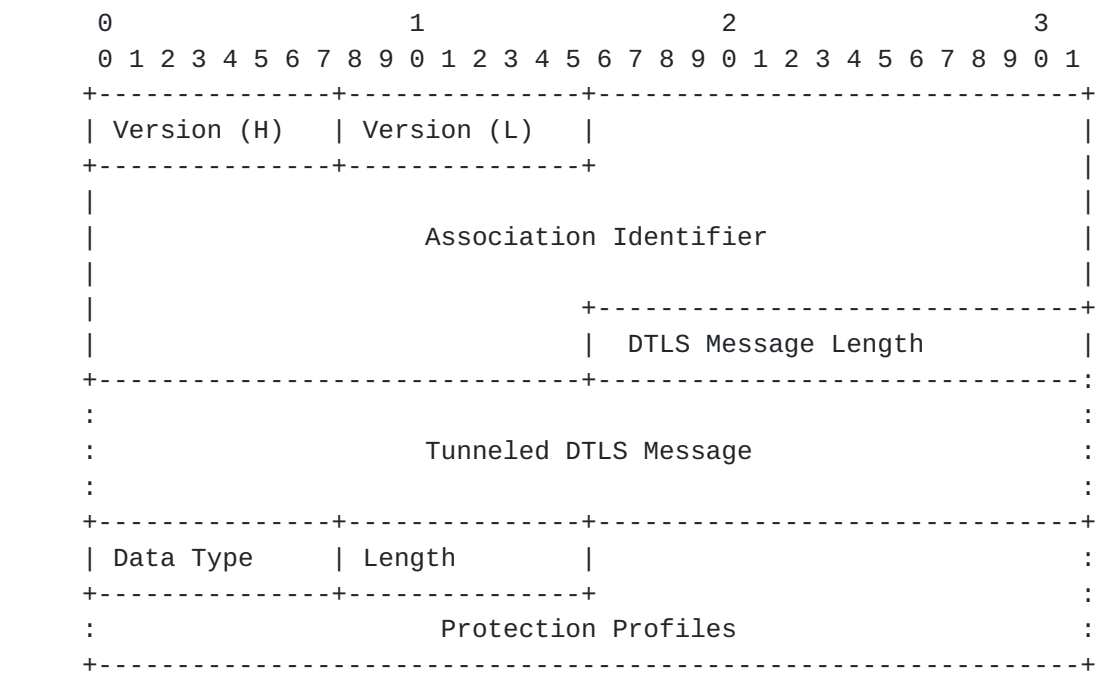
Association Identifier: This is the 16-octet association identifier.

DTLS Message Length: Length in octets of following Tunneled DTLS Message.

Tunneled DTLS Message: This is the DTLS message exchanged between the endpoint and KMF.

6.2. Tunnel Message + Profiles

Each Tunnel message transmitted by the MDD contains an array of SRTP protection profiles at the end of the message. The format of the message is shown below:



Beyond the fields included in the Tunnel message, this message introduces the following additional fields.

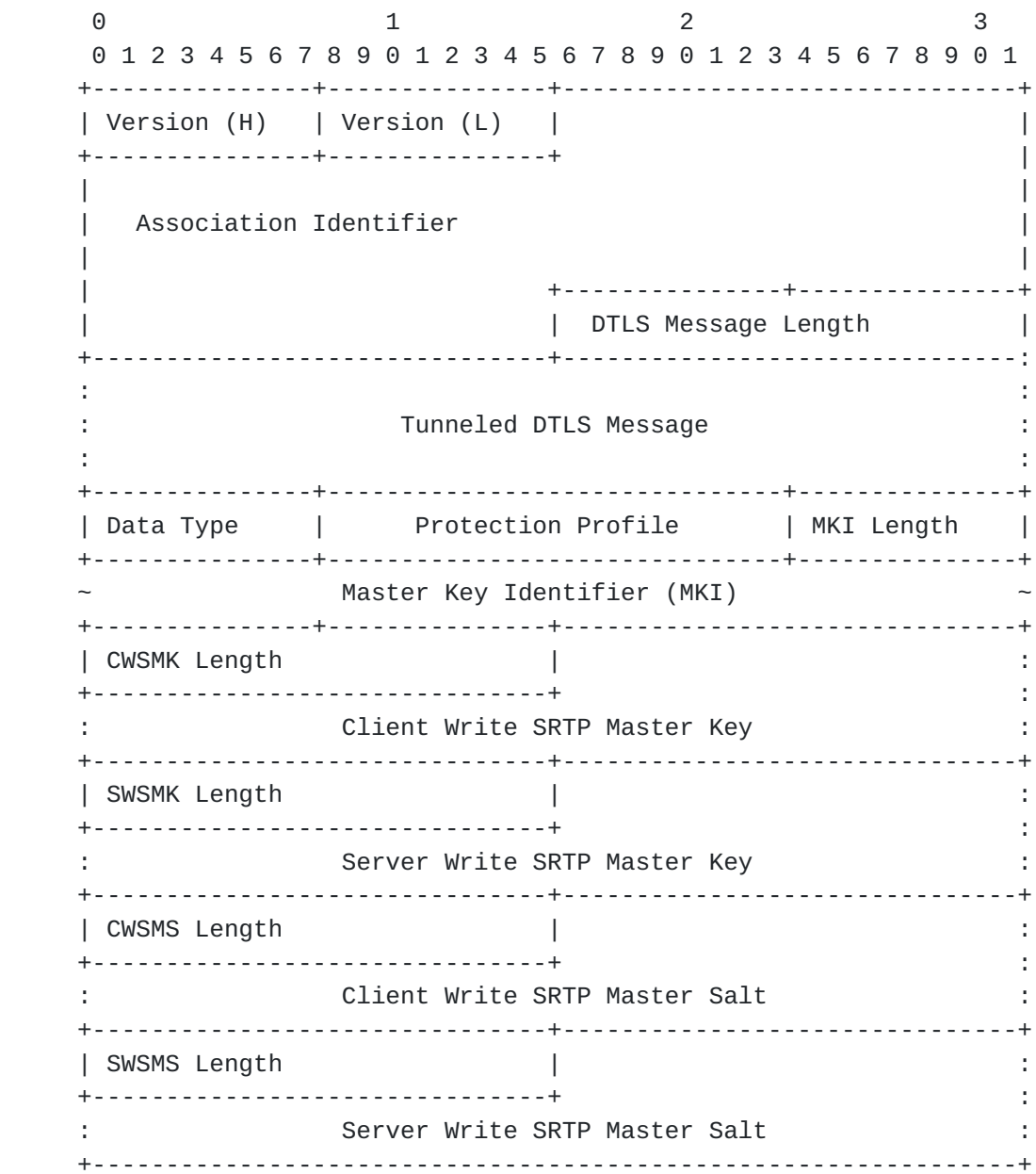
Data Type: Indicates the type of data that follows. For MDD supported SRTP protection profiles, this value is 0x01.

Length: This is the length in octets of the protection profiles. This length must be greater than or equal to 2.

Protection Profiles: This is an array of two-octet SRTP protection profile values as per [RFC5764], with each value represented in network byte order.

6.3. Tunnel Message + Key Info

When the KMF has key information to share with the MDD so it can perform HBH encryption and authentication on received media packets, the KMF will send a Tunnel message with the Key Info appended as shown below:



Beyond the fields included in the Tunnel message, this message introduces the following additional fields.

Data Type: Indicates the type of data that follows. For key information, this value is 0x02.

Protection Profile: This is the SRTP protection profile (see [RFC5764]) the MDD MUST use to encrypt and decrypt packets sent and received between itself and the endpoint.

MKI Length: This is the length in octets of the MKI field. A value of zero indicates that the MKI field is absent.

CWSMK Length: The length of the "Client Write SRTP Master Key" field.

Client Write SRTP Master Key: The value of the SRTP master key used by the client (endpoint).

SWSMK Length: The length of the "Server Write SRTP Master Key" field.

Server Write SRTP Master Key: The value of the SRTP master key used by the server (MDD).

CWSMS Length: The length of the "Client Write SRTP Master Salt" field.

Client Write SRTP Master Salt: The value of the SRTP master salt used by the client (endpoint).

SWSMS Length: The length of the "Server Write SRTP Master Salt" field.

Server Write SRTP Master Salt: The value of the SRTP master salt used by the server (MDD).

7. To-Do List

The MDD and KMF may need to coordinate or exchange a "conference identifier" common to the endpoints a MDD is bridging together. Alternatively, information the KMF needs to know about conference-to-endpoint correlations might be satisfied by getting info directly from the endpoints, or some trusted entity on their behalf, via some other means. Need to revisit this design choice in the context of all the alternatives.

8. IANA Considerations

There are no IANA considerations for this document.

9. Security Considerations

TODO - Much more needed.

The encapsulated data is protected by the DTLS session from the endpoint to KMF and the MDD is merely an on path entity. This does not introduce any additional security concerns beyond a normal DTLS-SRTP session.

The HBH keying material is protected by the mutual authenticated DTLS session between the MDD and KMF. The KMF MUST ensure that it only

forms associations with authorized MDDs or it could hand HBH keying information to untrusted parties.

The supported profile information send from the MDD to the KMF is not particularly sensitive as it only provides the crypt algorithms supported by the MDD but it is still protected by the DTLS session from the MDD to KMF.

10. Acknowledgments

The author would like to thank David Benham and Cullen Jennings for reviewing this document and providing constructive comments.

11. 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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <http://www.rfc-editor.org/info/rfc3550>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <http://www.rfc-editor.org/info/rfc3711>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

[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>.

Author's Address

Paul Jones
Cisco Systems
7025 Kit Creek Rd.
Research Triangle Park, North Carolina 27709
USA

Phone: +1 919 476 2048
Email: paulej@packetizer.com