Network Working Group
Internet-Draft
Expires: November 13, 2020

S. Garcia Murillo, Ed.
A. Gouaillard, Ed.
CoSMo Software Consulting, Pte Ltd
May 12, 2020

End to End Media Encryption Procedures
draft-murillo-perc-lite-01

Abstract

In some conferencing scenarios, it is desirable for an intermediary to be able to manipulate some RTP parameters, while still providing strong end-to-end security guarantees. This document defines a procedure to perform end to end media authenticated encryption.

Table of Contents

1. Introduction

RTP-based real-time multi-party interactive media conferencing is in widespread use today. Many of the deployments use one or more centrally located media distribution devices that perform selective forwarding of mixed-media streams received from the participating endpoints.

These conferences require security to ensure that the RTP media and related metadata of the conference is kept private and only available to the set of invited participants and other devices trusted by those participants with their media. At the same time, multi-party media conferences need source authentication and integrity checks to protect against modifications, insertions, and replay attacks.

To date, deployment models for these multi-party media distribution devices do not enable the devices to perform their functions without having keys to decrypt the participants' media. This trust model has limitations and prevents or hampers deployment of secure RTP conferencing in a multitude of cases, including outsourcing, legal requirements on confidentiality, and utilization of virtualized servers.

This specification defines an End to End Media Encryption procedure, so the media distribution devices can perform their media distribution function but without having access to the participant media, while focusing on introducing the minimun amount of changes on both the endpoints and the media distributor.

2. Terminology

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 BCP 14, RFC 2119 [RFC2119].

3. Overview

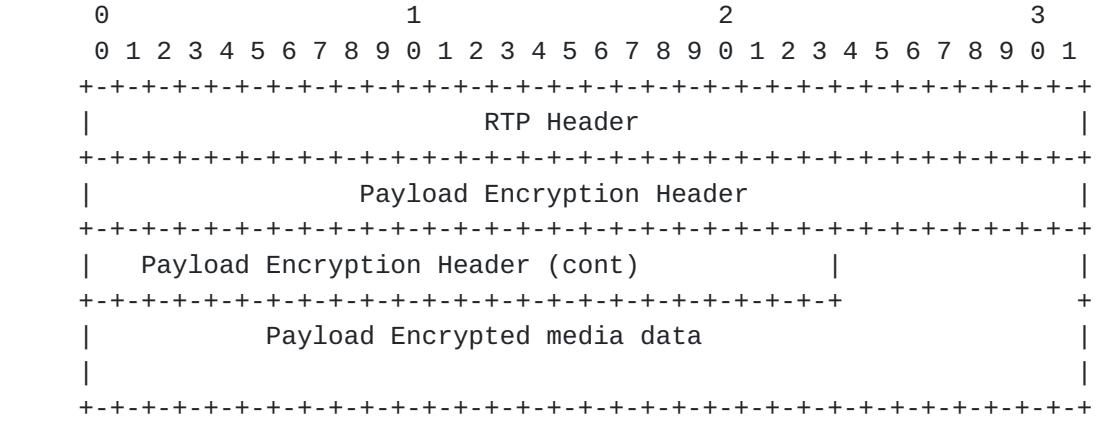
In order to prevent the Media Distributor (MD) to access the contents of the media passing through the system, the RTP media payload will be encrypted using SRTP, that will provide encryption, message authentication and integrity, and replay protection. The RECOMMENDED cipher to be used is AES-GCM.

All the participants on the conference will share a media encryption/ decryption key. How to distribute the shared key to all the participants of the conference is out of scope of this draft.

The encrypted media payload will be self-contained, so it can be decrypted by the media receiver side, regardless any RTP transformation done by the intermediary hosts.

4. Procedures at the Media Sender

The Media Sender will encode the media streams and packetize the encoded stream into RTP packets according to the codec specific specifications. Once done, the RTP payload will be replaced with an encrypted version of the media payload, prepending the required information for decrypting it.



RTP packet with E2E encrypted Media pPyload

As the payload will be encrypted, the sender MUST add a Frame Marking header extension with the appropriate values so any intermediate MD can perform the routing/SFU logic on the RTP stream.

Note that the SRTP encryption may also add trailing data (MKI and authentication tag) to the encrypted media, so the size overhead of this end to end media protection will vary.

Once the RTP packet payload is replaced, the media sender will be able to continue the RTP processing normally, like RTX, RED/FEC generation and SRTP/DTLS encryption.

5. Procedures at the Media Distributor

As the media payload of the RTP packets is encrypted, the MD MUST use the Frame Marking extension information to check for I frames, start/end of frame marks or SVC layer indexes instead of looking into the media data.

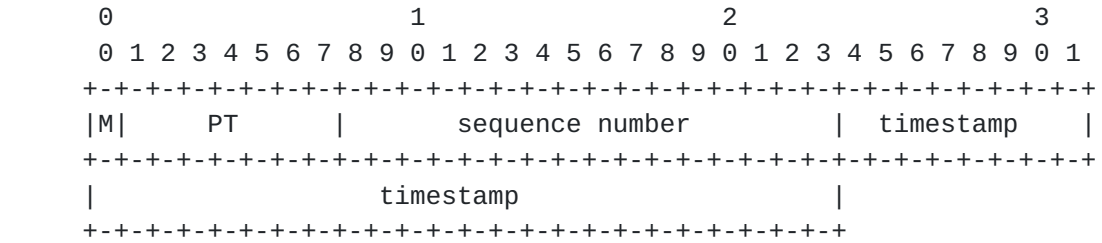
No other actions are required in the MD and it will be able to freely modify any RTP header information, like sequence number rewriting, add or remove RTP header extensions without affecting the encrypted media data.

6. Procedures at the Media Receiver

The process at the receiver is the reverse one used at the sender. Once an RTP packet has been received, the media receiver will create a new auxiliary RTP packet from the RTP packet payload, prepped the first byte of the RTP header with the default values v=2, x=0 and p=0 (0x80), and perform the SRTP decryption. If the decryption is successful, it will replace the payload of the original RTP packet with the decrypted payload of the auxiliary RTP packet.

7. Payload Encryption Header

The PEH payload will continue all the required information to decode the packet, and it will be very similar to an RTP header:



Payload Encryption Header

The values of the M, PT, sequence number and timestamp are the values from the original RTP header packet.

8. RTX/RED/FEC procedures

The procedures for NACK/RTX and RED/FEC are not affected by the end to end media encryption procedure as they will be applied after the media has been encrypted on the sender side, and before the end to end media encryption on the receiver side.

9. References

9.1. Normative References

[I-D.ietf-avtext-framemarking] Zanaty, M., Berger, E., and S. Nandakumar, "Frame Marking RTP Header Extension", draft-ietf-avtext-framemarking-10 (work in progress), November 2019.

[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>.

[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, <https://www.rfc-editor.org/info/rfc3711>.

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

[RFC6904] Lennox, J., "Encryption of Header Extensions in the Secure Real-time Transport Protocol (SRTP)", RFC 6904, DOI 10.17487/RFC6904, April 2013, <https://www.rfc-editor.org/info/rfc6904>.

[RFC7714] McGrew, D. and K. Igoe, "AES-GCM Authenticated Encryption in the Secure Real-time Transport Protocol (SRTP)", RFC 7714, DOI 10.17487/RFC7714, December 2015, <https://www.rfc-editor.org/info/rfc7714>.

9.2. Informative References

[I-D.ietf-perc-private-media-framework]
Jones, P., Benham, D., and C. Groves, "A Solution Framework for Private Media in Privacy Enhanced RTP Conferencing (PERC)", draft-ietf-perc-private-media-framework-12 (work in progress), June 2019.

Appendix A. Change Log

Note to RFC Editor: if this document does not obsolete an existing RFC, please remove this appendix before publication as an RFC.

Appendix B. Open Issues

Note to RFC Editor: please remove this appendix before publication as an RFC.

Authors' Addresses

Sergio Garcia Murillo (editor)
CoSMo Software Consulting, Pte Ltd

EMail: sergio.garcia.murillo@cosmosoftware.io

Alexandre Gouaillard (editor)
CoSMo Software Consulting, Pte Ltd

EMail: Alex.GOUAILLARD@cosmosoftware.io