Internet Engineering Task Force (IETF)
Request for Comments: 7202
Category: Informational
ISSN: 2070-1721

C. Perkins
University of Glasgow
M. Westerlund
Ericsson

April 2014

Securing the RTP Framework: Why RTP Does Not Mandate a Single Media Security Solution

Abstract

This memo discusses the problem of securing real-time multimedia sessions. It also explains why the Real-time Transport Protocol (RTP) and the associated RTP Control Protocol (RTCP) do not mandate a single media security mechanism. This is relevant for designers and reviewers of future RTP extensions to ensure that appropriate security mechanisms are mandated and that any such mechanisms are specified in a manner that conforms with the RTP architecture.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7202.

Table of Contents

1. Introduction

The Real-time Transport Protocol (RTP) [RFC3550] is widely used for voice over IP, Internet television, video conferencing, and other real-time and streaming media applications. Despite this use, the basic RTP specification provides only limited options for media security and defines no standard key exchange mechanism. Rather, a number of extensions are defined that can provide confidentiality and authentication of RTP media streams and RTP Control Protocol (RTCP) messages. Other mechanisms define key exchange protocols. This memo outlines why it is appropriate that multiple extension mechanisms are defined rather than mandating a single security and keying mechanism for all users of RTP.

The IETF policy "Strong Security Requirements for Internet Engineering Task Force Standard Protocols" [RFC3365] (the so-called "Danvers Doctrine") states that "we MUST implement strong security in all protocols to provide for the all too frequent day when the protocol comes into widespread use in the global Internet". The security mechanisms defined for use with RTP allow these requirements

to be met. However, since RTP is a protocol framework that is suitable for a wide variety of use cases, there is no single security mechanism that is suitable for every scenario. This memo outlines why this is the case and discusses how users of RTP can meet the requirement for strong security.

This document provides high-level guidance on how to handle security issues for the various types of components within the RTP framework as well as the role of the service or application using RTP to ensure strong security is implemented. This document does not provide the guidance that an individual implementer, or even specifier of an RTP application, really can use to determine what security mechanism they need to use; that is not intended with this document.

A non-exhaustive list of the RTP security options available at the time of this writing is outlined in [RFC7201]. This document gives an overview of the available RTP solutions and provides guidance on their applicability for different application domains. It also attempts to provide an indication of actual and intended usage at the time of writing as additional input to help with considerations such as interoperability, availability of implementations, etc.

2. RTP Applications and Deployment Scenarios

The range of application and deployment scenarios where RTP has been used includes, but is not limited to, the following:

- Point-to-point voice telephony;
- Point-to-point video conferencing and telepresence;
- Centralized group video conferencing and telepresence, using a Multipoint Conference Unit (MCU) or similar central middlebox;
- Any Source Multicast (ASM) video conferencing using the lightweight sessions model (e.g., the Mbone conferencing tools);
- Point-to-point streaming audio and/or video (e.g., on-demand TV or movie streaming);
- Source-Specific Multicast (SSM) streaming to large receiver groups (e.g., IPTV streaming by residential ISPs or the Third Generation Partnership Project (3GPP) Multimedia/Broadcast Multicast Service [T3GPP.26.346]);
- Replicated unicast streaming to a group of receivers;

- Interconnecting components in music production studios and video editing suites;
- Interconnecting components of distributed simulation systems; and
- Streaming real-time sensor data (e.g., electronic Very Long Baseline Interferometry (e-VLBI) radio astronomy).

As can be seen, these scenarios vary from point-to-point sessions to very large multicast groups, from interactive to non-interactive, and from low bandwidth (kilobits per second) telephony to high bandwidth (multiple gigabits per second) video and data streaming. While most of these applications run over UDP [RFC0768], some use TCP [RFC0793] [RFC4614] or the Datagram Congestion Control Protocol (DCCP) [RFC4340] as their underlying transport. Some run on highly reliable optical networks, while others use low-rate unreliable wireless networks. Some applications of RTP operate entirely within a single trust domain, while others run interdomain with untrusted (and, in some cases, potentially unknown) users. The range of scenarios is wide and growing both in number and in heterogeneity.

3. RTP Media Security

The wide range of application scenarios where RTP is used has led to the development of multiple solutions for securing RTP media streams and RTCP control messages, considering different requirements.

Perhaps the most widely applicable of these security options is the Secure RTP (SRTP) framework [RFC3711]. This is an application-level media security solution, encrypting the media payload data (but not the RTP headers) to provide confidentiality and supporting source origin authentication as an option. SRTP was carefully designed to be low overhead, including operating on links subject to RTP header compression, and to support the group communication and third-party performance monitoring features of RTP across a range of networks.

SRTP is not the only media security solution for RTP, however, and alternatives can be more appropriate in some scenarios, perhaps due to ease of integration with other parts of the complete system. In addition, SRTP does not address all possible security requirements, and other solutions are needed in cases where SRTP is not suitable. For example, ISMACryp payload-level confidentiality [ISMACryp2] is appropriate for some types of streaming video application, but is not suitable for voice telephony, and uses features that are not provided by SRTP.

The range of available RTP security options, and their applicability to different scenarios, is outlined in [RFC7201]. At the time of this writing, there is no media security protocol that is appropriate for all the environments where RTP is used. Multiple RTP media security protocols are expected to remain in wide use for the foreseeable future.

4. RTP Session Establishment and Key Management

A range of different protocols for RTP session establishment and key exchange exist, matching the diverse range of use cases for the RTP framework. These mechanisms can be split into two categories: those that operate in band on the media path and those that are out of band and operate as part of the session establishment signaling channel. The requirements for these two classes of solutions are different, and a wide range of solutions have been developed in this space.

A more-detailed survey of requirements for media security management protocols can be found in [RFC5479]. As can be seen from that memo, the range of use cases is wide, and there is no single key management protocol that is appropriate for all scenarios. The solutions have been further diversified by the existence of infrastructure elements, such as authentication systems, that are tied to the key management. The most important and widely used keying options for RTP sessions at the time of this writing are described in [RFC7201].

5. On the Requirement for Strong Security in Framework Protocols

The IETF requires that all protocols provide a strong, mandatory-to-implement security solution [RFC3365]. This is essential for the overall security of the Internet to ensure that all implementations of a protocol can interoperate in a secure way. Framework protocols offer a challenge for this mandate, however, since they are designed to be used by different classes of applications in a wide range of different environments. The different use cases for the framework have different security requirements, and implementations designed for different environments are generally not expected to interwork.

RTP is an example of a framework protocol with wide applicability. The wide range of scenarios described in Section 2 show the issues that arise in mandating a single security mechanism for this type of framework. It would be desirable if a single media security solution, and a single key management solution, could be developed that is suitable for applications across this range of use scenarios. The authors are not aware of any such solution, however, and believe it is unlikely that any such solution will be developed. In part, this is because applications in the different domains are not intended to interwork, so there is no incentive to develop a single

mechanism. More importantly, though, the security requirements for the different usage scenarios vary widely, and an appropriate security mechanism in one scenario simply does not work for some other scenarios.

For a framework protocol, it appears that the only sensible solution to the strong security requirement of [RFC3365] is to develop and use building blocks for the basic security services of confidentiality, integrity protection, authorization, authentication, and so on. When new uses for the framework protocol arise, they need to be studied to determine if the existing security building blocks can satisfy the requirements, or if new building blocks need to be developed.

Therefore, when considering the strong and mandatory-to-implement security mechanism for a specific class of applications, one has to consider what security building blocks need to be integrated, or if any new mechanisms need to be defined to address specific issues relating to this new class of application. To maximize interoperability, it is important that common media security and key management mechanisms are defined for classes of application with similar requirements. The IETF needs to participate in this selection of security building blocks for each class of applications that use the protocol framework and are expected to interoperate, in cases where the IETF has the appropriate knowledge of the class of applications.

6. Securing the RTP Framework

The IETF requires that protocols specify mandatory-to-implement (MTI) strong security [RFC3365]. This applies to the specification of each interoperable class of application that makes use of RTP. However, RTP is a framework protocol, so the arguments made in Section 5 also apply. Given the variability of the classes of application that use RTP, and the variety of the currently available security mechanisms described in [RFC7201], no one set of MTI security options can realistically be specified that apply to all classes of RTP applications.

Documents that define an interoperable class of applications using RTP are subject to [RFC3365], and thus need to specify MTI security mechanisms. This is because such specifications do fully specify interoperable applications that use RTP. Examples of such documents under development in the IETF at the time of this writing are "WebRTC Security Architecture" [WebRTC-SEC] and "Real Time Streaming Protocol 2.0 (RTSP)" [RTSP]. It is also expected that a similar document will be produced for voice-over-IP applications using SIP and RTP.

The RTP framework includes several extension points. Some extensions can significantly change the behavior of the protocol to the extent that applications using the extension form a separate interoperable class of applications to those that have not been extended. Other extension points are defined in such a manner that they can be used (largely) independently of the class of applications using RTP. Two important extension points that are independent of the class of applications are RTP payload formats and RTP profiles.

An RTP payload format defines how the output of a media codec can be used with RTP. At the time of this writing, there are over 70 RTP payload formats defined in published RFCs, with more in development. It is appropriate for an RTP payload format to discuss the specific security implications of using that media codec with RTP. However, an RTP payload format does not specify an interoperable class of applications that use RTP since, in the vast majority of cases, a media codec and its associated RTP payload format can be used with many different classes of application. As such, an RTP payload format is neither secure in itself nor something to which [RFC3365] applies. Future RTP payload format specifications need to explicitly state this and include a reference to this memo for explanation. It is not appropriate for an RTP payload format to mandate the use of SRTP [RFC3711], or any other security building blocks, since that RTP payload format might be used by different classes of application that use RTP and that have different security requirements.

RTP profiles are larger extensions that adapt the RTP framework for use with particular classes of application. In some cases, those classes of application might share common security requirements so that it could make sense for an RTP profile to mandate particular security options and building blocks (the RTP/SAVP profile [RFC3711] is an example of this type of RTP profile). In other cases, though, an RTP profile is applicable to such a wide range of applications that it would not make sense for that profile to mandate particular security building blocks be used (the RTP/AVPF profile [RFC4585] is an example of this type of RTP profile, since it provides building blocks that can be used in different styles of application). A new RTP profile specification needs to discuss whether or not it makes sense to mandate particular security building blocks that need to be used with all implementations of that profile; however, there is no expectation that all RTP profiles will mandate particular security solutions. RTP profiles that do not specify an interoperable usage for a particular class of RTP applications are neither secure in themselves nor something to which [RFC3365] applies; any future RTP profiles in this category need to explicitly state this with justification and include a reference to this memo.

7. Conclusions

The RTP framework is used in a wide range of different scenarios with no common security requirements. Accordingly, neither SRTP [RFC3711] nor any other single media security solution or keying mechanism can be mandated for all uses of RTP. In the absence of a single common security solution, it is important to consider what mechanisms can be used to provide strong and interoperable security for each different scenario where RTP applications are used. This will require analysis of each class of application to determine the security requirements for the scenarios in which they are to be used, followed by the selection of MTI security building blocks for that class of application, including the desired RTP traffic protection and key management. A non-exhaustive list of the RTP security options available at the time of this writing is outlined in [RFC7201]. It is expected that each class of application will be supported by a memo describing what security options are mandatory to implement for that usage scenario.

8. Security Considerations

This entire memo is about mandatory-to-implement security.

9. Acknowledgements

Thanks to Ralph Blom, Hannes Tschofenig, Dan York, Alfred Hoenes, Martin Ellis, Ali Begen, Keith Drage, Ray van Brandenburg, Stephen Farrell, Sean Turner, John Mattsson, and Benoit Claise for their feedback.

10. Informative References

[ISMACryp2] Internet Streaming Media Alliance (ISMA), "ISMA Encryption and Authentication Version 2.0", November 2007, <http://www.oipf.tv/images/site/DOCS/mpegif/ISMA/isma_easpec2.0.pdf>.

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[RFC3365] Schiller, J., "Strong Security Requirements for Internet Engineering Task Force Standard Protocols", BCP 61, RFC 3365, August 2002.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, March 2006.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, July 2006.

[RFC4614] Duke, M., Braden, R., Eddy, W., and E. Blanton, "A Roadmap for Transmission Control Protocol (TCP) Specification Documents", RFC 4614, September 2006.

[RFC5479] Wing, D., Fries, S., Tschofenig, H., and F. Audet, "Requirements and Analysis of Media Security Management Protocols", RFC 5479, April 2009.

[RFC7201] Westerlund, M. and C. Perkins, "Options for Securing RTP Sessions", RFC 7201, April 2014.

[RTSP] Schulzrinne, H., Rao, A., Lanphier, R., Westerlund, M., and M. Stiemerling, "Real Time Streaming Protocol 2.0 (RTSP)", Work in Progress, February 2014.

[T3GPP.26.346] 3GPP, "Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs", 3GPP TS 26.346 10.7.0, March 2013, <http://www.3gpp.org/ftp/Specs/html-info/26346.htm>.

[WebRTC-SEC] Rescorla, E., "WebRTC Security Architecture", Work in Progress, February 2014.

Authors' Addresses

Colin Perkins
University of Glasgow
School of Computing Science
Glasgow G12 8QQ
United Kingdom

EMail: csp@csperkins.org
URI: http://csperkins.org/

Magnus Westerlund
Ericsson
Farogatan 6
Kista SE-164 80
Sweden

EMail: magnus.westerlund@ericsson.com