AVTCORE   M. Petit-Huguenin
Internet-Draft   Impedance Mismatch
Updates: 5764 (if approved)   G. Salgueiro
Intended status: Standards Track   Cisco Systems
Expires: September 7, 2015   March 6, 2015

Multiplexing Scheme Updates for Secure Real-time Transport Protocol (SRTP) Extension for Datagram Transport Layer Security (DTLS)
draft-petithuguenin-avtcore-rfc5764-mux-fixes-02

Abstract

This document defines how Datagram Transport Layer Security (DTLS), Real-time Transport Protocol (RTP), Real-time Transport Control Protocol (RTCP), Session Traversal Utilities for NAT (STUN), and Traversal Using Relays around NAT (TURN) packets are multiplexed on a single receiving socket. It overrides the guidance from SRTP Extension for DTLS [RFC5764], which suffered from three issues described and fixed in this document.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Section 5.1.2 of Secure Real-time Transport Protocol (SRTP) Extension for DTLS [RFC5764] defines a scheme for a Real-time Transport Protocol (RTP) [RFC3550] receiver to demultiplex Datagram Transport Layer Security (DTLS) [RFC6347], Session Traversal Utilities for NAT (STUN) [RFC5389] and Secure Real-time Transport Protocol (SRTP)/Secure Real-time Transport Control Protocol (SRTCP) [RFC3711] packets that are arriving on the RTP port. Unfortunately, this demultiplexing scheme has created three problematic issues:

1. It implicitly allocated codepoints for new STUN methods without an IANA registry reflecting these new allocations.

2. It implicitly allocated codepoints for new Transport Layer Security (TLS) ContentTypes without an IANA registry reflecting these new allocations.

3. It did not take into account the fact that the Traversal Using Relays around NAT (TURN) usage of STUN can create TURN channels that also need to be demultiplexed with the other packet types explicitly mentioned in Section 5.1.2 of RFC 5764.

These flaws in the demultiplexing scheme were unavoidably inherited by other documents, such as [RFC7345] and [I-D.ietf-mmusic-sdp-bundle-negotiation]. These will need to be corrected with the updates this document provides when it become normative.

1.1. Implicit Allocation of Codepoints for New STUN Methods

The demultiplexing scheme in [RFC5764] states that the receiver can identify the packet type by looking at the first byte. If the value of this first byte is 0 or 1, the packet is identified to be STUN. The problem that arises as a result of this implicit allocation is that this restricts the codepoints for STUN methods (as described in Section 18.1 of [RFC5389]) to values between 0x000 and 0x07F, which in turn reduces the number of possible STUN method codepoints assigned by IETF Review (i.e., the range from (0x000 - 0x7FF) from 2048 to only 128 and entirely obliterating those STUN method codepoints assigned by Designated Expert (i.e., the range 0x800 - 0xFFF). In fact, RFC 5764 implicitly (and needlessly) allocated a very large range of STUN methods, but at a minimum the IANA STUN Methods registry should properly reflect this.

There are only a few STUN method codepoints currently allocated. For this reason, simply marking the implicit allocations made by RFC 5764 in the STUN Method registry may create a shortage of codepoints at a time when interest in STUN and STUN Usages (especially TURN) is growing rapidly. Consequently, this document also changes the RFC 5764 packet identification algorithm to expand the range assigned to the STUN protocol from 0 - 1 to 0 - 19, as the values 2-19 are unused.

In addition to explicitly allocating STUN methods codepoints from 0x500 to 0xFFF as Reserved values, this document also updates the IANA registry such that the STUN method codepoints assigned via IETF Review are in the 0x000-0x27F range and those assigned via Designated Expert are in the 0x280-0x4FF range. The proposed changes to the STUN Method Registry is:

OLD:

```
0x000-0x7FF     IETF Review
0x800-0xFFF     Designated Expert
```

NEW:

0x000-0x27F	IETF Review
0x280-0x4FF	Designated Expert
0x500-0xFFF	Reserved

1.2. Implicit Allocation of New Codepoints for TLS ContentTypes

The demultiplexing scheme in [RFC5764] dictates that if the value of the first byte is between 20 and 63 (inclusive), then the packet is identified to be DTLS. The problem that arises is that this restricts the TLS ContentType codepoints (as defined in Section 12 of [RFC5246]) to this range, and by extension implicitly allocates ContentType codepoints 0 to 19 and 64 to 255. Unlike STUN, TLS is a mature protocol that is already well established and widely implemented and thus we expect only relatively few new codepoints to be assigned in the future. With respect to TLS packet identification, this document simply explicitly reserves the codepoints from 0 to 19 and from 64 to 255 so they are not inadvertently assigned in the future.

1.3. Multiplexing of TURN Channels

When used with ICE [RFC5245], an RFC 5764 implementation can receive packets on the same socket from three different paths, as shown in Figure 1:

1. Directly from the source

2. Through a NAT

3. Relayed by a TURN server

```
       +------+
       | TURN |<------------------------+
       +------+                         |
          |                             |
          | +-------------------------+ |
          | |                         | |
          v v                         | |
   NAT -----------                    | |
          | | +---------------------+ | |
          | | |                     | | |
          v v v                     | | |
      +----------+              +----------+
      | RFC 5764 |              | RFC 5764 |
      +----------+              +----------+
```

Figure 1: Packet Reception by an RFC 5764 Implementation

Even if the ICE algorithm succeeded in selecting a non-relayed path, it is still possible to receive data from the TURN server. For instance, when ICE is used with aggressive nomination the media path can quickly change until it stabilizes. Also, freeing ICE candidates is optional, so the TURN server can restart forwarding STUN connectivity checks during an ICE restart.

TURN channels are an optimization where data packets are exchanged with a 4-byte prefix, instead of the standard 36-byte STUN overhead (see Section 2.5 of [RFC5766]). The problem is that the RFC 5764 demultiplexing scheme does not define what to do with packets received over a TURN channel since these packets will start with a first byte whose value will be between 64 and 127 (inclusive). If the TURN server was instructed to send data over a TURN channel, then the current RFC 5764 demultiplexing scheme will reject these packets. Current implementations violate RFC 5764 for values 64 to 127 (inclusive) and they instead parse packets with such values as TURN. In order to prevent future documents from assigning values from the unused range to a new protocol, this document modifies the RFC 5764 demultiplexing algorithm to properly account for TURN channels.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] when they appear in ALL CAPS. When these words are not in ALL CAPS (such as "must" or "Must"), they have their usual English meanings, and are not to be interpreted as RFC 2119 key words.

3. RFC 5764 Updates

This document updates the text in Section 5.1.2 of [RFC5764] as follows:

OLD TEXT

The process for demultiplexing a packet is as follows. The receiver looks at the first byte of the packet. If the value of this byte is 0 or 1, then the packet is STUN. If the value is in between 128 and 191 (inclusive), then the packet is RTP (or RTCP, if both RTCP and RTP are being multiplexed over the same destination port). If the value is between 20 and 63 (inclusive), the packet is DTLS. This process is summarized in Figure 3.

```
                 +----------------+
                 | 127 < B < 192 -+--> forward to RTP
                 |                |
     packet -->  |  19 < B < 64  -+--> forward to DTLS
                 |                |
                 |       B < 2   -+--> forward to STUN
                 +----------------+
```

Figure 3: The DTLS-SRTP receiver's packet demultiplexing algorithm. Here the field B denotes the leading byte of the packet.

END OLD TEXT

NEW TEXT

The process for demultiplexing a packet is as follows. The receiver looks at the first byte of the packet. If the value of this byte is in between 0 and 19 (inclusive), then the packet is STUN. If the value is in between 128 and 191 (inclusive), then the packet is RTP (or RTCP, if both RTCP and RTP are being multiplexed over the same destination port). If the value is between 20 and 63 (inclusive), the packet is DTLS. If the value is between 64 and 127 (inclusive), the packet is TURN Channel. This process is summarized in Figure 3.

```
                    +----------------+
                    | 127 < B < 192 -+--> forward to RTP
                    |                |
                    |  63 < B < 128 -+--> forward to TURN Channel
        packet -->  |                |
                    |  19 < B < 64  -+--> forward to DTLS
                    |                |
                    |       B < 20  -+--> forward to STUN
                    +----------------+
```

Figure 3: The DTLS-SRTP receiver's packet demultiplexing algorithm. Here the field B denotes the leading byte of the packet.

END NEW TEXT

[[Note: we may want to use "<=" instead of "<" to make it easier on implementers.]]

4. Implementation Status

[[Note to RFC Editor: Please remove this section and the reference to [RFC6982] before publication.]]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC6982]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC6982], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

Note that there is currently no implementation declared in this section, but the intent is to add RFC 6982 templates here from implementers that support the modifications in this document.

5. Security Considerations

This document simply updates existing IANA registries and does not introduce any specific security considerations beyond those detailed in [RFC5764].

6. IANA Considerations

6.1. STUN Methods

This specification contains the registration information for 2816 STUN Methods codepoints, as explained in Section 1.1 and in accordance with the procedures defined in Section 18.1 of [RFC5389].

Value: 0x500-0xFFF

Name: Reserved

Reference: RFC5764, RFCXXXX

This specification also reassigns the ranges in the STUN Methods Registry as follow:

Range: 0x000-0x27F

Registration Procedures: IETF Review

Range: 0x280-0x4FF

Registration Procedures: Designated Expert

6.2. TLS ContentType

This specification contains the registration information for 212 TLS ContentType codepoints, as explained in Section 1.2 and in accordance with the procedures defined in Section 12 of [RFC5246].

Value: 0-19

Description: Reserved

DTLS-OK: N/A

Reference: RFC5764, RFCXXXX

Value: 64-255

Description: Reserved

DTLS-OK: N/A

Reference: RFC5764, RFCXXXX

6.3. TURN Channel Numbers

This specification contains the registration information for 32768 TURN Channel Numbers codepoints, as explained in Section 1.3 and in accordance with the procedures defined in Section 18 of [RFC5766].

Value: 0x8000-0xFFFF

Name: Reserved

Reference: RFCXXXX

[RFC EDITOR NOTE: Please replace RFCXXXX with the RFC number of this document.]

7. Acknowledgements

The implicit STUN Method codepoint allocations problem was first reported by Martin Thomson in the RTCWEB mailing-list and discussed further with Magnus Westerlund.

Thanks to Simon Perreault, Colton Shields and Cullen Jennings for the comments, suggestions, and questions that helped improve this document.

8. References

8.1. Normative References

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

[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.

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

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[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.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

8.2. Informative References

[RFC6982] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", RFC 6982, July 2013.

[RFC7345] Holmberg, C., Sedlacek, I., and G. Salgueiro, "UDP Transport Layer (UDPTL) over Datagram Transport Layer Security (DTLS)", RFC 7345, August 2014.

[I-D.ietf-mmusic-sdp-bundle-negotiation] Holmberg, C., Alvestrand, H., and C. Jennings, "Negotiating Media Multiplexing Using the Session Description Protocol (SDP)", draft-ietf-mmusic-sdp-bundle-negotiation-12 (work in progress), October 2014.

Appendix A. Release notes

This section must be removed before publication as an RFC.

A.1. Modifications between draft-petithuguenin-avtcore-rfc5764-mux-fixes-00 and draft-petithuguenin-avtcore-rfc5764-mux-fixes-01

- o Change affiliation.

Authors' Addresses

Marc Petit-Huguenin
Impedance Mismatch

Email: marc@petit-huguenin.org

Gonzalo Salgueiro
Cisco Systems
7200-12 Kit Creek Road
Research Triangle Park, NC 27709
US

Email: gsalguei@cisco.com