

Internet Engineering Task Force (IETF)
Request for Comments: 7345
Category: Standards Track
ISSN: 2070-1721

C. Holmberg
I. Sedlacek
Ericsson
G. Salgueiro
Cisco
August 2014

UDP Transport Layer (UDPTL) over Datagram Transport Layer Security (DTLS)

Abstract

This document specifies how the UDP Transport Layer (UDPTL) protocol, the predominant transport protocol for T.38 fax, can be transported over the Datagram Transport Layer Security (DTLS) protocol, how the usage of UDPTL over DTLS is indicated in the Session Description Protocol (SDP), and how UDPTL over DTLS is negotiated in a session established using the Session Initiation Protocol (SIP).

Status of This Memo

This is an Internet Standards Track document.

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). Further information on Internet Standards is available in 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/rfc7345>.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Conventions	5
3. Secure Channel	5
4. SDP Offerer/Answerer Procedures	6
4.1. General	6
4.2. Generating the Initial Offer	7
4.3. Generating the Answer	7
4.4. Offerer Processing of the Answer	7
4.5. Modifying the Session	7
5. Miscellaneous Considerations	8
5.1. Anonymous Calls	8
5.2. NAT Traversal	8
5.2.1. ICE Usage	8
5.2.2. STUN Interaction	8
5.3. Rekeying	9
5.4. Compatibility with UDPTL over UDP	9
6. Security Considerations	9
7. IANA Considerations	10
8. Acknowledgments	10
9. References	11
9.1. Normative References	11
9.2. Informative References	12
Appendix A. Examples	13
A.1. General	13
A.2. Basic Message Flow	13
A.3. Message Flow of T.38 Fax Replacing Audio Media Stream in an Existing Audio-Only Session	20

1. Introduction

While it is possible to transmit highly sensitive documents using traditional telephony encryption devices, secure fax on the Public Switched Telephone Network (PSTN) was never widely considered or prioritized. This was mainly because of the challenges involved with malevolent physical access to telephony equipment. As real-time communications transition to IP networks, where information might potentially be intercepted or spoofed, an appropriate level of security for fax that offers integrity and confidentiality protection is vital.

The overwhelmingly predominant fax transport protocol is UDPTL-based, as described in Section 9.1 of [ITU.T38.2010]. The protocol stack for fax transport using UDPTL is shown in Figure 1.

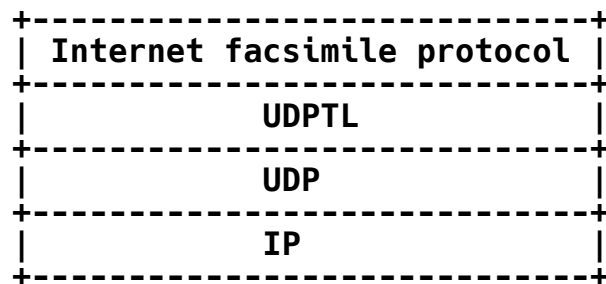


Figure 1: Protocol Stack for UDPTL over UDP

The following mechanisms are available for securing fax:

- o Annex H of [ITU.T30.2005] specifies an application-layer integrity and confidentiality protection of fax that is independent of the transport protocol and is based on the RSA algorithm for use with the T.30 telephony protocol by Group 3 facsimile equipment (G3FE).
- o [ITU.T38.2010] specifies fax transport over RTP/SAVP, which enables integrity and confidentiality protection of fax in IP networks.

Both of these mechanisms have been available for many years and never gained any significant adoption in the market. This has prompted an effort to develop an approach, based on open standards, for securing fax communications over an IP-based transport.

Telephony-based protocols like T.30 offer application-level security options like the RSA-based approach detailed in Annex H of the T.30 specification [ITU.T30.2005]. The problem is that it is very sparingly implemented and not enforced at the transport level.

It is worth noting that while T.38 over RTP offers a very viable option for such standards-based IP security solution using Secure Realtime Transport Protocol (SRTP), this fax-over-IP transport never gained any traction in the marketplace and accounts for a negligible percentage of fax-over-IP implementations.

Thus, security mechanisms offering integrity and confidentiality protection should be limited to UDPTL-based fax transport, which is the only broad-based fax-over-IP solution. The 3rd Generation Partnership Project (3GPP) launched a study on how best to provide secure fax in the IP Multimedia Subsystem (IMS) for UDPTL. Results of the study confirmed that this security was best achieved by using UDPTL over DTLS.

This document specifies fax transport using UDPTL over DTLS [RFC6347], which enables integrity and confidentiality protection of fax in IP networks. The protocol stack that enhances fax transport to offer integrity and confidentiality using UDPTL over DTLS is shown in Figure 2.

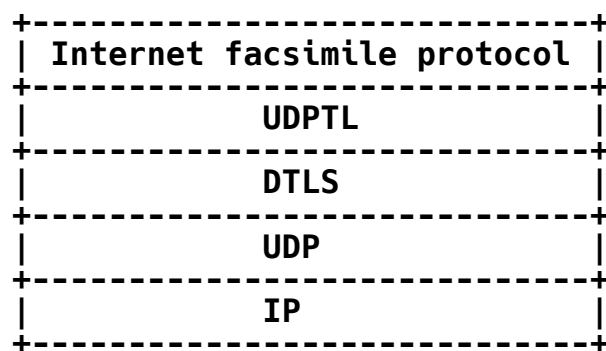


Figure 2: Protocol Stack for UDPTL over DTLS over UDP

The primary motivations for the mechanism in this document are:

- o The design of DTLS [RFC6347] is clearly defined and well understood, and implementations are widely available.
- o No DTLS extensions are required in order to enable UDPTL transport over DTLS.
- o Fax transport using UDPTL over DTLS only requires insertion of the DTLS layer between the UDPTL layer and the UDP layer, as shown in Figure 2. The UDPTL layer and the layers above the UDPTL layer require no modifications.

- o UDPTL [ITU.T38.2010] is by far the most widely deployed fax transport protocol in IP networks.
- o 3GPP and the IP fax community need a mechanism to transport UDPTL over DTLS in order to provide secure fax in SIP-based networks (including IMS).

This document specifies the transport of UDPTL over DTLS using the DTLS record layer "application_data" packets [RFC5246] [RFC6347].

Since the DTLS record layer "application_data" packet does not indicate whether it carries UDPTL or some other protocol, the usage of a dedicated DTLS association for transport of UDPTL needs to be negotiated, e.g., using the Session Description Protocol (SDP) [RFC4566] and the SDP offer/answer mechanism [RFC3264].

Therefore, this document specifies a new <proto> value [RFC4566] for the SDP media description ("m=" line) [RFC3264], in order to indicate UDPTL over DTLS in SDP messages [RFC4566].

2. Conventions

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].

DTLS uses the term "session" to refer to a long-lived set of keying material that spans DTLS associations. In this document, in order to be consistent with SIP/SDP usage of "session" terminology, we use "session" to refer to a multimedia session and use the term "DTLS session" to refer to the DTLS construct. We use the term "DTLS association" to refer to a particular DTLS cipher suite and keying material set that is associated with a single host/port quartet. The same DTLS session can be used to establish the keying material for multiple DTLS associations. For consistency with other SIP/SDP usage, we use the term "connection" when what's being referred to is a multimedia stream that is not specifically DTLS.

3. Secure Channel

The UDPTL-over-DTLS media stream is negotiated using the SDP offer/answer mechanism [RFC3264]. See Section 4 for more details.

DTLS is used as specified in [RFC6347]. Once the DTLS handshake is successfully completed (in order to prevent facsimile data from being transmitted insecurely), the UDPTL packets MUST be transported in DTLS record layer "application_data" packets.

4. SDP Offerer/Answerer Procedures

4.1. General

An endpoint (i.e., both the offerer and the answerer) **MUST** create an SDP media description ("m=" line) for each UDPTL-over-DTLS media stream and **MUST** assign a UDP/TLS/UDPTL value (see Table 1) to the "proto" field of the "m=" line.

The procedures in this section apply to an "m=" line associated with a UDPTL-over-DTLS media stream.

In order to negotiate a UDPTL-over-DTLS media stream, the following SDP attributes are used:

- o The SDP attributes defined for UDPTL over UDP, as described in [ITU.T38.2010]; and
- o The SDP attributes, defined in [RFC4145] and [RFC4572], as described in this section.

The endpoint **MUST NOT** use the SDP "connection" attribute [RFC4145].

In order to negotiate the TLS roles for the UDPTL-over-DTLS transport connection, the endpoint **MUST** use the SDP "setup" attribute [RFC4145].

If the endpoint supports, and is willing to use, a cipher suite with an associated certificate, the endpoint **MUST** include an SDP "fingerprint" attribute [RFC4572]. The endpoint **MUST** support SHA-256 for generating and verifying the SDP "fingerprint" attribute value. The use of SHA-256 is preferred. UDPTL over DTLS, at a minimum, **MUST** support TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 and **MUST** support TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256. UDPTL over DTLS **MUST** prefer TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 and any other Perfect Forward Secrecy (PFS) cipher suites over non-PFS cipher suites. Implementations **SHOULD** disable TLS-level compression.

If a cipher suite with an associated certificate is selected during the DTLS handshake, the certificate received during the DTLS handshake **MUST** match the fingerprint received in the SDP "fingerprint" attribute. If the fingerprint does not match the hashed certificate, then the endpoint **MUST** tear down the media session immediately. Note that it is permissible to wait until the other side's fingerprint has been received before establishing the connection; however, this may have undesirable latency effects.

4.2. Generating the Initial Offer

The offerer **SHOULD** assign the SDP "setup" attribute with a value of "actpass", unless the offerer insists on being either the sender or receiver of the DTLS ClientHello message, in which case the offerer can use either a value of "active" (the offerer will be the sender of ClientHello) or "passive" (the offerer will be the receiver of ClientHello). The offerer **MUST NOT** assign an SDP "setup" attribute with a "holdconn" value.

If the offerer assigns the SDP "setup" attribute with a value of "actpass" or "passive", the offerer **MUST** be prepared to receive a DTLS ClientHello message before it receives the SDP answer.

4.3. Generating the Answer

If the answerer accepts the offered UDPTL-over-DTLS transport connection, in the associated SDP answer, the answerer **MUST** assign an SDP "setup" attribute with a value of either "active" or "passive", according to the procedures in [RFC4145]. The answerer **MUST NOT** assign an SDP "setup" attribute with a value of "holdconn".

If the answerer assigns an SDP "setup" attribute with a value of "active" value, the answerer **MUST** initiate a DTLS handshake by sending a DTLS ClientHello message on the negotiated media stream, towards the IP address and port of the offerer.

4.4. Offerer Processing of the Answer

When the offerer receives an SDP answer, if the offerer ends up being active it **MUST** initiate a DTLS handshake by sending a DTLS ClientHello message on the negotiated media stream, towards the IP address and port of the answerer.

4.5. Modifying the Session

Once an offer/answer exchange has been completed, either endpoint **MAY** send a new offer in order to modify the session. The endpoints can reuse the existing DTLS association if the key fingerprint values and transport parameters indicated by each endpoint are unchanged. Otherwise, following the rules for the initial offer/answer exchange, the endpoints can negotiate and create a new DTLS association and, once created, delete the previous DTLS association, following the same rules for the initial offer/answer exchange. Each endpoint needs to be prepared to receive data on both the new and old DTLS associations as long as both are alive.

5. Miscellaneous Considerations

5.1. Anonymous Calls

When making anonymous calls, a new self-signed certificate **SHOULD** be used for each call, and attributes inside the certificate **MUST NOT** contain information that allows either correlation or identification of the user making anonymous calls. This is particularly important for the "subjectAltName" and "commonName" attributes.

5.2. NAT Traversal

5.2.1. ICE Usage

When Interactive Connectivity Establishment (ICE) [RFC5245] is being used, the ICE connectivity checks are performed before the DTLS handshake begins. Note that if aggressive nomination mode is used, multiple candidate pairs may be marked valid before ICE finally converges on a single candidate pair. User Agents (UAs) **MUST** treat all ICE candidate pairs associated with a single component as part of the same DTLS association. Thus, there will be only one DTLS handshake even if there are multiple valid candidate pairs. Note that this may mean adjusting the endpoint IP addresses if the selected candidate pair shifts, just as if the DTLS packets were an ordinary media stream. In the case of an ICE restart, the DTLS handshake procedure is repeated, and a new DTLS association is created. Once the DTLS handshake is completed and the new DTLS association has been created, the previous DTLS association is deleted.

5.2.2. STUN Interaction

The UA **MUST** send the Session Traversal Utilities for NAT (STUN) packets [RFC5389] directly over UDP, not over DTLS.

The UA **MUST** support the following mechanism for demultiplexing packets arriving on the IP address and port associated with the DTLS association:

- o If the value of the first byte of the packet is 0 or 1, then the packet is STUN.
- o If the value of the first byte of the packet is between 20 and 63 (inclusive), the packet is DTLS.

5.3. Rekeying

During rekeying, packets protected by the previous set of keys can arrive after the DTLS handshake caused by rekeying has completed, because packets can be reordered on the wire. To compensate for this fact, receivers **MUST** maintain both sets of keys for some time in order to be able to decrypt and verify older packets. The duration of maintaining the previous set of keys after the finish of the DTLS handshake is out of the scope of this document.

5.4. Compatibility with UDPTL over UDP

If a user requires fax to be transported securely using UDPTL over DTLS, and if the remote user does not support UDPTL over DTLS, then a fax media stream cannot be established.

If a user prefers fax to be transported securely using UDPTL over DTLS but is willing to transport the fax insecurely in case the remote user does not support UDPTL over DTLS, then the SDP Capability Negotiation mechanism [RFC5939] can be used to offer both UDPTL over DTLS and UDPTL over UDP. Alternatively, if the remote user rejects an SDP offer for UDPTL over DTLS, a new SDP offer for a UDPTL-over-UDP media stream can be sent.

6. Security Considerations

Fax may be used to transmit a wide range of sensitive data, including personal, corporate, and governmental information. It is therefore critical to be able to protect against threats to the confidentiality and integrity of the transmitted data.

The mechanism in this document provides integrity and confidentiality protection for fax by specifying fax transport using UDPTL over DTLS [RFC6347].

DTLS media stream negotiated using SIP/SDP requires a mechanism to ensure that the certificate received via DTLS was issued by the remote party of the SIP session.

The standard DTLS strategy for authenticating the communicating parties is to give the server (and optionally the client) a PKIX [RFC5280] certificate. The client then verifies the certificate and checks that the name in the certificate matches the server's domain name. This works because there are a relatively small number of servers and the cost for issuing and deploying PKIX certificates can be justified. Issuing and deploying PKIX certificates to all clients is not realistic in most deployment scenarios.

The design described in this document is intended to leverage the integrity protection of the SIP signaling, while not requiring confidentiality. As long as each side of the connection can verify the integrity of the SDP received from the other side, then the DTLS handshake cannot be hijacked via a man-in-the-middle attack. This integrity protection is easily provided by the caller to the callee via the SIP Identity [RFC4474] mechanism. Other mechanisms, such as the S/MIME mechanism [RFC3261] or perhaps future mechanisms yet to be specified, could also serve this purpose.

While this mechanism can still be used without such integrity mechanisms, the security provided is limited to defense against passive attack by intermediaries. An active attack on the signaling plus an active attack on the media plane can allow an attacker to attack the connection (R-SIG-MEDIA in the notation of [RFC5479]).

7. IANA Considerations

This document updates the "Session Description Protocol (SDP) Parameters" registry as specified in Section 8.2.2 of [RFC4566]. Specifically, the values in Table 1 have been added to the SDP "proto" field registry.

Type	SDP Name	Reference
proto	UDP/TLS/UDPTL	[RFC7345]

Table 1: SDP "proto" Field Values

8. Acknowledgments

Special thanks to Peter Dawes, who provided comments on the initial draft version of the document, and to Paul E. Jones, James Rafferty, Albrecht Schwarz, Oscar Ohlsson, David Hanes, Adam Gensler, Ari Keranen, Flemming Andreasen, John Mattsson, and Marc Petit-Huguenin, who provided valuable feedback and input. Barry Leiba, Spencer Dawkins, Pete Resnick, Kathleen Moriarty, and Stephen Farrell provided valuable feedback during the IESG review. Thanks to Scott Brim for performing the Gen-ART review. Thanks to Alissa Cooper for her help as sponsoring Area Director.

9. References

9.1. Normative References

[ITU.T30.2005]

International Telecommunications Union, "Procedures for document facsimile transmission in the general switched telephone network", ITU-T Recommendation T.30, September 2005.

[ITU.T38.2010]

International Telecommunications Union, "Procedures for real-time Group 3 facsimile communication over IP networks", ITU-T Recommendation T.38, September 2010.

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

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC4145] Yon, D. and G. Camarillo, "TCP-Based Media Transport in the Session Description Protocol (SDP)", RFC 4145, September 2005.

[RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[RFC4572] Lennox, J., "Connection-Oriented Media Transport over the Transport Layer Security (TLS) Protocol in the Session Description Protocol (SDP)", RFC 4572, July 2006.

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

- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.
- [RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.
- [RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

9.2. Informative References

- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.
- [RFC5479] Wing, D., Fries, S., Tschofenig, H., and F. Audet, "Requirements and Analysis of Media Security Management Protocols", RFC 5479, April 2009.
- [RFC5939] Andreasen, F., "Session Description Protocol (SDP) Capability Negotiation", RFC 5939, September 2010.

Appendix A. Examples

A.1. General

Prior to establishing the session, both Alice and Bob generate self-signed certificates that are used for a single session or, more likely, reused for multiple sessions.

The SIP signaling from Alice to her proxy is transported over TLS to ensure an integrity-protected channel between Alice and her identity service. Alice's identity service asserts identity of Alice and protects the SIP message, e.g., using SIP Identity. Transport between proxies should also be protected, e.g., by use of TLS.

In order to simplify the flow, only one element is shown for Alice's and Bob's proxies.

For the sake of brevity and simplicity, only the mandatory SDP T.38 attributes are shown.

A.2. Basic Message Flow

Figure 3 shows an example message flow of session establishment for T.38 fax securely transported using UDPTL over DTLS.

In this example flow, Alice acts as the passive endpoint of the DTLS association, and Bob acts as the active endpoint of the DTLS association.

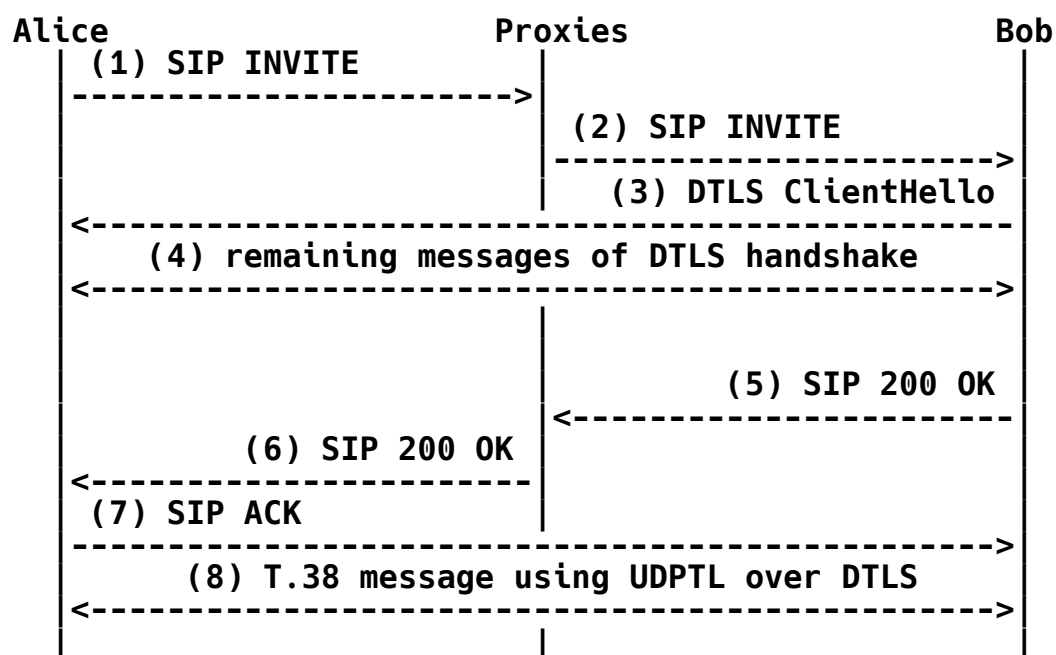


Figure 3: Basic Message Flow

Message (1):

Figure 4 shows the initial INVITE request sent by Alice to Alice's proxy. The initial INVITE request contains an SDP offer.

The "m=" line in the SDP offer indicates T.38 fax using UDPTL over DTLS.

The SDP "setup" attribute with a value of "actpass" in the SDP offer indicates that Alice has requested to be either the active or passive endpoint.

The SDP "fingerprint" attribute in the SDP offer contains the certificate fingerprint computed from Alice's self-signed certificate.

```
INVITE sip:bob@example.com SIP/2.0
To: <sip:bob@example.com>
From: "Alice"<sip:alice@example.com>;tag=843c7b0b
Via: SIP/2.0/TLS ua1.example.com;branch=z9hG4bK-0e53sadfkasldkfj
Contact: <sip:alice@ua1.example.com>
Call-ID: 6076913b1c39c212@REVMTEpG
CSeq: 1 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, UPDATE
Max-Forwards: 70
Content-Type: application/sdp
Content-Length: xxxx
Supported: from-change

v=0
o=- 1181923068 1181923196 IN IP4 ua1.example.com
s=-
c=IN IP4 ua1.example.com
t=0 0
m=image 6056 UDP/TLS/UDPTL t38
a=setup:actpass
a=fingerprint: SHA-1 \
    4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=T38FaxRateManagement:transferredTCF
```

Figure 4: Message (1)

Message (2):

Figure 5 shows the SIP INVITE request sent by Bob's proxy to Bob.

When received, Bob verifies the identity provided in the SIP INVITE request.

```
INVITE sip:bob@ua2.example.com SIP/2.0
To: <sip:bob@example.com>
From: "Alice"<sip:alice@example.com>;tag=843c7b0b
Via: SIP/2.0/TLS proxy.example.com;branch=z9hG4bK-0e53sadfkasldk
Via: SIP/2.0/TLS ua1.example.com;branch=z9hG4bK-0e53sadfkasldkfj
Record-Route: <sip:proxy.example.com;lr>
Contact: <sip:alice@ua1.example.com>
Call-ID: 6076913b1c39c212@REVMTEpG
CSeq: 1 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, UPDATE
Max-Forwards: 69
Content-Type: application/sdp
Content-Length: xxxx
Supported: from-change

v=0
o=- 1181923068 1181923196 IN IP4 ua1.example.com
s=-
c=IN IP4 ua1.example.com
t=0 0
m=image 6056 UDP/TLS/UDPTL t38
a=setup:actpass
a=fingerprint: SHA-1 \
    4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=T38FaxRateManagement:transferredTCF
```

Figure 5: Message (2)

Message (3):

Assuming that Alice's identity is valid, Bob sends a DTLS ClientHello directly to Alice.

Message (4):

Alice and Bob exchange further messages of DTLS handshake (HelloVerifyRequest, ClientHello, ServerHello, Certificate, ServerKeyExchange, CertificateRequest, ServerHelloDone, Certificate, ClientKeyExchange, CertificateVerify, ChangeCipherSpec, and Finished).

When Bob receives the certificate of Alice via DTLS, Bob checks whether the certificate fingerprint calculated from Alice's certificate received via DTLS matches the certificate fingerprint received in the a=fingerprint SDP attribute of Figure 5. In this message flow, the check is successful; thus, session setup continues.

Note that, unlike in this example, it is not necessary to wait for the DTLS handshake to finish before the SDP answer is sent. If Bob has sent the SIP 200 (OK) response and later detects that the certificate fingerprints do not match, he will terminate the session.

Message (5):

Figure 6 shows a SIP 200 (OK) response to the initial SIP INVITE request, sent by Bob to Bob's proxy. The SIP 200 (OK) response contains an SDP answer.

The "m=" line in the SDP answer indicates T.38 fax using UDPTL over DTLS.

The SDP "setup" attribute with a value of "active" in the SDP answer indicates that Bob has requested to be the active endpoint.

The SDP "fingerprint" attribute in the SDP answer contains the certificate fingerprint computed from Bob's self-signed certificate.

```
SIP/2.0 200 OK
To: <sip:bob@example.com>;tag=6418913922105372816
From: "Alice" <sip:alice@example.com>;tag=843c7b0b
Via: SIP/2.0/TLS proxy.example.com:5061;branch=z9hG4bK-0e53sadfkasldk
Via: SIP/2.0/TLS ua1.example.com;branch=z9hG4bK-0e53sadfkasldkfj
Record-Route: <sip:proxy.example.com;lr>
Call-ID: 6076913b1c39c212@REVMTEpG
CSeq: 1 INVITE
Contact: <sip:bob@ua2.example.com>
Content-Type: application/sdp
Content-Length: xxxx
Supported: from-change

v=0
o=- 8965454521 2105372818 IN IP4 ua2.example.com
s=-
c=IN IP4 ua2.example.com
t=0 0
m=image 12000 UDP/TLS/UDPTL t38
a=setup:active
a=fingerprint: SHA-1 \
    FF:FF:FF:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=T38FaxRateManagement:transferredTCF
```

Figure 6: Message (5)

Message (6):

Figure 7 shows a SIP 200 (OK) response to the initial SIP INVITE request, sent by Alice's proxy to Alice. Alice checks if the certificate fingerprint calculated from the Bob's certificate received via DTLS is the same as the certificate fingerprint received in the a=fingerprint SDP attribute of Figure 7. In this message flow, the check is successful; thus, the session setup continues.

```
SIP/2.0 200 OK
To: <sip:bob@example.com>;tag=6418913922105372816
From: "Alice" <sip:alice@example.com>;tag=843c7b0b
Via: SIP/2.0/TLS ua1.example.com;branch=z9hG4bK-0e53sadfkasldkfj
Record-Route: <sip:proxy.example.com;lr>
Call-ID: 6076913b1c39c212@REVMTEpG
CSeq: 1 INVITE
Contact: <sip:bob@ua2.example.com>
Content-Type: application/sdp
Content-Length: xxxx
Supported: from-change

v=0
o=- 8965454521 2105372818 IN IP4 ua2.example.com
s=-
c=IN IP4 ua2.example.com
t=0 0
m=image 12000 UDP/TLS/UDPTL t38
a=setup:active
a=fingerprint: SHA-1 \
  FF:FF:FF:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=T38FaxRateManagement:transferredTCF
```

Figure 7: Message (6)

Message (7):

Alice sends the SIP ACK request to Bob.

Message (8):

At this point, Bob and Alice can exchange T.38 fax securely transported using UDPTL over DTLS.

A.3. Message Flow of T.38 Fax Replacing Audio Media Stream in an Existing Audio-Only Session

Traditionally, most sessions with non-secure transport of T.38 fax, transported using UDPTL, are established by modifying an ongoing audio session into a fax session. Figure 8 shows an example message flow of modifying an existing audio session into a session with T.38 fax securely transported using UDPTL over DTLS.

In this example flow, Alice acts as the passive endpoint of the DTLS association, and Bob acts as the active endpoint of the DTLS association.

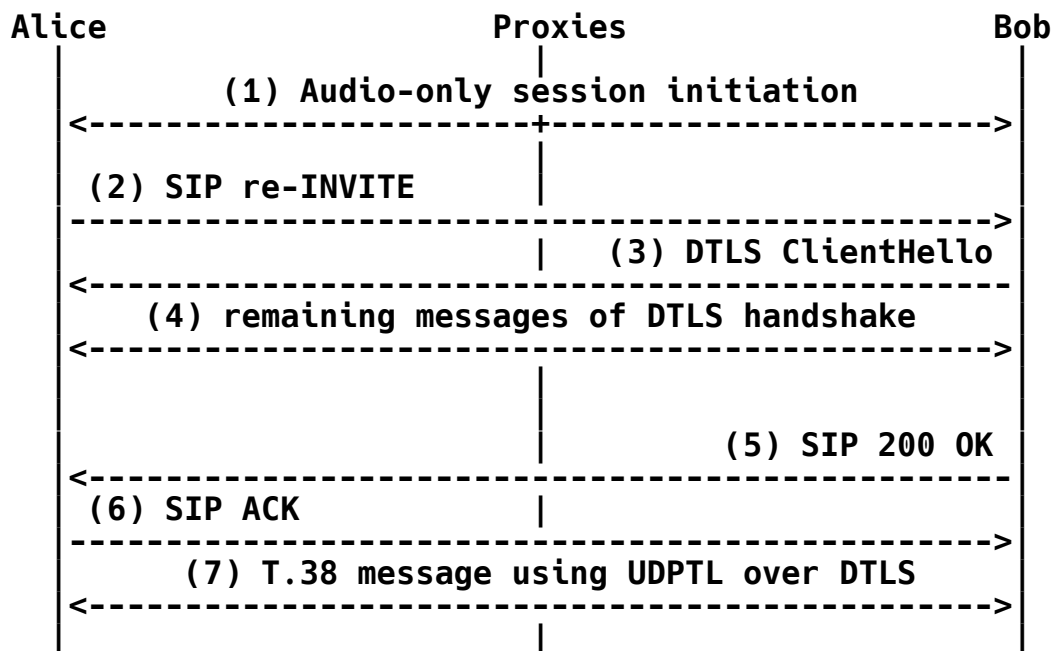


Figure 8: Message Flow of T.38 Fax Replacing Audio Media Stream in an Existing Audio-Only Session

Message (1):

Session establishment of audio-only session. The proxies decide not to record-route.

Message (2):

Alice sends SIP re-INVITE request. The SDP offer included in the SIP re-INVITE request is shown in Figure 9.

The first "m=" line in the SDP offer indicates audio media stream being removed. The second "m=" line in the SDP offer indicates T.38 fax using UDPTL over DTLS being added.

The SDP "setup" attribute with a value of "actpass" in the SDP offer indicates that Alice has requested to be either the active or passive endpoint.

The SDP "fingerprint" attribute in the SDP offer contains the certificate fingerprint computed from Alice's self-signed certificate.

```
v=0
o=- 2465353433 3524244442 IN IP4 ua1.example.com
s=-
c=IN IP4 ua1.example.com
t=0 0
m=audio 0 UDP/TLS/RTP/SAVP 0
m=image 46056 UDP/TLS/UDPTL t38
a=setup:actpass
a=fingerprint: SHA-1 \
  4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=T38FaxRateManagement:transferredTCF
```

Figure 9: SDP Offer of Message (2)

Message (3):

Bob sends a DTLS ClientHello directly to Alice.

Message (4):

Alice and Bob exchange further messages of DTLS handshake (HelloVerifyRequest, ClientHello, ServerHello, Certificate, ServerKeyExchange, CertificateRequest, ServerHelloDone, Certificate, ClientKeyExchange, CertificateVerify, ChangeCipherSpec, and Finished).

When Bob receives the certificate of Alice via DTLS, Bob checks whether the certificate fingerprint calculated from Alice's certificate received via DTLS matches the certificate fingerprint received in the SDP "fingerprint" attribute of Figure 9. In this message flow, the check is successful; thus, session setup continues.

Message (5):

Bob sends a SIP 200 (OK) response to the SIP re-INVITE request. The SIP 200 (OK) response contains an SDP answer shown in Figure 10.

The first "m=" line in the SDP offer indicates audio media stream being removed. The second "m=" line in the SDP answer indicates T.38 fax using UDPTL over DTLS being added.

The SDP "setup" attribute with a value of "active" in the SDP answer indicates that Bob has requested to be the active endpoint.

The SDP "fingerprint" attribute in the SDP answer contains the certificate fingerprint computed from Bob's self-signed certificate.

```
v=0
o=- 4423478999 5424222292 IN IP4 ua2.example.com
s=-
c=IN IP4 ua2.example.com
t=0 0
m=audio 0 UDP/TLS/RTP/SAVP 0
m=image 32000 UDP/TLS/UDPTL t38
a=setup:active
a=fingerprint: SHA-1 \
  FF:FF:FF:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=T38FaxRateManagement:transferredTCF
```

Figure 10: SDP Answer of Message (5)

Message (6):

Alice sends the SIP ACK request to Bob.

Message (7):

At this point, Bob and Alice can exchange T.38 fax securely transported using UDPTL over DTLS.

Authors' Addresses

Christer Holmberg
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

EMail: christer.holmberg@ericsson.com

Ivo Sedlacek
Ericsson
Sokolovska 79
Praha 18600
Czech Republic

EMail: ivo.sedlacek@ericsson.com

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

EMail: gsalguei@cisco.com