Internet Engineering Task Force
MSEC Working Group
INTERNET-DRAFT
EXPIRES: August 2004

Baugher (Cisco)
Carrara (Ericsson)

February 2004

The Use of TESLA in SRTP
<draft-ietf-msec-srtp-tesla-00.txt>

Status of this memo

Abstract

This memo describes the use of the Timed Efficient Stream loss-tolerant Authentication (TESLA) transform within the Secure Real-time Transport Protocol (SRTP), to provide data origin authentication for multicast and broadcast data streams.

TABLE OF CONTENTS

1. Introduction

Multicast and broadcast communication introduce some new security challenges compared to unicast communication. Many multicast and broadcast applications need "data origin authentication" (DOA), or "source authentication", in order to guarantee that a received message originated from a given source, and was not manipulated during the transmission. In unicast communication, a pairwise security association between one sender and one receiver can provide data origin authentication using symmetric-key cryptography (such as a message authentication code, MAC). When the communication is strictly pairwise, the sender and receiver agree upon a key that is known only to them.

In groups, however, a key is shared among more than two members, and this symmetric-key approach does not guarantee data origin

authentication. When there is a group security association [gkmarch] instead of a pairwise security association, any of the members can alter the packet and impersonate any other member. The MAC in this case only guarantees that the packet was not manipulated

by an attacker outside the group (and hence not in possession of the group key), and that the packet was sent by a source within the group.

Some applications cannot tolerate source ambiguity and must discern the true sender from any other group member. A common way to solve the problem is by use of asymmetric cryptography, such as digital signatures. This method, unfortunately, suffers from high overhead, in terms of time (to sign and verify) and bandwidth (to convey the signature in the packet).

Several schemes have been proposed to provide efficient data origin authentication in multicast and broadcast scenarios. The Timed Efficient Stream loss-tolerant Authentication (TESLA), is one such scheme.

This memo specifies TESLA authentication for SRTP. SRTP TESLA can provide data origin authentication to RTP applications that use group security associations (such as multicast RTP applications) so long as receivers abide by the TESLA security invariants [TESLA1, TESLA2].

1.1. Notational Conventions

The keywords "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].

This specification assumes the reader familiar with both SRTP and TESLA. Almost none of their details will be explained, and the reader can find them in their respective specifications [SRTP, TESLA1, TESLA2]. Also, this specification uses the same definitions as TESLA for common terms.

2. SRTP

The Secure Real-time Transport Protocol (SRTP) [SRTP] is a profile of RTP, which can provide confidentiality, message authentication, and replay protection to the RTP traffic and to the RTP control protocol, the Real-time Transport Control Protocol (RTCP).

SRTP is a framework that allows new security functions and new transforms to be added. SRTP currently does not define any mechanism to provide data origin authentication for group security associations. Fortunately, it is straightforward to add TESLA to the SRTP cryptographic framework.

The TESLA extension to SRTP is defined in this specification, which assumes that the reader is familiar with the SRTP specification [SRTP], its packet structure, and processing rules.

3. TESLA

TESLA provides delayed per-packet data authentication and is specified in two documents, an introductory overview [TESLA1] and a second specification that defines signaling and data packet parameters [TESLA2]. This specification assumes that the reader is familiar with these two documents.

In addition to its SRTP data-packet definition given here, TESLA needs an initial synchronization protocol and initial bootstrapping procedure. The synchronization protocol allows the sender and the receiver to compare their clocks and determine an upper bound of the difference. The synchronization protocol is outside the scope of this document.

TESLA also requires an initial bootstrapping procedure to exchange needed parameters and the initial commitment to the key chain [TESLA2]. For SRTP, it is assumed that the bootstrapping is performed out-of-band, possibly using the key management protocol that is exchanging the security parameters for SRTP, e.g. [GDOI, MIKEY]. Initial bootstrapping of TESLA is outside the scope of this document.

4. Usage of TESLA within SRTP

The present specification is an extension to the SRTP specification [SRTP] and describes the use of TESLA with only a single key chain, and the delayed-authentication TESLA elements of procedure [TESLA1, TESLA2].

4.1. The TESLA extension

TESLA is an OPTIONAL authentication algorithm for SRTP. When used, TESLA adds the fields showed in Figure 1 per-packet. The fields added by TESLA are called "TESLA authentication extensions" altogether, whereas "authentication tag" or "integrity protection tag" indicate the normal integrity protection tag when the SRTP master key is shared by more than two endpoints [SRTP].

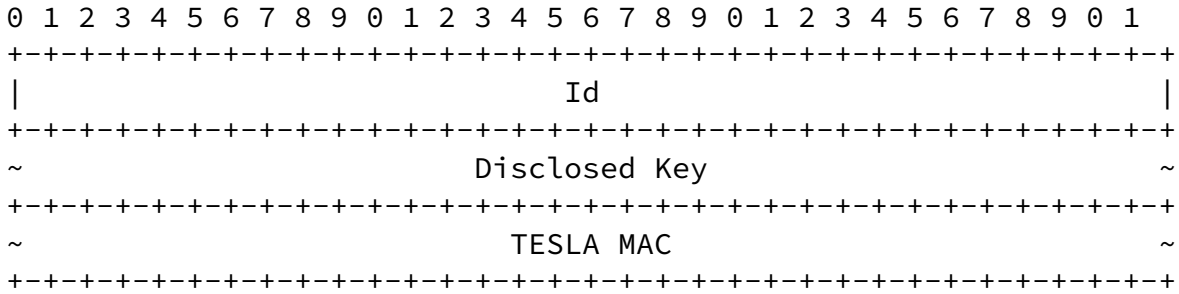


Figure 1: The "TESLA authentication extension".

Id: identifier of K_i, MANDATORY
The identifier of the key that was used to calculate the MAC present in the packet during interval i.

Disclosed Key: variable length, MANDATORY
The disclosed key, that can be used to authenticate previous packets from earlier time intervals, i.e. K_{i-d}.

TESLA MAC (Message Authentication Code): variable length, MANDATORY

The MAC computed using K'_i, which is disclosed in a subsequent packet. The MAC coverage is defined in Section 4.6.

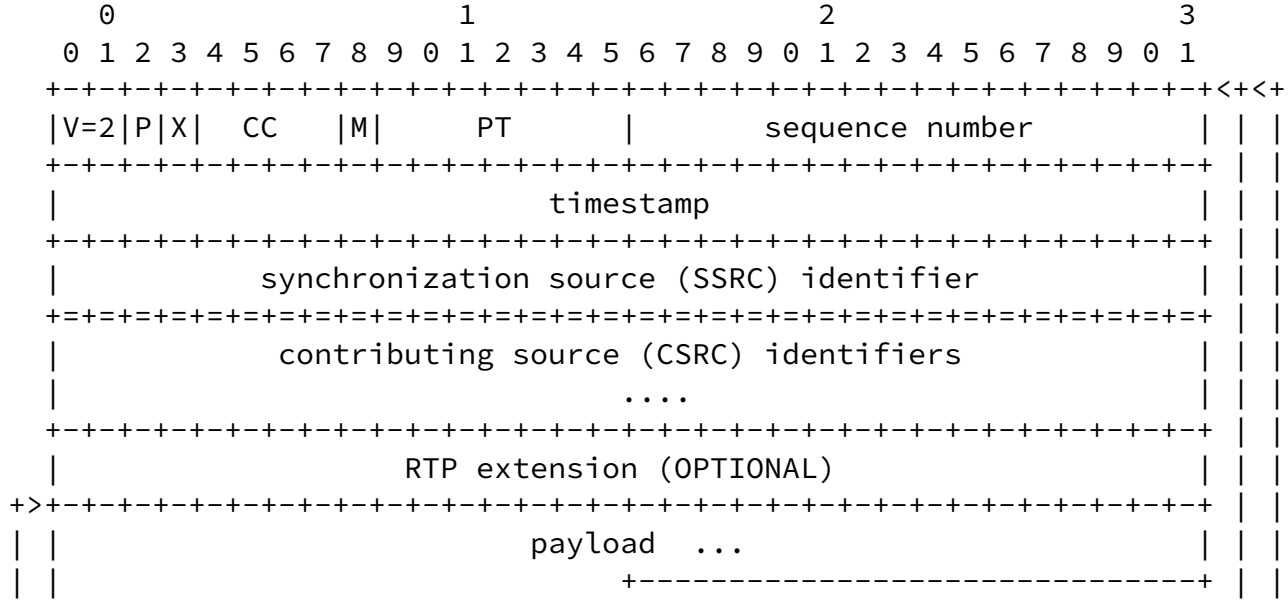
4.2. SRTP Packet Format

Figure 2 illustrates the format of the SRTP packet when TESLA is applied. When applied to RTP, the TESLA authentication extension SHALL be inserted before the (optional) SRTP MKI and (recommended) authentication tag.

As in SRTP, the "Encrypted Portion" of an SRTP packet consists of the encryption of the RTP payload (including RTP padding when present) of the equivalent RTP packet.

The "Authenticated Portion" of an SRTP packet consists of the RTP header, the Encrypted Portion of the SRTP packet, and the TESLA authentication extension. Note that the definition is extended from [SRTP] by the inclusion of the TESLA authentication extension.

The "TESLA Authenticated Portion" of an SRTP packet consists of the RTP header, the Encrypted Portion of the SRTP packet, the TESLA Id field, and the TESLA disclosed key.



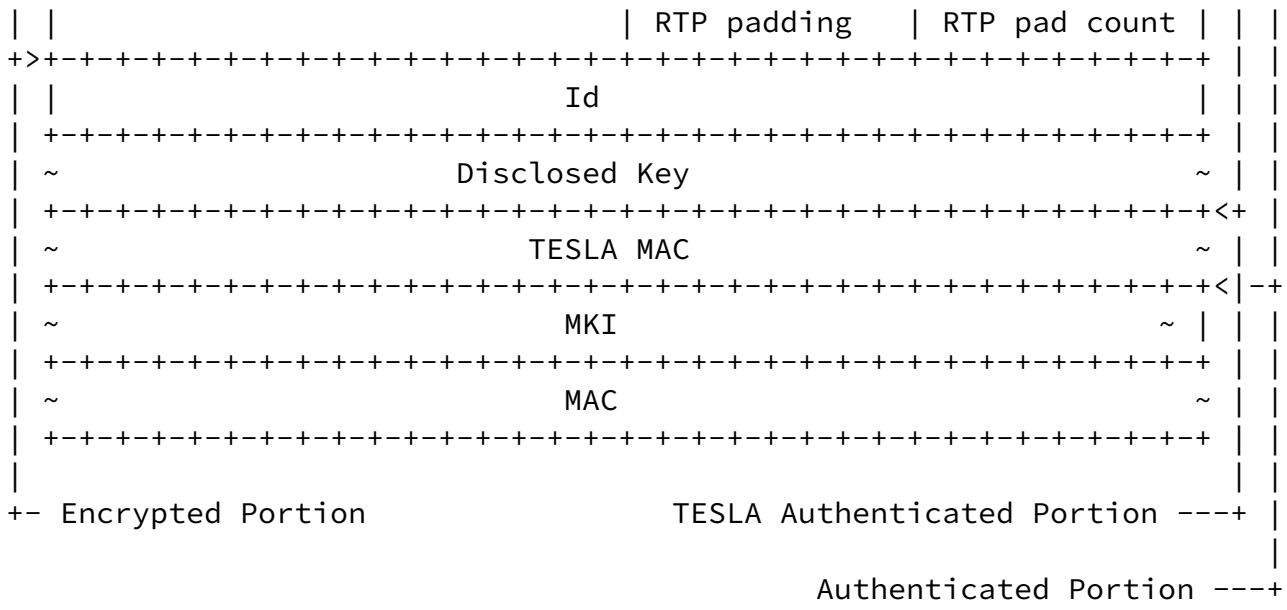


Figure 2. The format of the SRTP packet when TESLA is applied. Note that it is OPTIONAL to apply TESLA, i.e. the TESLA fields are OPTIONAL.

4.3. Extension of the SRTP Cryptographic Context

When TESLA is used, the definition of cryptographic context in Section 3.2 of SRTP SHALL include the following extensions:

1. an identifier for the PRF, f, implementing the one-way function F(x) in TESLA (to derive the keys in the chain), e.g. HMAC-SHA1,

2. a non-negative integer n_c, determining the length of the F output, i.e. the length of the keys in the chain (that is also the key disclosed in an SRTP packet),

3. an identifier for the PRF, f', implementing the one-way function F'(x) in TESLA (to derive the keys for the TESLA MAC, from the keys in the chain), e.g. HMAC-SHA1,

4. a non-negative integer n_f, determining the length of the

output of F', i.e. of the key for the TESLA MAC,

5. an identifier for the TESLA MAC, that accepts the output of F'(x) as its key,

6. a non-negative integer n_m, determining the length of the output of the TESLA MAC,

7. the identifier id_j of a specific time interval I_j,

8. an NTP timestamp TI_j describing the beginning of I_j,

9. an NTP timestamp T_int describing the interval duration,

10. the key-disclosure interval, d,

11. the id_n of the final key in the keychain, K_n,

12. the interval d_n of the last key chain element.

F(x) is used to compute a keychain of keys in SRTP TESLA, as defined in Section 6. Also according to TESLA, F'(x) computes a TESLA MAC key with inputs as defined in Section 6.

Note that the replay list is now containing indices of recently received packets that have been authenticated by TESLA. I.e. replay list updates MUST NOT be based on SRTP MAC.

These parameters are all "transform-specific" parameters. There is one transform-independent parameter that declares that SRTP message authentication is extended with TESLA DOA authentication. Section 6 of this document defines the default values for the transform-independent and transform-specific TESLA parameters.

4.4. SRTP Processing

The SRTP packet processing is described in Section 3.3 of the SRTP specification [SRTP]. The use of TESLA slightly changes the

processing, as the SRTP MAC is checked upon packet arrival for DoS prevention, but the current packet is not TESLA-authenticated. Each

packet is buffered until a subsequent packet discloses its TESLA key. The TESLA verification itself consists of some steps, such as tests of TESLA security invariants, that are described in Section 4 of [TESLA1]. The words "TESLA computation" and "TESLA verification" hereby imply all those steps, which are not all spelled out in the following.

4.4.1 Sender Processing

The sender processing is as described in Section 3.3 of [SRTP], up to step 5 included. After that the following process is followed:

6. When TESLA is applied, identify the key in the TESLA chain to be used in the current time interval, and the TESLA MAC key derived from it. Execute the TESLA computation to obtain the TESLA authentication extension for the current packet, by appending the key Id, the disclosed key of the chain for an earlier packet, and the TESLA MAC under the current key from the chain. This step uses the related TESLA parameters from the crypto context as for Step 4.

7. If the MKI indicator is set to one, append the MKI to the packet.

8. When TESLA is applied, compute the authentication tag as described in step 7 of Section 3.3 of the SRTP specification, but with coverage as defined in this specification (see Section 4.6).

9. If necessary, update the ROC (step 9 in Section 3.3 of [SRTP]).

4.4.2 Receiver Processing

The receiver processing is as described in Section 3.3 of [SRTP], up to step 4 included.

To authenticate and replay-protect the current packet, the processing is the following:

First check if the packet has been replayed (as for Section 3.3 of [SRTP]). If the packet is judged to be replayed, then the packet MUST be discarded, and the event SHOULD be logged.

Next, perform verification of the SRTP integrity protection tag (note, not the TESLA MAC), if present, using the rollover counter from the current packet, the authentication algorithm indicated in the cryptographic context, and the session authentication key. If

the verification is unsuccessful, the packet MUST be discarded from further processing and the event SHOULD be logged.

If the verification is successful, remove the MKI (if present) and authentication tag fields from the packet. The packet is buffered, awaiting disclosure of the TESLA key in a subsequent packet.

TESLA authentication is performed on a packet when the key is disclosed in a subsequent packet. When such key is disclosed, perform the TESLA verification of the packet using the rollover counter from the packet, the TESLA security parameters from the cryptographic context, and the disclosed key. If the verification is unsuccessful, the packet MUST be discarded from further processing and the event SHOULD be logged. If the TESLA verification is successful, remove the TESLA authentication extension from the packet.

To decrypt the current packet, the processing is the following:

Decrypt the Encrypted Portion of the packet, using the decryption algorithm indicated in the cryptographic context, the session encryption key and salt (if used) found in Step 4 with the index from Step 2.

Update the rollover counter and highest sequence number, s_l, in the cryptographic context, using the packet index estimated in Step 2. If replay protection is provided, also update the Replay List (i.e., the Replay List is updated after the TESLA authentication is successfully verified).

4.5. SRTCP Packet Format

Figure 3 illustrates the format of the SRTCP packet when TESLA is applied. The TESLA authentication extension SHALL be inserted before the MKI and authentication tag. Recall from [SRTP] that in SRTCP the MKI is OPTIONAL, while the E-bit, the SRTCP index, and the authentication tag are MANDATORY.

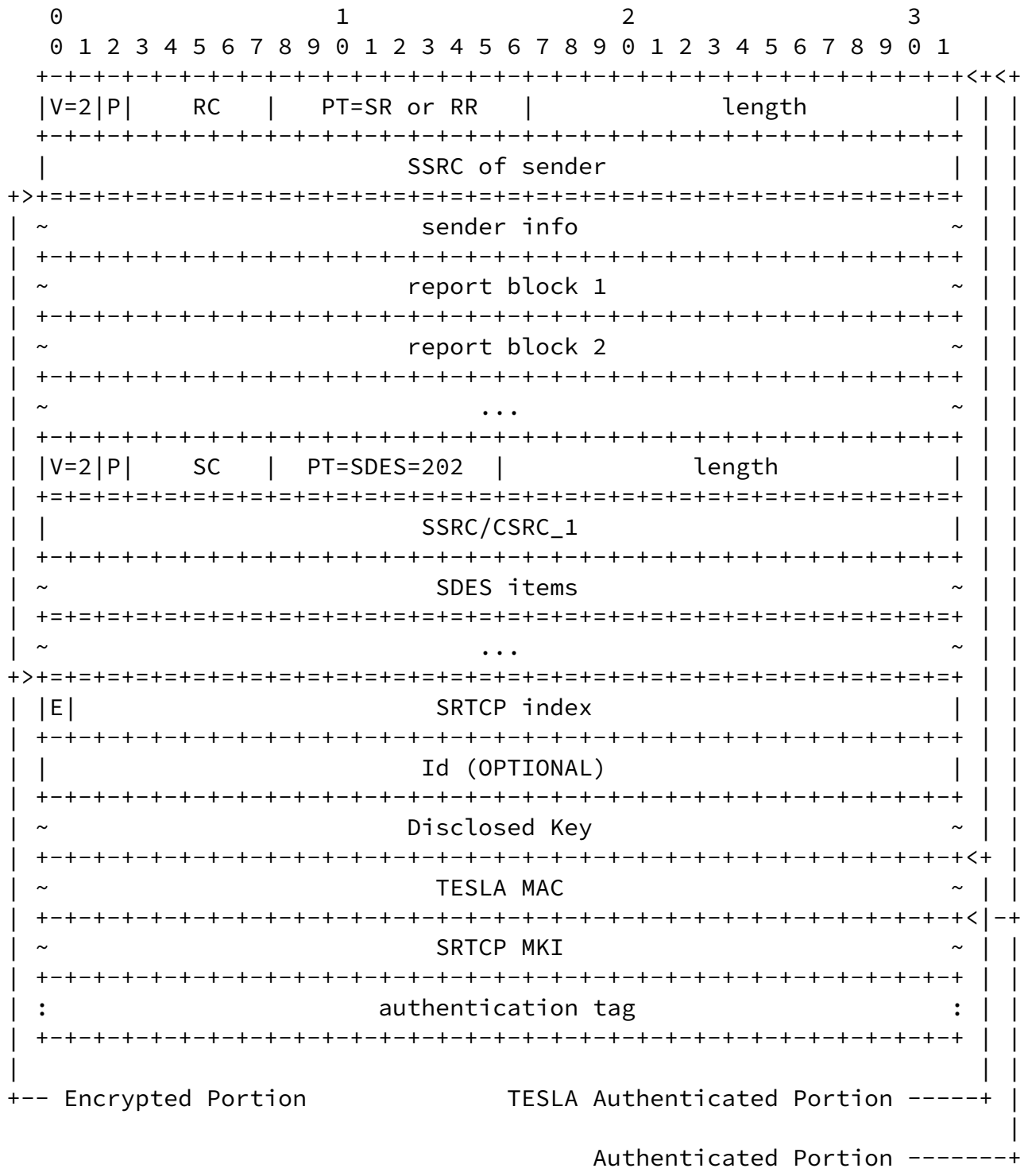

0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
|V=2|P| RC | PT=SR or RR | length |
SSRC of sender
sender info
report block 1
report block 2
...
|V=2|P| SC | PT=SDES=202 | length |
SSRC/CSRC_1
SDES items
...
|E| SRTCP index
Id (OPTIONAL)
Disclosed Key
TESLA MAC
SRTCP MKI
authentication tag
Encrypted Portion
TESLA Authenticated Portion
Authenticated Portion

Figure 3. The format of the SRTCP packet when TESLA is applied. Note that it is OPTIONAL to apply TESLA, i.e. the TESLA fields are OPTIONAL.

As in SRTP, the "Encrypted Portion" of an SRTCP packet consists of the encryption of the RTCP payload of the equivalent compound RTCP

packet, from the first RTCP packet, i.e., from the ninth (9) octet to the end of the compound packet.

The "Authenticated Portion" of an SRTCP packet consists of the entire equivalent (eventually compound) RTCP packet, the E flag, the SRTCP index (after any encryption has been applied to the payload), and the TESLA extension. Note that the definition is extended from [SRTP] by the inclusion of the TESLA authentication extension.

We define the "TESLA Authenticated Portion" of an SRTCP packet as consisting of the RTCP header (first 8 bytes), the Encrypted Portion of the SRTCP packet, the Id field, and the TESLA disclosed key.

Processing of an SRTCP packets is similar to the SRTP processing (Section 4.3), but there are SRTCP-specific changes described in Section 3.4 of the SRTP specification [SRTP] and in Section 4.6 of this memo.

4.6. TESLA MAC

Let M' denote packet data to be TESLA-authenticated. In the case of SRTP, M' SHALL consist of the SRTP TESLA Authenticated Portion (RTP header, SRTP Encrypted Portion, TESLA Id, and disclosed key) of the packet concatenated with the ROC of the same packet:

M' = ROC || TESLA Authenticated Portion.

In the case of SRTCP, M' SHALL consist of the SRTCP TESLA Authenticated Portion only (RTCP header, SRTCP Encrypted Portion, TESLA Id, and disclosed key).

The normal authentication tag SHALL be applied with the same

coverage as specified in [SRTP], i.e. Authenticated Portion || ROC for SRTP, and Authenticated Portion for SRTCP.

The pre-defined authentication transform in SRTP, HMAC-SHA1 [RFC2104], is also used to generate the TESLA MAC. For SRTP (respectively SRTCP), the HMAC SHALL be applied to the key in the TESLA chain corresponding to a particular time interval, and M' as specified above. The HMAC output SHALL then be truncated to the n_m left-most bits. Default values are in Section 6.2.

4.7. PRFs

TESLA requires two pseudo-random functions (PRFs), f and f', to implement

* one one-way function F(x) to derive the key chain, and
* one one-way function F'(x) to derive (from each key of the chain) the key that is actually used to calculate the TESLA MAC.

When TESLA is used within SRTP, the default choice of the two PRFs SHALL be HMAC-SHA1. Default values are in Section 6.2.

Other PRFs can be chosen, and their use SHALL follow the common guidelines in [SRTP] when adding new security parameter.

5. TESLA Bootstrapping

The extensions to the SRTP cryptographic context include a set of TESLA parameters that are listed in section 4.3 of this document. Key management procedures establish these parameters prior to the commencement of an SRTP session where TESLA authentication is used.

A critical factor for the security of TESLA is that the sender and receiver need to be loosely synchronized. TESLA assumes that the local internal clocks do not drift too much during the session. Use of TESLA in SRTP assumes that the time synchronization is guaranteed by out-of-band schemes, i.e. it is not in the scope of SRTP. The TESLA overview specification [TESLA2] describes some methods, which might be accomplished as part of SRTP key management. At least one

SRTP key management protocol, MIKEY, requires time synchronization [MIKEY].

6. SRTP TESLA Default parameters

Key management procedures establish SRTP TESLA operating parameters listed in section 4.3 of this document. The operating parameters appear in the SRTP cryptographic context and have the following default values. In the future, an Internet RFC MAY define alternative settings for SRTP TESLA that are different than those specified here. In particular, it should be noted that the settings defined in this memo can have a large impact on bandwidth, as it adds 38 bytes to each packet (when the field length values are the default ones) . For certain applications, this overhead may represent more than a 50% increase in packet size. Alternative settings might seek to reduce the number and length of various TESLA fields and outputs. No such optimizations are considered in this memo.

6.1 Transform-independent Parameter: SRTP MAC with TESLA MAC

Section 3.2.1 of the SRTP specification identifies "message authentication" as one of the transform-independent parameters. By default, this is HMAC-SHA1 for SRTP. With the addition of TESLA, SRTP message authentication becomes a compound parameter since it is necessary to identify two message authentication algorithms, one for the SRTP MAC and one for the TESLA MAC. Thus, the use or non-use of TESLA SHALL be indicated by the presence of a TESLA bit in the SRTP cryptographic context. When this bit is set, the SRTP implementation MUST inspect the TESLA transform-dependent parameters to determine the particular TESLA configuration.

It is RECOMMENDED that the SRTP MAC be truncated to four bytes since the SRTP MAC provides only group authentication and serves only as protection against DoS.

6.2 Transform-dependent Parameters for TESLA MAC

The default values for the security parameters are listed in the following. "OWF" denotes a one-way function.

Parameter	Mandatory-to-support	Default
TESLA KEYCHAIN OWF (F(x))	HMAC-SHA1	HMAC-SHA1
OUTPUT LENGTH	160	160
TESLA MAC KEY OWF (F'(F(x)))	HMAC-SHA1	HMAC-SHA1
OUTPUT LENGTH n_f	160	160
TESLA MAC	HMAC-SHA1	HMAC-SHA1
(TRUNCATED) OUTPUT LENGTH n_m	80	80
id_j		
TI_j		
T_int		
id_n		
d_n		

As shown above, TESLA implementations MUST support HMAC-SHA1 for the TESLA MAC, the MAC key generator, and the TESLA keychain generator one-way function. The TESLA keychain generator is recursively defined as follows.

K_i=HMAC_SHA1(K_{i+1},0), i=0..N-1

The TESLA MAC key generator is defined as follows.

K'_i=HMAC_SHA1(K_i,1)

The TESLA MAC uses a truncated output of ten bytes [RFC2104] and is defined as follows.

HMAC_SHA1(K'_i, M')

where M' is as specified in Section 4.6.

The TESLA interval parameters are id_j and id_n, both are 32 bits in length. The times associated with the intervals are TI_j, T_int, and d_n, which are 64-bit values in Network Time Protocol (NTP) format.

7. Security Considerations

Denial of Service (DoS) attacks when delayed authentication is used are discussed in [PCST]. TESLA requires receiver's buffering before authentication, therefore the receiver can suffer a denial of service attack due to a flood of bogus packets. To address this problem, the current specification REQUIRES the use of a four-byte SRTP MAC in addition to TESLA MAC. The shorter size of the SRTP MAC is here motivated by the fact that that MAC served purely for DoS prevention from attackers external to the group.

SRTP TESLA depends on the effective security of the systems that perform bootstrapping (time synchronization) and key management. These systems are external to SRTP and are not considered in this specification.

8. IANA Considerations

No IANA registration is required.

9. Acknowledgements

The authors would like to thanks Karl Norrman, Mats Näslund, and Ran Canetti, for their valuable help.

10. Author's Addresses

Questions and comments should be directed to the authors and msec@ietf.org:

Mark Baugher
Cisco Systems, Inc.
5510 SW Orchid Street Phone: +1 408-853-4418
Portland, OR 97219 USA Email: mbaugher@cisco.com

Elisabetta Carrara
Ericsson Research
SE-16480 Stockholm Phone: +46 8 50877040
Sweden EMail: elisabetta.carrara@ericsson.com

11. References

Normative

[PCST] Perrig, A., Canetti, R., Song, D., Tygar, D., "Efficient and Secure Source Authentication for Multicast", in Proc. of Network and Distributed System Security Symposium NDSS 2001, pp. 35-46, 2001.

[SRTP] Baugher, McGrew, Carrara, Naslund, Norrman, "The Secure Real-time Transport Protocol", July 2003, <draft-ietf-avt-srtp-09.txt>.

[TESLA1] Perrig, Canetti, Song, Tygar, Briscoe, "TESLA: Multicast Source Authentication Transform Introduction", October 2002, draft-ietf-msec-tesla-intro-01.txt.

[TESLA2] Perrig, Canetti, Whillock, "TESLA: Multicast Source Authentication Transform Specification", October 2002, draft-ietf-msec-tesla-spec-00.txt

Informative

[gkmarch] Baugher, Canetti, Dondeti, Lindholm, "MSEC Group Key Management Architecture", January 2003, <draft-ietf-msec-gkmarch-07.txt>.

[GDOI] Baugher, Weis, Hardjono, Harney, "The Group Domain of Interpretation", RFC 3547, July 2003.

[MESP] Baugher, Canetti, Cheng, Rohatgi, "MESP: A Multicast Framework for the IPsec ESP", March 2003, <draft-ietf-msec-mesp-01.txt>.

[MIKEY] Arkko, Carrara, Lindholm, Naslund, Norrman, "MIKEY: Multimedia Internet KEYing", December 2003, <draft-ietf-msec-mikey-08.txt>

Intellectual Property Right Considerations

Full Copyright Statement

This draft expires in August 2004.