TTC技術レポート
Technical Report

TR-1078

セキュアなリアルタイム転送プロトコル（SRTP）に関する技術報告書

Technical Report on the Secure Real-time Transport Protocol (SRTP)

第 1 版
2019 年 3 月 14 日制定

一般社団法人
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE

TTC Telecommunication Technology Committee

目次

＜参考＞

１．国際勧告等の関連

本技術レポートは、RFC3711 “The Secure Real-time Transport Protocol (SRTP)” を調査したものである。

２．上記国際勧告等に対する追加項目等

なし

３．改版の履歴

版数	制定日	改版内容
第 1 版	2019 年 3 月 14 日	制定

４．参考文献

[1] JF-IETF-RFC3711 : TTC 標準、＜簡略標準＞セキュアリアルタイムトランスポートプロトコル（SRTP）

[2] IETF RFC3711 : The Secure Real-time Transport Protocol (SRTP)

[3] JJ-22.01 : TTC 標準、企業 SIP 網間における相互接続インタフェース技術仕様

５．工業所有権

本標準に関わる「工業所有権等の実施の権利に係る確認書」の提出状況は、TTC ホームページでご覧になれます。

６．技術レポート作成部門

第 1 版 ： 企業ネットワーク専門委員会

I 本技術レポートの概要

1. 本報告書の策定の背景と位置づけについて

日本国内企業ネットワークでは、SIP 端末間のセキュアな RTP セッションの確立に関する標準はなかった。セキュアな RTP は、TTC 標準 JF-IETF-RFC3711 [1] として既に規定されており、本標準を企業網へ適用可能かを検討し、下記に示す企業網アーキテクチャにおいて適用可能と判断した。

なお、JF-IETF-RFC3711 [1] は IETF RFC3711 [2] 準拠の簡易標準であり、和訳されていないため、本技術レポートの II 章に IETF RFC3711 [2] の和訳を示す。

2. 企業網への適用

JF-IETF-RFC3711 は、JJ-22.01 [3] に規定されるフレームワーク標準の網接続アーキテクチャにおいて、SIP 端末間 (インタフェース C、E) の SRTP を用いたセキュアな RTP セッション確立のための推奨仕様を規定するものである。

また、インタフェース B を経由して接続される端末が、本報告書の範囲を超えた能力を保持することを妨げるものではない。但し、その場合においても本報告書に準拠する端末との接続性について考慮することが望ましい。

2.1 基本接続形態

JF-IETF-RFC3711 は、図.1 で示す企業 SIP 網相互接続モデルに規定されるインタフェース C、E に適用可能な管理された企業 SIP 網との接続インタフェースの条件を示す。本インタフェースの規定を遵守できるインタフェースを有する企業 SIP 網に関して、本報告書では“管理された企業 SIP 網”と呼ぶ。以下企業 SIP 網と表記する場合は、“管理された企業 SIP 網”であることを前提とする。

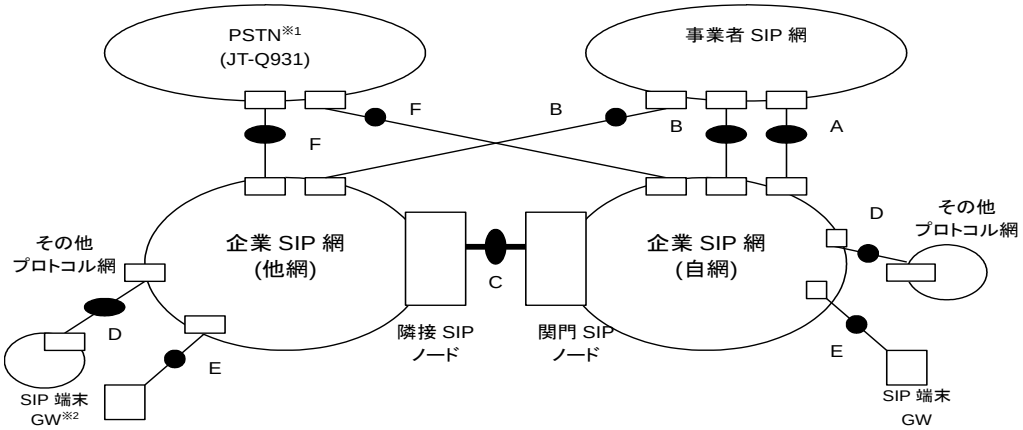


図 I.1 企業 SIP 網相互接続モデル

2.2 企業網への適用の目的と規定

JF-IETF-RFC3711 を、私設総合サービス網交換機 (PINX) 及び SIP 端末に実装するに際して、

接続条件に関わる規定の解釈を一意とすることで、実装可能とする。
SIP 網を介した SIP 端末同士の接続において、共通的に適用することを可能とする。
本規定の範囲を超えるまたは、厳密に本規定を遵守していない SIP UA との接続性にも最大限配慮したものとする。

ことを目的に以下の規定を適用する。

- SRTP が定義されている RFC3711 [1] に関する事項

2.3 JF-IETF-RFC3711 の企業網への適用内容

JF-IETF-RFC3711 に記載の下記を企業網へ適用する。

- SIP 端末同士で SRTP 通信を行う場合のフレームワーク
- SRTP の暗号化で用いる事前定義された暗号変換方式
- デフォルトで使用されるキー導出関数のテストデータ

II RFC3711 の和訳

セキュアなリアルタイム転送プロトコル (SRTP)

本文書のステータス

本文書はインターネット コミュニティのためのインターネット標準トラックプロトコルを規定し、改善にむけた協議や提案を要求する。このプロトコルの標準化状況やステータスについては、”インターネット公式プロトコル標準” (STD1) の最新版を参照されたい。本文書の配布は制限されない。

概要

この文書は、リアルタイム転送プロトコル (RTP) の 1 プロファイルであるセキュアなリアルタイム転送プロトコル (SRTP) について記載する。そして、それは RTP トラヒックおよび RTP の制御トラフィック (RTCP : リアルタイム転送制御プロトコル) に対して守秘性、メッセージ認証、再送防御を提供することがでる。

1. はじめに (RFC3711 セクション 1)

この文書は、リアルタイム転送プロトコル (RTP) の 1 プロファイルであるセキュアなリアルタイム転送プロトコル (SRTP) について記載する。そして、それは RTP トラヒックおよび RTP の制御トラヒック (RTCP : リアルタイム転送制御プロトコル) に対して守秘性、メッセージ認証、再送防御を提供することがでる。

SRTP は、RTP と RTCP ストリームの暗号化のフレームワークとメッセージ認証を提供します (セクション 3) 。SRTP は一式のデフォルト暗号変換を定めます (セクション 4 と 5) 、そして、それは将来新しい変換を許容します (セクション 6) 。適切なキー管理 (セクション 7 と 8) により、SRTP はユニキャストとマルチキャスト RTP アプリケーション (セクション 11) に対して安全化します (セクション 9) 。

SRTP は、高いスループットと低いパケット拡張を実現できる。SRTP は、異なる環境 (有線と無線ネットワークの混在) に対しても適切な保護手段であることがわかる。そのような特徴を得るために、次のようなデフォルト変換が記述されている。1 つ目は、暗号化のための付加的なストリーム暗号をベースに、2 つ目は、メッセージ認証のためのキーハッシュベースの機能と、3 つ目は、SRTP に対する RTP シーケンス番号に基づく配列/同期のための「暗黙の」インデックスであり、セキュアな RTCP (SRTCP) に対するインデックス番号も同様である。

1.1 用語

このドキュメント内でのキーワード「MUST」、「MUST NOT」、「REQUIRED」、「SHALL」、「SHALL NOT」、「SHOULD」、「SHOULD NOT」、「RECOMMENDED」、「MAY」と「OPTIONAL」は [RFC2119] に記載されているように、解釈される。用語は、以下の例外を除いて [RFC2828] に従う。簡単にするために、我々はランダムに、または、疑似ランダム的に発生する値を意味するために、文書を通して「ランダムな」という語を使用する。大量のランダムなビットを得るのが難しいかもしれないが、SRTP の安全性に対し、擬似ランダムは十分である [RFC1750] 。

慣例によって、採用している表現は、ネットワークバイトオーダである。すなわち、最も左のビット (オクテット中) が最上位である。XOR によって、我々はバイナリの文字列の 2 進法のビットごとの加算を意味し、そして、 ‖ は連結を意味する。言い換えれば、C = A‖B ならば、C で最も上位のビットは A のビットで、C の最も低位なビットは B のビットに等しい。十六進数の先頭には 0x を付ける。

「暗号化」の用語は、NULL アルゴリズム (それは、実際にはデータをクリアなままである) の使用も含む。

表記法が少し濫用されている場合には、「メッセージ認証」と「認証タグ」を同じように使う。ある状況、例えばグループコミュニケーション、においては、提供されるサービスが実は完全性保証だけで、データ発信元認証ではないことがある。

2. 目標と機能 (RFC3711 セクション 2)

SRTP のセキュリティ目標を確実にするためには :

* RTP と RTCP ペイロードの守秘性

* リプライされたパケットに対する保護とともに、全 RTP と RTCP パケットの完全性、。

これらのセキュリティサービスはオプションで互いから独立しているが、SRTCP 完全性保護は必須である (そまなければ、RTCP メッセージの悪意のある変更、または誤った変更は、RTP ストリームの処理を妨害する可能性がある) 。

他方、プロトコルの機能面の目標は以下です：

* 新しい暗号変換でアップグレードを可能にするフレームワーク、

* 低帯域幅コスト。すなわち RTP ヘッダ圧縮効率を維持するフレームワーク

さらに、あらかじめ定義された変換によって宣言される

* 低計算コスト、

* 小さなフットプリント (すなわちキー情報とリプライリストにかける少ないコードサイズとデータメモリ) 、

* 帯域幅を少なく抑える目標をサポートするための制限されたパケット拡張、

* RTP によって使われる下位のトランスポートレイヤ、ネットワークレイヤと物理レイヤからの独立性、特に、パケット損失と再配列に対する高い耐性。

これらのことは、有線と無線シナリオにおける RTP/RTCP に対して、SRTP が適切な保護スキームであることを保証する。

2.1 機能

上述の直接的な目標に加え、SRTP はさらに若干の追加機能を提供する。それらは、キー管理の負荷を軽減し、さらにセキュリティを高めるために導入された。それらは、以下を含む

SRTP ストリームと対応する SRTCP ストリームのために、一つの「マスターキー」が、守秘性と完全性保護のためのキーとなる情報を提供することができる。これは、それぞれのセキュリティプリミティブ、マスターキーからの安全な生成のための「セッションキー」を提供するキーの生成機能 (セクション 4.3 参照) により達成される。

さらに、キーの生成は、セッションキーの定期的なリフレッシュするように構成され、そして、これは、敵対者が利用可能な固定キーにより生成される暗号文の量を制限する。

「Salting キー」は、事前計算や、時間とメモリのトレードオフ攻撃からの防御に使用される [MF00] [BS00] 。

これらの機能の詳細な理論的な根拠は、セクション 7 に記載されている。

3. SRTP Framework (RFC3711 セクション 3)

RTP は、リアルタイムトランスポートプロトコル [RFC3550] です。SRTP を RTP の 1 つのプロファイルとして定義します。このプロフィールは、RTP Audio/Video プロファイル [RFC3551] への拡張です。はっきりと注意される場合を除き、SRTP セキュリティ機構の追加にとって、そのプロフィールのすべての面で応用できます。概念的に、我々は SRTP が「スタック内のこぶ」の実装であると考えます。それは、RTP アプリケーションとトランスポート層の間に位置します。送っている側では、SRTP は RTP パケットを横取りして、それ相当の SRTP パケットを送信します。また、受け入れ側では SRTP パケットを横取りして、上位のスタックにそれ相当の RTP パケットを渡します。

SRTP が RTP にするのと同様に、安全な RTCP (SRTCP) が、同じセキュリティサービスを RTCP に提供します。SRTCP メッセージ認証は必須で、それによって、認証の一部として RTCP フィールドを保護し、RTP 送信者にフィードバックを提供し、または、パケットのシーケンスカウンターを維持します。SRTCP は、セクション 3.4 で記述されます。

3.1 セキュアな RTP

SRTP パケットのフォーマットは、図 1 の中で例示されます。

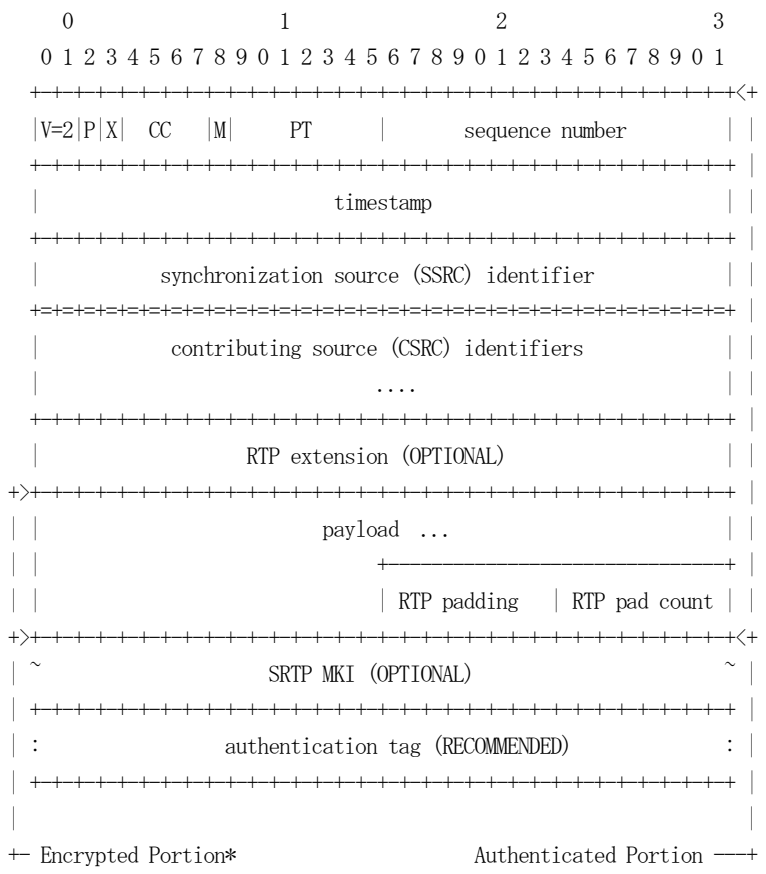


図 1.SRTP パケットのフォーマット。*暗号化部分は、セクション 4 で決められた変換による平文と同じサイズです。

SRTP パケットの「暗号化部分」は、相当の RTP パケットの RTP ペイロード (存在するとき、RTP パッドを含む) の暗号化から成ります。暗号化部分は、元の文と正確に同じサイズでもいいし、大きくなってもよい。図 1 は、RTP [RFC3550] のために、考えられるあらゆるパッドを含む RTP ペイロードを表します。

あらかじめ定義された暗号化の変換でパディングを使用していなければ、それにより、RTP と SRTP ペイロードサイズは、正確に合います。SRTP (セクション 6 参照) に加えられる新しい変換は、その結果、より大きなペイロードを生じるかもしれません。RTP はそれ自身のパッドフォーマット (図 1 の中で見られるように) を提供します。それにより、RTP ヘッダのパッド指標は、プレフィックスのないコードを使用したパッドに関して、ぎっしり詰まっている観点から見ると、受けるに値する。この RTP パディングは、パディングを要求する変換方法のデフォルトであるべきである。変換は他のパディング方法を定義してもよい。そのさいは、それらのパディングの量、フォーマットと処理を定義しなくてはならない。特にメッセージ認証が使われない [V02] とき、パディングを使用した暗号変換が巧妙な攻撃に弱いことに注意することは重要です。新しい暗号化変換の使用を決めるときは、それが使うパッドのセキュリティとの係わり合いについて注意深く考慮して、記述される必要がある。メッセージ認証コードはそれら自身のパッドを定めるので、このデフォルトは認証変換にはあてはまりません。

オプション MKI と推奨の認証タグは、RTP でない SRTP によって定義される唯一のフィールドです。8 ビット長とだけ決められています。

MKI (マスターキー識別子) : 可変長、オプション

MKI は定められて、キー管理によって決められ、信号化され、使われます。MKI は、特定のパケットを認証しておよび／または暗号化するセッションキーが引き出されたマスターキーを示します。MKI が SRTP 暗号文を示しているべきではないに注意すべきです。そして、それはセクション 3.2.3 によって示されています。MKI は、暗号文の範囲内で特定のマスターキーの変更、認証のためのキー管理により用いられてもよい。 (セクション 3.2.1)

認証タグ : 可変長、推奨

認証タグは、メッセージ認証データを転送するのに用いられます。SRTP パケットの認証部分は、SRTP パケットの暗号化部分の後の RTP ヘッダから成ります。このように、暗号化と認証が適用されるならば、送り側は暗号化を認証の前に、そして、受け側では逆に適用されます。認証タグは RTP ヘッダとペイロードの認証を提供します、そして、それはシーケンス番号を認証

することによって間接的にリプレイの保護を提供します。これがいかなる余分の保護も提供しなくて、MKI が完全には保護されていないことに注意すべきです。

3.2 SRTP 暗号文

各々の SRTP ストリームは、送り側と受け側に、暗号状態情報を維持することを要求します。この情報は、「暗号文」と呼ばれています。

SRTP は、2 種類のキーを使います : セッションキーとマスターキー。「セッションキー」、これは暗号の変換 (例えば暗号化またはメッセージ認証) において直接使われるキーを意味します。そして、「マスターキー」、これは、セッションキーが暗号によって安全な方法で引き出されるランダムなビット列 (キー管理プロトコルによって与えられる) を意味します。暗号文のマスターキーと他のパラメータは、SRTP に外部キー管理メカニズムで提供されます。セクション 8 参照。

3.2.1 変換独立パラメータ

変換独立パラメータは、使用されている特定の暗号化と認証とは独立に認証文中にある。SRTP のための暗号文の変換独立パラメータは以下から、成ります。

* 32 ビットアンサインロールオーバーカウンター (ROC) 、これは、16 ビット RTP シーケンス番号が 65,535 を通過してゼロにリセットされた回数を記録する。シーケンス番号 (SEQ) (それを SRTP は RTP パケットヘッダから抽出します) と違って、セクション 3.3.1 で記述されているとおり、ROC は SRTP によって維持されます。

  SRTP パケットのインデックスを ROC と RTP シーケンス番号から 48 ビット量に相当すると決める。

  $$i = 2^{16} * ROC + SEQ.$$

* 受け側だけのために、16 ビットシーケンス s_l があります。それは、受信した RTP シーケンス番号の最高値 (その取扱いについてはセクション 3.3.1 参照) と考えられることができます。また、それは、メッセージ認証が推奨であるから、認証されるべきである。

* 暗号化アルゴリズム (すなわち暗号と作動のモード) の識別子。

* メッセージ認証アルゴリズムのための識別子。

* リプレイリスト。それは、受け側 (認証とリプレイ保護が提供されるとき) だけによって維持される。また、最近受信して、認証された SRTP パケットのインデックスを含んでいる。

* MKI 識別子 (0/1) 。それは、SRTP や SRTCP パケット内に MKI が存在するかどうかを示している。

* もし、MKI 識別子が 1 に設定されるならば、MKI フィールドの長さ (オクテットで) と、 (送り側のための) 現在の MKI の有効な値の MKI フィールドと現在活発な MKI (MKI 識別子の値と長さは、文のライフタイムうちは、固定され続けなければならない) 。

* マスターキー、それは、ランダムで、秘密にしなければならない。

* 各々のマスターキーにとって、そのマスターキー (保安に必要なもの、Sections 3.3.1 と 9 を見てください) で処理された (送られた) SRTP パケットの数のカウンタがあります。

* 非負整数 n_e (そして、n_a) 、それらは、暗号化とメッセージ認証のためにセッションキーの長さを決める。

さらに、各々のマスターキーに対して、SRTP ストリームは以下の関連した値を使用する。

* マスターsalt、それは、セッションキーのキー生成で使われる。使用されるとき、この値は、短ダ無でなければならないが、公であってもよい。マスターsalt の使用は強く推奨であり、セクション 9.2 参照。「NULL」salt は、00...0 とみなされます。

* {1, 2, 4, ... , 2^24} にセットされる整数、「key_derivation_rate」、これは価値がゼロとみなされる。2 の累乗である制約はセッションキーの生成実装を単純化する。セクション 4.3 参照。

* MKI 値

* <From, To> 値、これは、マスターキーのためにライフタイムを指定します。それらは 2 つの 48 ビットインデックス値として表現され、その範囲内 (範囲の最終ポイントを含む) であれば、マスターキーは有効である。<From, To> の証に関しては、セクション 8.1.1 参照。<From, To> のどちらか一報が MKI であり、マスターキーが 1 対 1 において、SRTP セッションキー (それは <From, To> 範囲により決められた) に相当すると推測できる。

SRTCP はデフォルトで SRTP と暗号文を共有すべきですが、以下を除外してください：

* RTCP インデックスは、各々の SRTCP パケットではっきりと転送されるように、ロールオーバーカウンターと s_l の値は、維持される必要がありません。

* 別々のリプレイリストが維持されます (リプレイ保護が提供されるとき) 、

* SRTCP はそのマスターキーのために別々にカウンタを維持する。 (たとえマスターキーが SRTP のためにそれと同じであるとしても、下記参照) 。そのキーで処理された SRTCP パケットの数のカウントを維持する手段としてである。

特に注意すべきは、マスターキーが SRTP と対応する SRTCP で共有するかもしれない。これは、あらかじめ定義された変換 (キーの生成を含む) が使用されるが、セッションキーはそのように分けてはいけないときなどである。

そのうえ、特定の RTP セッション (それらの同期ソース (RTP ヘッダの一部である SSRCs) によって確認される) 以内のいくつかの SRTP ストリームが大部分の暗号文パラメータ (できればマスターでセッションキーを含む) を共有するケース (セクション 8 と 9.1 参照) が、あることができます。そのような場合、ちょうど、通常の SRTP/SRTCP パラメータは共有して、リプライリストは別々であり、それぞれのストリーム (SSRC) のパケットカウンターはまだ維持されるべきだ。また、別々の SRTP インデックスが、そのときは維持されなくてはならない。

パラメータ、あらかじめ定義された変換と上記のパラメータ (そして、他の SRTP パラメータ) のデフォルト値の概要は、セクション 5 と 8.2 で見つかることができます。

3.2.2 変換依存パラメータ

すべての暗号化、認証/完全性とキー生成パラメータは、変換部 (セクション 4) で定められます。そのようなパラメータの典型的例は暗号のブロックサイズ、セッションキー、Initialization Vector (IV) 構造のためのデータ、などです。そして、将来の SRTP 変換仕様は、あるとしても、その変換のための追加となる暗号文のパラメータをリストするためにセクションを含みます。

3.2.3 SRTP パケットを暗号文へのマッピング

各々の参加者のための RTP セッションが一対の相手の転送先アドレス (RTP と RTCP のための 1 つのネットワークアドレスとポートの対) によって定義される [RFC3550] 、また、マルチメディアのセッションが RTP セッションのコレクションと定義されることを思い出してください。たとえば、特定のマルチメディアのセッションは、音声 RTP セッション、ビデオ RTP セッションとテキスト RTP セッションを含むことができました。

暗号文は、三つの文識別子によってユニークに確認されるべきです：

文 id = <SSRC, 相手のネットワークアドレス, 相手の転送ポート番号>

ここで、相手のネットワークアドレスと相手のトランスポートポートは、SRTP パケットのなかのものです。この情報で示されるとき、セクション 3.2 で記述されているとおり、キー管理が情報で文を返すと想定されます。

上記したように、SRTP と SRTCP は、暗号文中の大半のデフォルトでパラメータを共有します。このように、実際には SRTCP ストリームの暗号文パラメータを検索することは、対応する SRTP 暗号文に、関係するかもしれません。RTCP ポートが RTP ポートだけから直接演繹できないかもしれないのと同じように、それはそのような関連を保証するために実装されていきます。あるいは、キー管理は別々の SRTP-と SRTCP-文を提供するほうを選ぶかもしれません。そして、一般のパラメータ (例えばマスターキー) は複製します。とても希望されるならば、後者のアプローチはそれからまた、SRTP と SRTCP が、例えば、異なった変換を使うのを可能にします。複数の SRTP ストリームが、1 つの RTP セッションの一部を作って、キーと他のパラメータを共有するとき、類似した考慮すべき問題は起こります。

有効な前後関係が特定の前後関係識別子と一致しているパケットのために見つからないならば、そのパケット MUST は捨てられます。

3.3 SRTP パケット処理

以下が SRTP に適用される。 SRTCP についてはセクション 3.4 に記載する。

暗号文の初期化がキー管理を通して起こったならば、送り側は SRTP パケットを造るために以下をすべきです：

1. セクション 3.2.3 中で記述されるように、どの暗号文を使うべきかについて決定してください。

2. セクション 3.3.1 で記述されるように、ロールオーバーカウンターを使用している SRTP パケットのインデックス、暗号文で最も高いシーケンス番号と RTP パケットのシーケンス番号を決定してください。

3. マスターキーと、マスターsalt を決定してください。この前のステップで決定されるインデックス、または暗号化文内の現在の MKI を使用してなされます。セクション 8.1 参照。

4. セッションキーとセッション salt (それらが変換によって使われるならば) を決定してください。それはセクション 4.3 で記述されていて、インデックスで暗号文内の、ステップ 2 と 3 において決定されたマスターキー、マスターsalt、key_derivation_rate とセッションキー長を使用します。

5. パケットの暗号部分を生成ために、RTP ペイロードを暗号化してください。 (定義済み暗号のために、セクション 4.1 参照) このステップは、ステップ 2 で見つかるインデックスと共にステップ 4 で見つかる暗号文、セッション暗号化キーとセッション salt (使われるならば) で示される暗号化アルゴリズムを使用します。

6. MKI 指標が 1 に設定されるならば、パケットに MKI を追加してください。

7. メッセージ認証のために、パケットの認証部分のための認証タグを計算することセクション 4.2 で記述されています。このステップは現在のロールオーバーカウンター、認証文が示している認証アルゴリズム、ステップ 4 で見つけたセッション認証キーを使用します。パケットに認証タグを追加してください。

8. 必要に応じて、ステップ 2 で決定されるパケット・インデックスを用いて、セクション 3.3.1 の場合のように ROC を更新してください。

SRTP パケットを認証して、解読するために、受け側は、以下をすべきです：

セクション 3.2.3 中で記述される、どの暗号文を使うべきかについて決定してください。

SRTP パケットのインデックスを得るために、セクション 3.3.1 のアルゴリズムを走らせてください。セクション 3.3.1 で記述されているように、アルゴリズムは SRTP パケット内のシーケンス番号で暗号文のロールオーバーカウンターと最も高いシーケンス番号を使います。

マスターキーとマスターソルトを決定してください。文中の MKI 識別子が 1 に設定されるならば、SRTP パケットの MKI を使ってください、さもなければ、セクション 8.1 によるように、前のステップからインデックスを使用してください。

インデックスを使用して、暗号文内の、セクション 4.3 で記述されているセッションキー、セッション salt (変換によって使われるならば) を決定してください。それは、マスターキー、マスターsalt、key_derivation_rate とセッションキー長を使用します。

メッセージ認証とリプレイ保護のために、最初に、ステップ 2 で決定されたリプライリストとインデックスを用いて、パケットがリプレイされたか (セクション 3.3.2) どうか調べてください。パケットがリプレイされると判断されたならば、それからパケットは捨てられなければなりません。そして、イベントに記録されるべきです。

次に、ステップ 2 からのロールオーバーカウンター、暗号文で示される認証アルゴリズムとステップ 4 からのセッション認証キーを用いて、認証タグの確認を実行してください。結果が「認証失敗」セクション 4.2 参照）であるならば、パケットは、更なる処理を実行してはならなく、イベントを記録すべきです。

ステップ 2 からのインデックスを使用して、ステップ 4 で見つかる暗号文、セッション暗号化キーと salt (使われるならば) で示される解読アルゴリズムを用いて、パケット (定義済み暗号、セクション 4.1 参照) の暗号部分を解読してください。

ステップ 2 で推定されるパケット・インデックスを用いて、セクション 3.3.1 の場合のように暗号文のロールオーバーカウンターと最も高いシーケンス番号 (s_l) を更新してください。リプレイ保護が提供されるならば、また、セクション 3.3.2 で記述される、リプレイリストを更新してください。

存在するとき、パケットから MKI と認証タグフィールドを取り出してください。

3.3.1 パケット・インデックスの決定、それと ROC, s_l のアップデート

SRTP 実装は、配列のために「陰の」パケット・インデックスを使用します、すなわち、インデックスの全てが SRTP パケットの中にはっきりともたらされるというわけではありません。あらかじめ定義された変換のために、そして、キー生成 (セクション 4.3) のために、インデックス i が、リプレイ保護 (セクション 3.3.2) 、暗号化 (セクション 4.1) 、メッセージ認証 (セクション 4.2) において使われます。

セッションが始まるとき、送り側はロールオーバーカウンター (ROC) をゼロに設定しなければなりません。RTP シーケンス番号 (SEQ) がモジュロ 2^16 を超えるたびに、送り側は 1 つ ROC を増加させなければなりません。 (下記のセキュリティ面を参照) 送り側のパケット・インデックスは以下のように定義される

i = 2^16 * ROC + SEQ.

受信側の実装はパケットの正しいインデックスを決定するために RTP シーケンス番号を使います。そして、それはすべての SRTP パケットのシーケンスのパケットの位置です。ロールオーバーカウンターの適正使用のための強いアプローチは、はっきりしていることをその取扱いと使用を要求します。特に、乱れていて、シーケンス番号約 2^16 または 0 による RTP パケットは、きちんと取り扱われなければなりません。

インデックス予想は、受け側の地元で維持された ROC と s_l 価値に基づきます。セッションのセットアップで、ROC は、ゼロにセットされなければなりません。進行中のセッションに加わっている受け側は、帯域外周波数信号方式 (例えばキー管理シグナリング) を使用している現在の ROC 値を与えられなければなりません。さらにまた、受け側は、最初の観察された SRTP パケット (初期値がキー管理のようなバンドシグナリングからによって提供されない限り) の RTP シーケンス数 (SEQ) に、s_l を初期化します。

連続的な SRTP パケットにおいて、受け側はインデックスを以下として推定するべきです。

i = 2^16 * v + SEQ,

v がセット {ROC-1, ROC, ROC+1} (モジュロ 2^32) から選ばれる場合、そのようなその i は値 2^16 * ROC + s_l (疑似コードについては付録 A 参照) に最も近いです (モジュロ 2^48) 。

パケットが処理されて、認証された (セッションの間 SRTP パケットのために可能にされるとき) あと、受け側は以下の通りに条件つきでその s_l と ROC 変数を更新するために v を使わなければならない。v = (ROC-1) モッズ 2^32 ならば、s_l または ROC のアップデートはありません。v = ROC ならば、SEQ が現在の s_l より大きい場合に限り、s_l は SEQ にセットされます;変化が、ROC にありません。v = (ROC+1) モッズ 2^32 ならば、s_l は SEQ にセットされます、そして、ROC は v にセットされます。

キー再交換の後、 (新しいマスターキーに変わる) 、ロールオーバーカウンターは、常に値のそのシーケンスを維持します、すなわち、ゼロにリセットされなければなりません。

ロールオーバーカウンターが長さ 32 ビットである、そして、シーケンス番号が長さ 16 ビットで、同じキーで守られることができる所定の SRTP ストリームに属しているパケットの最大数は前もって定義の変換を用いた 2^48 です。その数の SRTP パケットが所定の (マスターまたはセッション) キーで送られたあと、送り側はそのキーでどんなより多くのパケットでも送ってはいけない。(SRTCP にとっても類似した制限が存在する。そして、それは実際にはより制限的かもしれないと、セクション 9.2 参照。) この制限は、暗号キーが変わる前に、通ることができるトラヒックの量の上で上界を提供することによって、セキュリティ利益を実施します。トラヒックのこの量の前に、キー変換が引き起こされなければなりません (セクション 8.1 参照) 、そして、もっと以前に、例えばメディアへのさらなる保安とアクセス制御のために、引き起こされるかもしれません。ゼロ以外の key_derivation_rate (セクション 4.3 参照) による繰り返されているキー生成は、また、より強い保安をするが、上記の絶対の最大値を変えません。

受け側で、s_l と ROC を更新することに警告があります : メッセージ認証も存在しないならば、s_l も ROC アップデートの初期化は完全に強くされることができます。リオーダとパケットの損失があまり大きくない限り、受け側の「陰のインデックス」のアプローチはあらかじめ定義された変換が働きます、そして、ビットエラーは不運な方向で起こりません。特に、2^15 パケットは失われる必要があります、あるいは、同期が失われる前に、シーケンスから 2^15 パケットである必要があります。そのような思い切った損失またはリオーダは、RTP アプリケーションそのものを崩壊させそうです。

インデックス予想と ROC アップデートのためのアルゴリズムは、実装の問題で、最初のシーケンス番号 (ランダムに、RTP によって選ばれる) が前もって知られていなくて (キー管理プロトコルで送られない) 、モジュロ 2^16 を包むために近いかもしれないとき同期が、例えば、失われそうなとき、環境 (例えばパケット損失率) とケースを考慮に入れなければなりません。

上で与えられるものより綿密でより強力なスキームは RTP の自身の「ロールオーバーカウンター」の取扱いです、付録 A.1 参照 [RFC3550] 。

3.3.2 リプレイ保護

完全性保護が存在するとき、安全なリプレイ保護は可能なだけです。RTP と RTCP の両方ために、完全性保護だけでは反射攻撃に対する保安を保証することができないように、リプレイ保護を使うことは、推奨です。

それが相手によって保存されるとき、パケットは「リプレイされて」、それから、ネットワークに再送されます。メッセージ認証が提供されるとき、SRTP はリプライリストを通してそのような攻撃から保護します。各々の SRTP の受け側はリプライリストを維持します。そして、それは概念的に、受け取られて、認証されたパケットの全てのインデックスを含みます。実際には、リストは「スライドするウィンドウ」アプローチを使うことができます、そのため、保管の一定の量はリプレイ保護にとって十分です。SRTP-WINDOW-SIZE より多くの文のパケット・インデックスより遅れているパケット・インデックスは、SRTP-WINDOW-SIZE がレシーバー側であるばあい、実装に依存するパラメータを受け取られたと推測できます。また、少なくとも 64 でなければならず、より高い価値にセットされてもよい。

受け側は、入って来るパケットのインデックスをリプレイリストとウィンドウと照合します。ウィンドウの前のインデックスであるか、または、ウィンドウ内であるがまだ受け取られていない場合のみ、受け入れられるべきです。

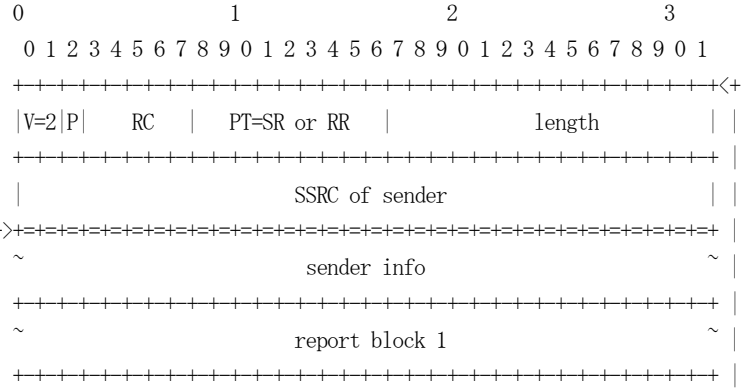
パケットが認証された (必要に応じて、ウィンドウは前に最初に動かされます) あと、リプレイリストは新しいインデックスで更新されるべきです。

IP のためにセキュリティアーキテクチャ [RFC2401] で記述されているように、受け取られたパケットを表現するためにビットマップを用いることにより、リプライリストは効率的に実装されることができる。

3.4 セキュアな RTCP

Secure RTCP は Secure RTP の定義に従属する。SRTP は新規に 3 つの必須フィールド (SRTCP index、encrypt-flag、認証タグ) と 1 つのオプションフィールド (MKI) を RTCP パケットの定義に追加する。同等の SRTCP パケットを形成するためには、3 つの必須フィールドが RTCP パケットに追加されなければならない [MUST] 。追加されたフィールドは、他のプロファイル固有の内線番号に従属する。

[RFC3550] のセクション 6.1 によると、合成パケット (compound packets) には、必須で要求されるパケットフォーマットが存在する [REQUIRED] 。SRTCP は、前半部分が sender report や reciever report でなければならないという意味では、その要求に従ったパケットである必要がある [MUST] 。しかし、そのセクションで規定されている RTCP 暗号化プレフィックス (ランダムな 32bit の値) は、[RFC3550] で規定される暗号化方法にのみ適用されることと、SRTP ではその暗号化メカニズムは不要であるということから、そのままの状態で使用してはならない [MUST NOT] 。



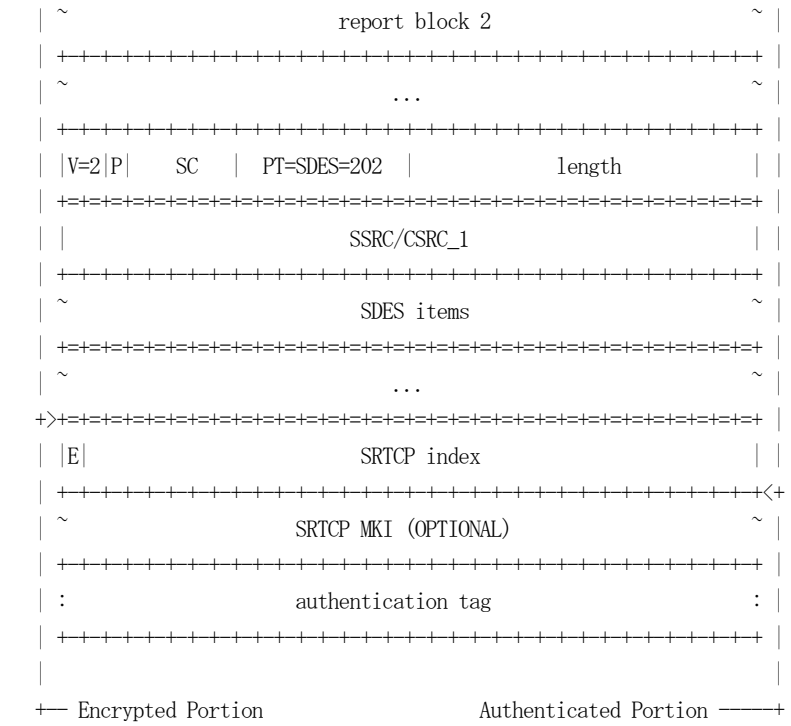


図 2. Sender Report と SDES パケットの場合の、RTCP 合成パケットを基礎に構成される、Secure RTCP パケットのフォーマットの例

SRTCP パケットの暗号化された部分は、最初の RTCP パケットから、同等の合成 RTCP パケットの RTCP ペイロードの暗号化 (セクション 4.1) 、つまり合成パケットの 9 番目のオクテットから最後で構成される。SRTCP パケットの認証された部分は、 (暗号化がペイロードに適用されたあとの) まったく同等の (最終的には合成) RTCP パケット、E フラグ、および SRTCP index で構成される。

追加されたフィールドを以下に記す。

E-flag: 1 ビット、必須

E-flag は現在の SRTCP パケットが暗号化されているかどうかを示す。[RFC3550] のセクション 9.1 では、合成 RTCP パケットを暗号化されたパケットとそうでないパケットの 2 つの低レイヤパケットに分割することを許可している。E ビットが ”1” にセットされている場合、暗号化パケットであることを示しており、”0” は非暗号化パケットであることを示している。

SRTCP index: 31 ビット、必須

SRTCP index は、SRTCP パケットのための 31 ビットのカウンタである。このインデックスは、SRTP で使用する ”implicit” index のアプローチとは対照的に、各パケットに明示的に含まれる。SRTCP index は最初の SRTCP パケットが送信される前に 0 に設定しなければならない [MUST] 。また、SRTCP index は、SRTCP パケットが送信されるたびに、1 ずつ増加し、2^31 でモジュロ演算されなければならない [MUST] 。特に、re-key のあと、SRTCP index は 0 にリセットしてはならない [MUST NOT] 。

Authentication Tag: 可変長、必須

Authentication Tag はメッセージの認証データを格納するのに使用される。

MKI: 可変長、オプション

MKI は Master Key Indicator の略であり、セクション 3 の MKI 定義に従って機能する。

SRTCP は、下記の変更とともに、デフォルトで暗号化コンテキストパラメータと SRTP のパケット処理を使用する。

- 受信側は、パケット内で明示的に通知されるため、index の ”推定”を必要としない。
- 事前定義された SRTCP 暗号化はセクション 4.1 に示されているが、このセクションで与えられた SRTCP Encrypted Portion の定義を使用し、SRTCP index を index i として使用する。暗号化変換とそれに関連するパラメータは、関連する SRTP ストリームの保護のために選択されたものと同じでなければならないが、一方で NULL アルゴリズムは暗号化されない RTCP パケットに適用される [SHALL] 。SRTCP は、対応する SRTP によって使用されるものとは異なる暗号化変換を有することができる [MAY] 。この機能の期待される使用方法は、前者が NULL 暗号化を持ち、後者が非 NULL 暗号化を持つ場合である。

E フラグには、パケットが暗号化されているかどうかに応じて、送信側によって値が割り当てられる。

- SRTCP の復号化は、セクション 4 で示されている通り実行されるが、E フラグが 1 に等しい場合にのみ実行される。その場合、暗号化された部分は、SRTCP index に index i を使用して復号化される。 E フラグが 0 である場合、ペイロードは単純に変更されずに残される。
- SRTCP リプレイ保護はセクション 3.3.2 で定義されているとおりであるが、SRTCP index を index i として使用し、SRTCP に個別の固有リプレイリストを使用する。
- 事前定義された SRTCP 認証タグはセクション 4.2 のように指定されるが、SRTCP パケットの認証された部分はこのセクションで指定される (index を含む) 。認証変換および関連するパラメータ (例えばキーのサイズ) は、関連する SRTP ストリームの保護のために選択されたものと同じでなければならない。
- 処理の最後のステップでは、送信側だけが SRTCP index を 2 ^ 31 でモジュロ演算した結果にインクリメントして更新する必要があり、セキュリティ上の理由から送信側も処理済みの SRTCP パケットの数をチェックしなければならない [MUST] 。

RTP のための制御プロトコル (例えば、それは BYE パケットを持つ) であるため、RTCP のためのメッセージ認証は必須である [REQUIRED] 。

SRTCP のパケット拡張 (追加されたフィールドによる) が、SRTCP メッセージに共有された RTCP 帯域幅以上に使用されないように注意する必要がある。これを回避するためには、次の 2 つの措置を講じなければならない。

1. [RFC3550] のセクション 6.3 で定義された RTCP 変数 "avg_rtcp_size" を初期化するときは、SRTCP によって追加されるフィールド (index、E ビット、認証タグ、および存在する場合は MKI) のサイズを含める必要がある。
2. 変数 "packet_size" ([RFC3550] のセクション 6.3.3) を使用して "avg_rtcp_size" を更新するとき、 "packet_size" の値は SRTCP によって追加された追加フィールドのサイズを含まなければならない [MUST] 。

これらの措置を講じることにより、SRTCP メッセージは、割り当てられた以上の帯域幅を使用することはない。追加されたフィールドのサイズが SRTCP トラフィックに及ぼす影響は、メッセージがより長いパケット間隔で送信されるということである。間隔の増加は、追加されたフィールドのサイズに正比例する。事前定義された変換では、追加されたフィールドのサイズは少なくとも 14 オクテットになり、MKI と認証タグのサイズに応じて上限が決まる。

4. 事前定義された暗号変換 (RFC3711 セクション 4)

SRTP で使われることができる多数の暗号化とメッセージ認証アルゴリズムがある間、アルゴリズムとパラメータ識別子のシグナリングのためにエンコーディングを指定する複雑さを避けるために、デフォルト・アルゴリズムを定義する。
セクション 2 にリストされたゴールを実現させるために定義されたアルゴリズムは選ばれた。新しい変換を用いて SRTP を拡張する方法に関する推奨事項は、セクション 6 で与えられる。

4.1 暗号化

以下のパラメータは、このセクションで規定している定義されている事前定義された暗号変換、非 NULL 暗号変換に共通である。

- BLOCK_CIPHER-MODE は、使われるブロック暗号とブロック暗号の利用モードを示す。
- n_b はブロック暗号の該当ブロックのビットサイズである。
- k_e はセッション暗号キーである。
- n_e は k_e のビット長である。
- k_s はセッションのソルトキーである。
- n_s は k_s のビット長である。
- SRTP_PREFIX_LENGTH はキーストリーム・プレフィックス、使用中のメッセージ認証コードにより特定される正の整数のオク

テット長である。

SRTP/SRTCP のための異なったセッションキーとソルトのデフォルトはセクション 4.3 により得られる。

SRTP で定義された暗号化変換は SRTP パケット・インデックスとの秘密キーを疑似乱数キーストリーム・セグメント内にマッピングする。各キーストリーム・セグメントは単一の RTP パケットを暗号化する。パケットを暗号化するプロセスは、パケットに対応するキーストリーム・セグメントを生成し、次にそのキーストリーム・セグメントを RTP パケットのペイロードにビットごとの排他的論理和にして、SRTP パケットの暗号化された部分を生成することからなる。ペイロードサイズが n_b ビットの整数倍でない場合、キーストリームの超過 (最下位) ビットは単純に破棄される。 暗号解読は平文と暗号文の役割を入れ替えて同じ方法で行われる。

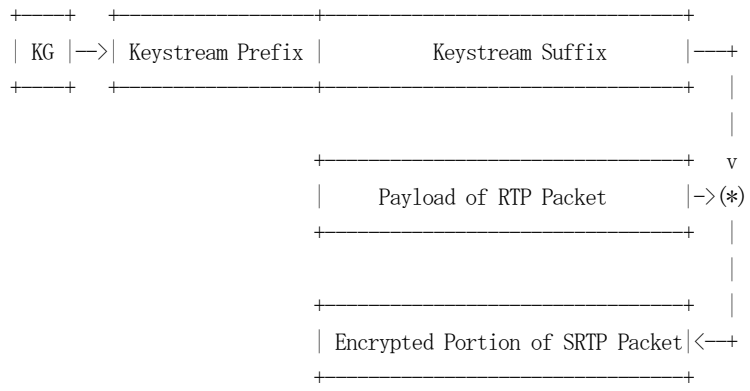


図 3 : デフォルト SRTP 暗号化処理。ここでは、KG はキーストリーム・ジェネレータを意味し、(*) はビットごとの排他論理和を意味する。

インデックスが与えられたときのキーストリームの生成方法の定義は、暗号とその動作モードによって異なる。以下では、このようなキーストリーム・ジェネレータが 2 つ定義されている。NULL 暗号も定義されており、RTP の暗号化が不要な場合に使用される。

キーストリームの SRTP 定義を図 3 に示す。各キーストリーム・セグメントの最初のオクテットは、メッセージ認証コードで使用するために予約することができる [MAY] 。この場合、暗号化に使用したキーストリームは最後の予約オクテットの直後に開始する。最初の予約されたオクテットは “キーストリーム・プレフィックス” ([RFC3550、セクション 6.1] の ”暗号化プレフィックス”と混同しないこと) と呼ばれ、残りのオクテットは"キーストリーム・サフィックス"と呼ばれる。“キーストリーム・プレフィックス” は暗号化に使用してはならない [MUST NOT] 。このプロセスを図 3 に示す。

キーストリーム・プレフィックスのオクテット数は SRTP_PREFIX_LENGTH と表される。キーストリーム・プレフィックスは、正の 0 以外の値の SRTP_PREFIX_LENGTH で示される。これは、機密性が提供されなくても、パケット認証のためにキーストリーム・ジェネレータの出力を計算する必要があるかもしれないことを意味する。この場合、デフォルトのキーストリーム・ジェネレータ (モード) が使用される [SHALL] 。

デフォルトの暗号は Advanced Encryption Standard (AES) であり、AES を実行する 2 つのモード、(1) セグメント化整数カウンタ・モード AES および (2) AES を f8 モードで定義する。このセクションの残りでは、E(k、x) をキーk と入力ブロック x に適用される AES とする。

4.1.1 カウンタ・モードの AES

概念的には、カウンタ・モード [AES-CTR] は、連続した整数を暗号化することで構成される。実際の定義は、整数シーケンスの開始点をランダム化するために、やや複雑である。各パケットは個別のキーストリーム・セグメントで暗号化される。これは次のように計算される [SHALL] 。

キーストリーム・セグメントは、AES 暗号の 128 ビット出力ブロックを、ブロックインデックスが昇順になるキーk = k_e を使用して、暗号化方向に連結するものとする [SHALL] 。象徴的に各キーストリーム・セグメントは次のようになる。

E(k, IV) || E(k, IV + 1 mod 2^128) || E(k, IV + 2 mod 2^128) ...

128 ビット整数値 IV は、以下のように、SSRC、SRTP パケット・インデックス i、および SRTP セッション・ソルトキーk_s によって定義される。

IV = (k_s * 2^16) XOR (SSRC * 2^64) XOR (i * 2^16)

上記の XOR 合計の 3 つの各項には、128 ビット値と見なされる操作を明確にするために必要な数の先頭ゼロが埋め込まれている。

SSRC を含めることにより、同一の RTP セッション内の異なる SRTP ストリームを保護するために同じキーを使用することができる。セクション 9.1 のセキュリティ上の注意を参照。

SRTCP の場合、複合パケットの最初のヘッダの SSRC が使用されなければならない [MUST] 。i は 31 ビット SRTCP index でなければならず [SHALL] 、k_e、k_s は SRTCP 暗号化セッションキーとソルトに置き換えられるべきである [SHALL] 。

初期値 IV は各パケットに対して固定され、カウンタのために最下位ビットに 16 個のゼロを「予約」することによって形成されることに注意すべきである。IV の任意の固定値に対して生成されるキーストリームのブロック数は、キーストリームの再使用を避けるために 2^16 を超えてはいけない[MUST]。以下を参照。 AES は 128 ビットのブロックサイズを持っているので、2^16 の出力ブロックは、最大の可能な RTP パケットを暗号化するのに必要なキーストリームの 2^23 ビットを生成するのに十分である (IPv6 "jumbograms" [RFC2675] RTP ベースのマルチメディアトラフィックに使用される) 。暗号化できるパケットの最大ビットサイズに対するこの制限は、確率的攻撃 [BDJR] の有効性を制限することによって暗号化方法のセキュリティを保証する。

特定のカウンターモードキーの場合、入力として使用される各 IV 値は、2 回パッド状態 (セクション 9.1) のセキュリティエクスポージャーを避けるために、区別されなければならない [MUST] 。この制約を満たすために、実装は、ROC ‖ SEQ の SRTP パケット・インデックスの組み合わせ、および IV の構築に使用される SSRC は、いずれかの特定のキーについて異なる。この一意性を保証しないことは、Secure RTP にとって致命的なものになる。これは、RTP 自体の状況と対照的であり、そのような障害を許容できる可能性がある。専用のセキュリティモジュールが存在する場合､RTP シーケンス番号と SSRC がそのモジュールによって生成またはチェックされることが推奨される (すなわち、SRTP システム内のシーケンス番号と SSRC 処理はキーと同様に保護される必要がある) 。

4.1.2 f8 モードの AES

UMTS (3G ネットワークとしての Universal Mobile Telecommunications System) データを暗号化するために、f8 アルゴリズムとして知られるソリューション ([f8-a]、[f8-b] を参照) が開発された。高レベルでは、提案された方式は、より精巧な初期化およびフィードバック機能を有する出力フィードバックモード (OFB) [HAC] の変形である。通常の OFB と同様に、コアはブロック暗号で構成される。ここでは、AES をブロック暗号として使用して、「f8 モードの操作」RTP 暗号化と呼ばれるもので使用することも定義する。AES f8 モードでは、AES カウンタ・モードと同じセッションキーとソルトのデフォルトサイズを使用しなければならない [SHALL] 。

図 4 に f8-モード動作時のブロック暗号構造「E」を示す。

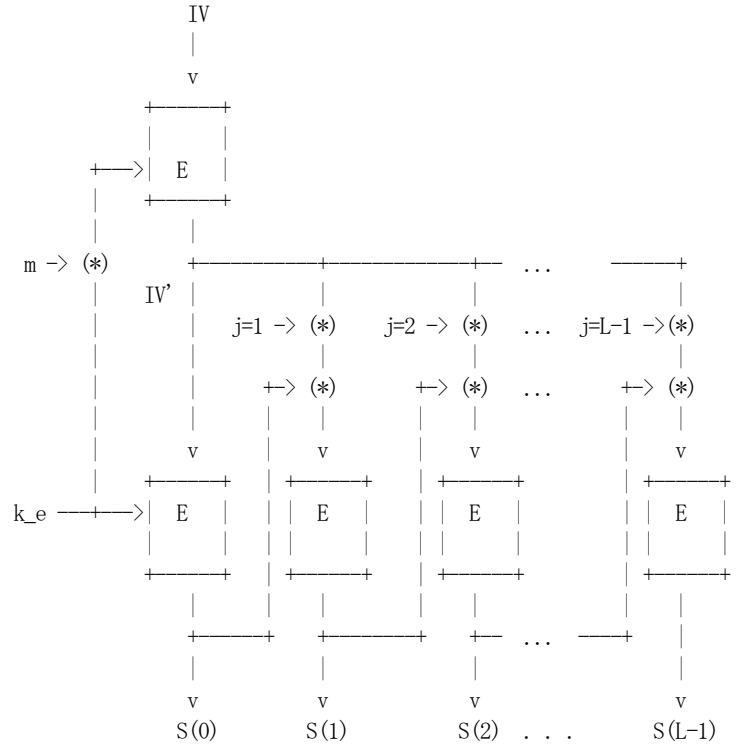


図 4. f8-利用モード (アスタリスク (*) はビット処理排他的論理和を意味する)

AES-f8 を使用した場合、この図は図 3 内の KG を表す。

4.1.2.1 f8 キーストリーム生成

初期化ベクトル (IV) はセクション 4.1.2.2 (SRTCP の場合はセクション 4.1.2.3) で記載される内容により決定されなければならない [SHALL] 。

IV’、S(j)、m は n_b ビットのブロックを示す。キーストリーム、S(0)‖... ‖S(L-1)、N ビットメッセージの場合は、IV' = E(k_e XOR m, IV) 、S(-1) = 00..0 で定義されなければならない [SHALL] 。j = 0,1,..,L-1 について、L = N/n_b (整数でない場合は、最も近い整数に切り上げ) が計算される。

S(j) = E(k_e, IV' XOR j XOR S(j-1))

IV は直接使用されないことに注意すること。代わりに、攻撃者が既知の入力／出力ペアを取得するのを防ぐために、別のキーの下で E を介して内部の「マスクされた」値 (IV 'で示される) を生成する。内部カウンタ j の役割は、短いキーストリームサイクルを防ぐことである。キーマスクの値 m は

m = k_s ‖ 0x555..5,

すなわち、所望のキーサイズ n_e 全体を埋めるためにバイナリ・パターン 0101 ...によって付加されたセッション・ソルトキーを含む [SHALL] 。

送信者は、2^39 ビットのキーストリームを生成するのに十分な 2^32 ブロック以上を生成してはならない [SHOULD NOT] 。カウンタ・モードとは異なり、f8 が安全でない (安全な) ことが保証されている絶対しきい値は上にない (下にもない) 。上記の境界は、十分なセキュリティマージンを持って、f8 キーストリーム生成における変性作用の確率を制限するように選択されている。

4.1.2.2 f8 SRTP IV 構造

以下で述べる IV 構造の目的は、暗黙のヘッダ認証 (IHA) という特徴を提供することである。セクション 9.5 を参照。

128 ビットの AES-f8 のための SRTP IV は以下のように作られなければならない [SHALL] 。

IV = 0x00 ‖ M ‖ PT ‖ SEQ ‖ TS ‖ SSRC ‖ ROC

M, PT, SEQ, TS, SSRC は RTP ヘッダから得られ、ROC は暗号コンテキストから得られなければならない [SHALL] 。

マスターキーが同一 RTP セッション内の複数のストリームで共有される場合、IV の一部として SSRC の存在は AES-f8 が使用されることを許容する。セクション 9.1 を参照。

4.1.2.3 f8 SRTCP IV 構造

128 ビットの AES-f8 のための SRTCP IV は以下のように作られなければならない[SHALL]。

IV= 0..0 ‖ E ‖ SRTCP index ‖ V ‖ P ‖ RC ‖ PT ‖ length ‖ SSRC

V, P, RC, PT, length, SSRC は RTCP パケットの最初のヘッダから得られなければならない [SHALL] 。

E は RTCP パケットの付加フィールドの 1 ビット目、SRTCP index は RTCP パケットの付加フィールドのドの 2 ビット目から 31 ビットの領域から得られる。

4.1.3 NULL 暗号化

NULL 暗号化は守秘性のない RTP/RTCP が要求された場合に使用される。キーストリームは、「000..0」とみなすことができる。例えば、単にプレーンテキスト入力を暗号文出力にコピーする [SHALL] 。

4.2 メッセージ認証と完全性

このセクション全体を通して、M は完全性保護されるべきデータを示す。 SRTP の場合、M は、ROC ,と連結されたパケットの認証された部分 (図 1 で指定されたもの) で構成されます [SHALL] 。M =認証された部分 ‖ ROC となります。また SRTCP の場合、M は認証された部分 (図 2 で指定されている) のみで構成されます [SHALL] 。

共通パラメータ

AUTH_ALG は認証アルゴリズムです。

k_a はセッションメッセージ認証キーです。

n_a は認証キーのビット長です。

n_tag は、出力認証タグのビット長です。

SRTP_PREFIX_LENGTH は、上で定義したキーストリーム・プレフィックスのオクテット長であり、AUTH_ALG のパラメータです。

SRTP / SRTCP の個別のセッション認証キーは、デフォルトはセクション 4.3 で指定されているように導出しています。

n_a、n_tag、及び SRTP_PREFIX_LENGTH の値は、キーのいずれかの特定の固定値のために固定されなければなりません [MUST] 。

認証タグを計算するプロセスについては、次のように説明します。 送信者は、M のタグを計算し、それをパケットに付加する。 SRTP 受信機は、選択されたアルゴリズムおよびキーを使用して M 上の新しい認証タグを計算することによってメッセージ/認証タグペアを検証し、受信したメッセージに関連付けられたタグと比較する。 2 つのタグが等しい場合、メッセージ/タグのペアは有効です。 さもなければそれは無効であり、エラー監査メッセージ "AUTHENTICATION FAILURE" が返されなければなりません [MUST] 。

4.2.1 HMAC-SHA1

SRTP のあらかじめ定義された認証変換は HMAC-SHA1 [RFC2104] です。 HMAC-SHA1 の場合、SRTP_PREFIX_LENGTH (図 3) は 0 とする [SHALL] 。SRTP (または SRTCP) の場合、HMAC はセッション認証キーと M に上記の、すなわち HMAC (k_a、M) を適用する [SHALL] 。HMAC 出力は、n_tag の左端のビットに切り詰められる [SHALL] 。

4.3 キー導出

4.3.1 キー導出アルゴリズム

使用される暗号化またはメッセージ認証変換 (SRTP 事前定義変換であるか、またはセクション 6 に従って新たに導入されたものであってもよい) にかかわらず、相互運用可能な SRTP 実装は、セッションキーを生成するために SRTP キー導出を使用しなければならない [MUST] 。キー導出率がセッションの開始時に適切に通知されると、SRTP キー導出を使用する当事者間の余分な通信は必要ありません。

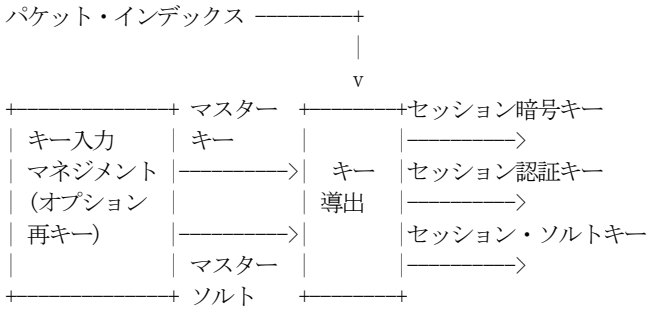


図 5 : SRTP キー導出

少なくとも 1 つの初期キー導出が SRTP によって実行されなければならない [SHALL] 。すなわち、最初のキー導出が必要である [REQUIRED] 。キーの導出のさらなる適用は、暗号文脈における「キー導出率」値に従って実行されてもよい [MAY] 。キー導出関数は最初に最初のパケットの前に呼び出され、次に r > 0 のときはインデックス ÷ r の余りがゼロのときは常にキー導出が実行されます [SHALL] 。これは、セッションキーを「リフレッシュする」と考えることができます。 「キー導出率」の値は、関連するマスターキーの存続期間中は固定されていなければなりません [MUST] 。

相互運用可能な SRTP 実装は、事前定義された両方の変換で行われるように、暗号化変換用のセッションソルティングキーを導出することもできます [MAY] 。

m と n を正の整数とする。 擬似乱数関数群は、一式のキー付き関数 {PRF_n(k,x)} であり、それは (秘密な) ランダムキー k と、与えられた m ビットの x に対して、PRF_n(k,x) が n ビットの文字列であり、計算上、ランダムな n ビットの文字列と区別できません。[HAC] を参照してください。 SRTP のキー導出の目的で、m = 128 (またはそれ以上) の安全な PRF を使用しなければならず、デフォルトの PRF 変換がセクション 4.3.3 で定義されていなければならない [MUST] 。

"a DIV t" は、a の t による整数除算を表し、丸められたものであり、すべての a に対して "a DIV 0 = 0" であるとする。 また、"a DIV t" を a と同じ長さのビット列として扱うという条約を作成し、したがって " a DIV t" は一般に 0 パディングされます。

キーの導出は、<ラベル>、8 ビット定数 (以下、参照) 、マスターソルトおよびキー導出率 (暗号文脈で決定される) 、および索引、パケット索引に関して以下のように定義されるものとする [SHALL] 。 (すなわち 48 ビットの ROC || SRTP の SEQ) :

- r =インデックス DIV キー導出率 (上で定義した DIV) 。
- key_id = <label> || r。
- x = key_id XOR master_salt。ここで、key_id と master_salt は、最下位ビットが一致するように整列されます (右揃え) 。

<label> は、導出するキーの種類ごとに一意でなければなりません [MUST] 。現在、<label> 0x00～0x05 (下記参照) を定義しており、将来の拡張では、他の目的のために 0x06～0xff の範囲の新しい値を指定することがあります [MAY] 。このパケットのための n ビットの SRTP キー (またはソルト) は、マスターキーと k_master から次のように導かれます。

PRF_n(k_master, x).
(PRF は、x の追加のフォーマットとパディングを内部的に指定することができます。例えば、デフォルトの PRF のセクション 4.3.3 を参照してください。)

セッションキーとソルトは、以下を使用して導出されるべきです [SHALL] :

- k_e (SRTP encryption): <label> = 0x00, n = n_e.
- k_a (SRTP message authentication): <label> = 0x01, n = n_a.
- k_s (SRTP salting key): <label> = 0x02, n = n_s.

n_e、n_s、および n_a は暗号コンテキストからのものです。

マスターキーとマスターソルトはランダムでなければならないが [MUST] 、マスターソルトは公開されていてもよい [MAY] 。

キー導出率が 0 の場合、キー導出のアプリケーションは正確に 1 回行われることに注意してください [SHALL] 。

上記の DIV の定義は、純粋に表記上の便宜のためのものです。 許容されるキー導出率のセットの中でゼロでない t の場合、 “a DIV t” は、t の 2 の対数による右シフトとして実装することができる。 レートが 256 の累乗になるように選択された場合、導出操作はさらに容易になるが、その細分性は本仕様の要件となるにはあまりに粗すぎると考えられた。

同じマスターキー (セクション 9.2 を参照) を使用して保護できるパケット数の上限は、キーの導出に依存しません。

4.3.2 SRTCP キー導出

SRTCP は、デフォルトで SRTP と同じマスターキー (およびマスターソルト) を使用するものとします [SHALL] 。これを確実に行うために、SRTCP にセッションキー導出を適用するときは、セクション 4.3.1 の定義に以下の変更を行うものとする [SHALL] 。

SRTP インデックスを 32 ビット数で置き換え、SRTCP 暗号化キーには <label> = 0x03 、SRTCP 認証キーには <label> = 0x04 、SRTCP ソルトキーには <label> = 0x05 を使用します。

4.3.3 AES-CM PRF

現在定義されている PRF は、128、192、または 256 ビットのマスターキーでキーイングされ入力ブロックサイズ m = 128 を有し、2^23 まで n の n ビット出力を生成することができる。 PRF_n(k_master,x) は、セクション 4.1.1 で説明したようにカウンタ・モードの AES であり、キーk_master に適用され、IV は (x*2^16) に等しく、出力キーストリームは最初の n ビットに切り詰められる。 (n / 128 して、切り上げ、AES のアプリケーションを要求する。)

5. デフォルトで実装が必須の変換 (RFC3711 セクション 5)

デフォルトの変換は、SRTP で実装必須な変換でもある。もちろん、「必ず実装すること」というのは「必ず使用すること」を意味するものではない。表 1 は、事前定義された変換をまとめたものである。以下のデフォルト値は、事前定義された変換に有効である。

	mandatory-to-impl.	optional	default
encryption	AES-CM, NULL	AES-f8	AES-CM
message integrity	HMAC-SHA1	-	HMAC-SHA1
key derivation (PRF)	AES-CM	-	AES-CM

表 1: SRTP と SRTCP の実装必須、オプションおよびデフォルトの変換。

5.1 暗号化:AES-CM および NULL

セグメンテーションされた整数カウンタ・モードで動作する AES は、セクション 4.1.1 で定義されているように、デフォルトの暗号化アルゴリズムでなければならない [SHALL] 。デフォルトのキー長は、セッション暗号キー (n_e) の 128 ビットである。デフォルトのセッションソルトのキー長 (n_s) は 112 ビットでなければならない [SHALL] 。

NULL 暗号もまた、実装必須である [SHALL] 。

5.2 メッセージ認証/整合性:HMAC-SHA1

HMAC-SHA1 は、セクション 4.2.1 で定義されているように、デフォルトのメッセージ認証コードであるべきである [SHALL] 。HMAC-SHA1 の場合、デフォルトセッション認証キー長 (n_a) は 160 ビット、デフォルト認証タグ長 (n_tag) は 80 ビット、SRTP_PREFIX_LENGTH はゼロでなければならない [SHALL] 。さらに、SRTCP の場合、事前定義された HMAC-SHA1 は、これらのデフォルトより小さい n_tag または n_a の値を適用してはならない [MUST NOT] 。 SRTP の場合、より小さい値は推奨されないが、セク

ション 7.5 と 9.5 の問題を慎重に考慮した上で使用することができる。

5.3 キー導出:AES-CM PRF

セクション 4.3.1～4.3.3 で定義された、AES カウンタモードベースのキー導出と PRF は、128 ビットのマスターキーを使用して、セッションキーを生成するためのデフォルトのメソッドである必要がある。デフォルトのマスターソルトの長さは 112 ビットであり、デフォルトのキー導出率はゼロでなければならない [SHALL] 。

6. SRTP 変換の追加 (RFC3711 セクション 6)

セクション 4 では、変換の定義に必要な詳細レベルの例を示す。SRTP に新しい変換が追加されるときは、新しい変換がどのように SRTP (および SRTCP) で使用されるかを正確に定義するために、Standards Track の RFC を記述しなければならない [MUST] 。そのような付随する RFC は、SRTP プロトコル文書との重複を避けるべきである [SHOULD] 。ただし、SRTP または SRTCP の暗号化コンテキストの定義を新しいパラメータ (固定値またはデフォルト値を含む) で拡張したり、パケット処理にステップを追加したり、SRTP / SRTCP パケットにフィールドを追加したりする必要があるかもしれない [MAY] 。付随 RFC は、変換と SRTP の他の側面との間の相互作用に関する任意の既知の問題を説明しなければならない [SHALL] 。

新しい変換のための文書は、キーのサイズ (最小、最大、推奨) 、キーのフォーマット、入力キー要素の推奨/必要処理、キーライフタイムに関する要件/推奨事項、キー変更とキー導出、 SRTP と SRTCP の間のキーの共有が許可されるかどうかなどの、キー属性を指定するべきである [SHOULD] 。

追加されたメッセージの完全変換は、セクション 5.2 で定義されている最小値と同等の SRTCP の最小許容キー/タグサイズを定義しなければならない [SHOULD] 。

7. 根拠 (RFC3711 セクション 7)

このセクションでは、SRTP のいくつかの重要な機能の背後にある根拠について説明する。

7.1 キーの導出

キーの導出は、キーの設立にかかる負担を軽減する。暗号コンテキスト (SRTP と SRTCP の暗号キーとキー、SRTP と SRTCP 認証キー) ごとに 6 つの異なるキーが必要だが、これらは暗号的に安全な方法で単一のマスターキーから得られる。したがって、キー管理プロトコルは、1 つのマスターキー (必要に応じてマスターソルト) を交換する必要があり、SRTP 自体が (キー導出機能の第 1 の必須アプリケーションを介して) 必要なセッションキーをすべて導出する。

キー導出機能の複数のアプリケーションはオプションとなっているが、有効にするとセキュリティ上の利点を得ることができる。攻撃者が 1 つの固定セッションキーで生成された大量の暗号文を取得できないようにする。攻撃者が特定のセッションキーに対して大量の暗号文を収集できた場合、特定の攻撃の実装に役立つ可能性がある。

キー導出機能の複数のアプリケーションは、漏洩したセッションキーが同じマスターキーから派生した他のセッションキーを漏洩しないという意味で、後方および前方のセキュリティを提供する。つまり、特定のセッションキーを回復できる攻撃者は、前と後のセッションキー (同じマスターキーから派生) で保護されたメッセージにはアクセスできなくなる。 (当然ながら、漏洩したマスターキーは、それから派生したすべてのセッションキーを明らかにする。)

高速のキーリフレッシュ、特に大規模なマルチキャスト設定での考慮すべき事項が発生する (セクション 11 を参照) 。

7.2 ソルティングキー

マスターソルトは、実効キーサイズ [MF00] を減らす可能性のあるキー導出に対するオフラインキー衝突攻撃に対するセキュリティを保証する。

暗号化に使用される派生したセッション・ソルトキーは、追加ストリーム暗号に対するいくつかの攻撃から保護するために導入された。セクション 9.2 を参照。 IV におけるソルトの明示的な包含方法は、ハードウェア実装を容易にするために選択されている。

7.3 ユニバーサルハッシュからのメッセージの整合性

セクション 4.1 (キーストリーム・プレフィックス) で与えられたキーストリームの特定の定義は、Wegman-Carter パラダイム [WC81] におけるメッセージ認証に適した特定の汎用ハッシュ関数のための規定を与えることである。そのような機能は、確かに安全で、簡単で、迅速で、特にデジタル信号プロセッサおよび高速の乗算操作を有する他のプロセッサに適している。

現在、HMAC-SHA1 以外の SRTP では認証変換は提供されていない。前述の汎用ハッシュ関数のような将来の変換は、セクション 6 のガイドラインに従って追加することができる [MAY] 。

7.4 データ発信元の認証に関する考慮事項

ペアワイズ通信では、整合性とデータ発信元認証が一緒に提供されることに注意しなければならない。しかし、メンバー間でキーが共有されるグループシナリオでは、MAC タグはそのグループのメンバーがパケットを送信したことは証明するが、別のメンバーを偽装するメンバーに対しては防御しない。マルチキャストおよびグループ RTP セッションのデータ発信元認証 (DOA) は、解決策が必要な難しい問題である。いくつかの有望な提案が検討されているが [PCST1] [PCST2] 、これらの技術を厳密に特定するためにはより多くの作業が必要である。したがって、グループ内の SRTP データ発信元認証は、今後の検討のためのものある。

DOA は、署名を使用して別の方法で実行できる。しかし、これは帯域幅と処理時間の点で大きな影響を与える。したがって、あらかじめ定義されたパケット完全性変換でこの形式の認証を提供しない。

ミキサーおよびトランスレータの存在は、RTP ペイロードおよびまたは RTP ヘッダが操作された場合にデータ発信元の認証を許可しない。これらのタイプの中間エンティティは、エンドツーエンドの機密性 (IV 形成は、例えば、RTP ヘッダの保存に依存するため) を中断させることにも留意されたい。特定のトラストモデルは、ミキサー/トランスレータがメディアを復号化/再暗号化することを信頼するということを選択することができる (これは、関連するセキュリティの影響を伴い、エンドツーエンドのセキュリティを壊すことを意味する) 。

7.5 長さが短いまたはゼロのメッセージ認証

図 1 に示すように、SRTP では認証タグが推奨されている [RECOMMENDED] 。完全な 80 ビット認証タグを使用すべきであるが [SHOULD] 、以下の 2 つのアプリケーション環境のいずれかをサポートするために、特定の条件の下で短いタグまたは長さゼロのタグ (つまり、メッセージ認証なし) を使用することができる [MAY] 。

1. 強力な認証は、帯域幅の保護が不可欠な環境では実用的ではない。特に重要で特殊なケースは、帯域幅が不足しがちで高価なリソースである無線通信システムである。いくつかのアプリケーションとリンクテクノロジでは、追加のバイトでスペクトル効率が大幅に低下する可能性があることが研究によって示されている [SWO] 。スペクトル効率を改善するために IP ヘッダ圧縮技術を設計するためにかなりの努力がなされている [RFC3095] 。一般的な音声アプリケーションは 20 バイトのサンプルを生成し、許容可能な無線帯域幅の経済性 [RFC3095] を得るために、RTP、UDP、および IP ヘッダを平均して 1〜2 バイトに圧縮する必要がある。この場合、強力な認証では 50%近くのオーバーヘッドが発生する。

2. 認証タグによる拡張に対応できない固定幅のフィールドを持つデータリンクを使用するアプリケーションでは、認証は実用的ではない。これは、いくつかの重要な既存の無線チャネルの場合である。例えば、ゼロバイトヘッダ圧縮は、CDMA2000 VoIP サービスにおけるレガシーIS-95 ベアラチャネルを用いて EVRC / SMV 音声を適応させるために使用される。追加のオクテットを 1 つ追加することはできず、ROHC [RFC3242] のためのゼロバイトプロファイルの作成を促した。

短いタグは、限られたアプリケーションのセットに対して安全である。例えば、32 ミリ秒のメッセージ認証タグによって保護された 20 ミリ秒のパケット化間隔を有する G.729 音声コーデックのような音声電話アプリケーションを考えてみる。与えられたパケットの偽造が成功する可能性は 2 ^ 32 に 1 つだけです。したがって、敵は、994 日間平均で 20 ミリ秒以下のオーディオ出力を制御できる。対照的に、アプリケーションがステートフルであれば、単一の偽造パケットの効果は非常に大きくなる可能性がある。パケット間で相対的または予測的な圧縮を使用するコーデックは、悪意を持って生成された状態を伝播し、より長い出力期間に影響する。

確かに、すべての SRTP アプリケーションまたはテレフォニーアプリケーションが、長さがゼロまたは長さの認証タグの基準を満たしているわけではない。セクション 9.5.1 では、メッセージの認証が弱い、またはないというリスクについて説明し、セクション 9.5 では、それが容認可能な状況、および、容認できない状況について説明します。

8. キー管理考察 (RFC3711 セクション 8)

SRTP 暗号コンテキスト (例えば、SRTP マスターキー) を確立するための新たなキー管理標準 [MIKEY] [KEYMGT] [SDMS] がある。独自技術とオープンスタンダードの両方のキー管理方法は、テレフォニーアプリケーション [MIKEY] [KINK] とマルチキャストアプリケーション [GDOI] に使用される可能性がある。このセクションでは、SRTP セッションに対応するキー管理システムのガイダンスを提供する。

初期化のために、相互運用可能な SRTP 実装は、SSRC を与えられなければならず [SHOULD] 、キー管理によって RTP ストリームのための初期 RTP シーケンス番号が与えられなければならない [MAY] (したがって、キー管理は RTP 運用パラメータに依存する) 。キー管理における RTP シーケンス番号の送信は、例えば、初期シーケンス番号が (同期問題を回避するために) ラッピングに近くにあり、(そのリプレイリストを適切に初期化するために) 現在のシーケンス番号を結合エンドポイントに伝える際に役に立つ。

事前定義された変換が使用される場合、SRTP は同じ RTP セッションに属する SRTP / SRTCP ストリーム間で同じマスターキーの共有を許可する。

第 1 に、同じ RTP セッションに属する SRTP ストリーム間の共有は、同期メカニズム、すなわち IV の設計がキーストリーム再利用 (two-time pad、セクション 9.1) を回避するならば安全である。これは、RTP が同じ RTP セッションに属するストリームのためのユニークな SSRC を提供するという事実によって対処される。詳細はセクション 9.1 を参照。

第 2 に、SRTP と対応する SRTCP との間の共有は安全である。SRTP ストリームおよびそれに関連する SRTCP ストリームが両方とも同じ SSRC を運ぶという事実は、キー導出のため two-time pad の問題を構成しない。したがって、1 つの RTP セッションに対応する SRTP と SRTCP は、マスターキーを共有するかもしれない (デフォルトではマスターキーを共有する) 。

メッセージ認証には、キーストリームの再利用の問題とは無関係の SSRC の一意性にも依存していることに注意すること。同じキーで認証された SRTP ストリームは、メッセージの送信者を識別するために個別の SSRC を持たなければならない [MUST] 。SSRC は異なる SRTP ストリームを区別するために使用される暗号的に認証されたフィールドであるため、この要件が必要です。2 つのストリームが同じ SSRC 値を使用する場合、攻撃者は検出せずに一方のストリームのメッセージを他方のストリームに置き換えることができる。

SRTP / SRTCP は、上記以外の状況下ではマスターキーを共有してはならない [MUST NOT] 。
例えば、SRTP とそれに対応する SRTCP との間で共有してはならない。同じ RTP セッションに属するストリーム間で共有してはならない。

8.1 キー変更

特定のキー管理システムが SRTP 内でキー変更を提供するために推奨される方法は、暗号コンテキスト内のマスターキーを MKI に関連付けることである。

これにより、簡単にマスターキーを検索できる (セクション 11 のシナリオを参照) 。 ただし、各パケットに余分なビットを追加するという欠点がある。セクション 7.5 で説明したように、いくつかの無線リンクは追加ビットを提供しない。したがって、SRTP は、いくつかの特定の単純なシナリオで動作する <From、To> を使用してキー変更トリガーのより経済的な方法を定義する (セクション 8.1.1 を参照) 。

SRTP 送信者は、マスターキーに使用されている SRTP と SRTCP トラフィックの量を数え、必要に応じてキー管理を呼び出してキー変更する (セクション 9.2) 。これらの相互作用は、SRTP へのキー管理インタフェースによって定義され、このプロトコル仕様では定義されていない。

8.1.1 キー変更のために <From, To> の使用

SRTP は、MKI の使用に加えて、マスターキー取得のための別のオプションメカニズムである <From, To> を定義する。<From, To> は、特定のマスターキーが有効で、暗号コンテキストの (使用されている) 部分である SRTP インデックス (シーケンス番号と ROC のペア) の範囲を指定する。現在の SRTP パケットの 48 ビット SRTP インデックスを調べることによって、それが所属する From-To インターバルを決定することによって、対応するマスターキーを見つけることができる。SRTCP については、SRTCP が独自の (31 ビット) インデックスを持っていても、この目的のために使用された最新の観測された／使用された SRTP インデックス (暗号コンテキストから取得可能) が使用される (以下の注意を参照) 。

この方法は、MKI と比較して、マスターキーを識別し、各パケットに余分なビットを追加することなくその生存時間を定義するという利点がある。これは既に述べたように、追加ビットを提供していない無線リンクにとっては便利である。しかし、その使用は特定の非常に単純なシナリオに限定されるべきである [SHOULD] 。 RTP セッションが単純な単方向ストリームまたは双方向ストリームである場合は、その使用を制限することを推奨する。これは、複数のストリームの場合、単一の RTP ストリームの <From, To> に基づいてキー変更をトリガーすることが困難であるためである。たとえば、複数のストリームがマスターキーを共有する場合、特定のストリームのインデックスシーケンス空間と <From, To> 値の基になるインデックスシーケンス空間との間には、単純な 1 対 1 対応はない。その結果、マスターキーがストリーム間で共有されるとき、これらのストリームの 1 つは、インデックス空間がキー変更ポイントを定義するものとしてキー管理によって指定されなければならない [MUST] 。また、SRTCP 上のキー変更トリガーは、対応する SRTP ストリームに基づいており、すなわち、SRTP ストリームがマスターキーを変更すると、対応する SRTCP も変更される。複数のストリームでは、これは明らかにますます複雑になる。

<From, To> のデフォルト値は、「最初に観測されたパケットから」および「以降の通知があるまでの間」である。しかしながら、与

えられたマスター／セッションキー (セクション 9. 2) に従って送られる SRTP / SRTCP パケットの最大限度を超過してはいけない [MUST NOT] 。

<From, To> がキー検索として使用される場合、MKI はパケットに挿入されない (暗号コンテキストのインジケータはゼロである) 。ただし、MKI を使用しても、 <From, To> キーの存続時間を同時に使用することは除外されない。これは、例えば、MKI をアクティブにする時点で送信側から信号を送るのに役立つ。

8.2 キー管理パラメータ

次の表に、キー管理が提供できるすべての SRTP パラメータを示す。参考として、セクション 5 で説明されているように、SRTP 実装のデフォルトおよび必須のサポート値の要約も提供する。

パラメータ	必須サポート	デフォルト
SRTP and SRTCP 暗号化変換 (その他の可能な値: AES_f8)	AES_CM, NULL	AES_CM
SRTP and SRTCP 認証変換	HMAC-SHA1	HMAC-SHA1
SRTP and SRTCP 認証パラメータ:		
n_tag (タグ長)	80	80
SRTP プレフィックス長	0	0
キー導出 PRF	AES_CM	AES_CM
キー主要パラメータ (マスターキーごとに):		
マスターキー長	128	128
n_e (セッション暗号キー長)	128	128
n_a (認証セッションキー長)	160	160
マスターソルトキー		
マスターソルトの長さ	112	112
n_s (セッション・ソルトキー長)	112	112
キー導出率	0	0
キー存続時間		
SRTP パケットの最大存続時間	2^48	2^48
SRTCP パケットの最大存続時間	2^31	2^31
from-to 存続時間<From, To>		
MKI インジケータ	0	0
MKI 長	0	0
MKI 値		
暗号コンテキストインデックスパラメータ ;		
SSRC 値		
ROC		
SEQ		
SRTCP インデックス		
転送アドレス		
ポート番号		
他の RTP プロファイルに対する関係 :		
FEC と SRTP FEC-SRTP 間の送信側の順序。FEC-SRTP (セクション 10 を参照)		

9. セキュリティ問題 (RFC3711 セクション 9)

9.1 SSRC 衝突とツータイムパッド

同じキーとインデックスから生成された固定のキーストリーム出力は、一度だけ、暗号化に使用されなければならない [MUST] 。キーストリームの再利用 (暗号使用者によって「ツータイムパッド」システムをコールされるような) は、セキュリティが大幅に損なわれる可能性がある。NSA の VENONA プロジェクト [C99] は、このような侵害の歴史的な例を提供している。SRTP および SRTCP のキー材料の確立および維持には、自動キー管理を使用する必要がある [REQUIRED] ；この要件は手動キー管理で発生する可能性が高いキーストリームの再利用を避けることである。さらに、SRTP では、RTP/RTCP ストリームおよびパケットごとに一意であるように、キーまたは暗号重要性の他のパラメータを要求することによって、「ツータイムパッド」は回避される。あらかじめ定義された SRTP 変換は、SSRC を含めることによってパケット・インデックスとストリーム一意性を含めて、パケット一意性を達成する。

事前定義された変換 (AES-CM および AES-f8) により、IV 内に SSRC を含めることによって、同じ RTP セッションに属するストリーム間でマスターキーを共有することができる。マスターキーは、異なる RTP セッション間で共有されてはならない [MUST NOT] 。したがって、SSRC は、同じマスターキーを共有する同じ RTP セッション内のすべての RTP ストリーム間で一意である必要がある [MUST] 。RTP 自体は、同じ RTP セッション内の SSRC 衝突を検出するためのアルゴリズムを提供する。

したがって、一時的な衝突は一時的なツータイムパッドにつながる可能性がある。残念なことに SSRC はストリームにも同じシーケンス番号がある時点で衝突する (確率約 2^ (-48) で発生) 。したがって、キー管理はセッションで使用される SSRC をネゴシエーションパラメータとして含めて、そのような SSRC の衝突を回避し、その一意性を積極的に保証すべきである [SHOULD] 。これは、例えば SSRC コリジョンが RTP レベルで検出される前に複数の送信者が同時に送信を開始できるシナリオでは、強力な要件である。

また、異なる SSRC であっても同じキーを広範囲に使用することで、確率的衝突やタイムメモリトレードオフ攻撃が成功する可能性がある。

記述されているように、マスターキーは同じ RTP セッションに属するストリーム間で共有されても良いが [MAY] 、各 SSRC がそれ自身のマスターキーを持つことを推奨する [RECOMMENDED] 。マスターキーが SSRC 参加者間で共有され、SSRC が上記で推奨されているキー管理モジュールによって管理されている場合、SSRC コリジョンエラーの推奨ポリシーは、誤動作の兆候がある時に参加者が SRTP セッションを終了することである。

9.2 キーの使用

有効キーサイズは、マスターキーのサイズと暗号化のためのソルトキーのサイズによって決定される (上限がある) 。追加ストリーム暗号は、プレーンテキストソースに関する統計的知識を使用し、キー衝突とタイムメモリトレードオフ攻撃を有効にする攻撃に対しては脆弱である [MF00] [H80] [BS00] 。

これらの攻撃は、プレーンテキスト間の共通性を利用して、暗号解読者が多くのキーまたは何バイトにもわたる出力で復号化の計算量を削減する方法を提供し、暗号の有効キーサイズを縮小します。これらの攻撃とインターネットトラフィックの暗号化への適用に関する詳細な分析は、 [MF00] である。

要約すると、SRTP の有効キーサイズは、m 個の異なるキーが使用されるセキュリティシステムで使用される場合、暗号のキーサイズから m の対数を引いたもの (2 を引いたもの) に等しい。そのような攻撃に対する保護は、使用されるキーのサイズを増やすことによって簡単に提供することができる。ここでは、ソルトキーを使用することによって達成できる。ソルトキーはランダムでなければならないが [MUST] 、公開キーでもよい [MAY] ことに注意すること。112 ビットの (推奨される) ソルトサイズは、最大 2^112 キーが使用されているシナリオでの攻撃を保護する。これは、すべての実用的な目的には十分である。実装は可能な限り大きなキーを使用するべきである [SHOULD] 。多くの場合、暗号キーのサイズを大きくしても、その暗号のスループットには影響しないことに注意する。

事前定義された変換で SRTP および SRTCP インデックスを使用すると、同じキーで保護できるパケットの最大数が固定される。この制限は、SRTP ストリームの場合は 2^48 SRTP パケットで、SRTP と SRTCP が独立して考慮される場合は 2^31 SRTCP パケットに固定される。例えば、再キーイングのためにこの制限に達することは、インデックスの折り返しと一致することもあれば一致しないこともあるので、送信者はパケット数を保持しなければならない [MUST] 。ただし、関連する SRTP および SRTCP ストリームのセッションキーが同じマスターキー (デフォルト動作、セクション 4.3) から派生する場合、考慮する必要がある上限は実際には 2 つの数量の最小値である。

すなわち、2^48SRTP パケットまたは 2^31SRTCP パケットが同じキーで保護されている場合 (以前、いずれかに発生した方) には、新しいマスターキーを提供するためにキー管理を呼び出さなければならない [MUST] (以前に格納され使用されたキーは再び使用してはならない) か、セッションを終了しなければならない [MUST] 。RTCP の送信者が、同じ SRTP (SRTCP) ストリームに属する 2^48 SRTP (または 2^31 SRTCP) パケットを送信する前に、SRTP (または SRTCP) の送信者がマスターキーまたはセッションキーを更新していないこ

とを検出した場合、例えば、RTCP BYE パケットを送信すべきかどうか、および/またはイベントをログに記録すべきかは、RTCP 送信者のセキュリティポリシーに従う。

注：ほとんどの一般的なアプリケーションでは (128,000 RTP パケットごとに少なくとも 1 つの RTCP パケットがあると仮定する) これが発生するまでの時間は非常に長いものの、最初に上限に達する SRTCP インデックスになる：200 SRTCP パケット/秒でも、SRTCP の 2^31 インデックススペースは約 4 ヶ月の通信を確保するのに十分である。

マスターキーが同じ RTP セッション (セクション 9.1) 内の SRTP ストリーム間で共有される場合、上記の境界はストリームごと (すなわち、SSRC 毎) に基づいているが、送信者はシーケンス番号が最初に枯渇したストリームの再決定を行わなければならない [MUST] 。

キー導出は、固定セッションキーで暗号化され、解析のために攻撃者が利用できるプレーンテキストの量を制限するが、マスターキーの生存期間を延長しない。

これを確認するには、ツータイムパッドを避けるための要件を考慮すること: 2 つの異なるパケットは、異なる IV、または異なるセッションキーで処理されなければならず [MUST] 、IV およびセッションキーの区別は、パケット・インデックスの明確性に依存して (事前定義された変換のために) 行われる。

キー導出では、派生したセッションキーがかなり長くても、有効キーのサイズはマスターキーのものと同じであることに注意する。事前定義された認証変換では、セッション認証キーは 160 ビットであるが、マスターキーはデフォルトで 128 ビットのみである。この設計選択は、 [RFC2104] の推奨事項に従うために行われ、既存の HMAC 実装を SRTP に問題なく接続できる。

デフォルトのタグサイズは 80 ビットなので、セキュリティ上の観点から見れば、アプリケーションを考慮しても許容されると考えられる。これに関心のあるユーザは、キー導出に 192 ビットのマスターキーを代わりに使用することを推奨する [RECOMMENDED] 。しかし、キー導出に使用される既存の AES 実装が常に 128 ビット以外のキー長をサポートするとは限らないため、192 ビットキーを要求しないことが選択された。AES は 160 ビットキーで使用するために定義されていない (または適切に分析されていない) ので、短いキーを 192 または 256 ビットに埋め込むためにアドホックなキーパッド方式が使用されることは推奨されない [NOT RECOMMENDED] 。

9.3 RTP ペイロードの機密性

SRTP の事前定義された暗号は「検索可能な」ストリーム暗号であり、すなわち、1 つのパケットの暗号化または復号化が先行するパケットに依存しないように、それらのキーストリーム内の任意の位置に効率的に検索できる暗号がある。検索可能なストリーム暗号を使用することにより、SRTP はこのプロパティが不足しているストリーム暗号で可能なサービス拒否攻撃を回避する。ストリーム暗号と同様に、ペイロードの正確な長さは暗号化によって明らかにされることに注意することが重要である。

コーデック出力の長さが特定のパラメータ設定などによって変化する可能性があるため、ペイロードの特定の「フォーマットビット」を推測することが可能であることを意味する。次にキーストリームの対応するビットを推定できることを意味する。しかし、ストリーム暗号が安全である場合 (カウンタ・モードであり、特定の前提 [BDJR] [KSYH] [IK] の下で f8 が確かに安全である) 、キーストリームの数ビットの知識は、攻撃者が後続のキーストリームビットを予測するのに役立たない。したがって、ペイロードの長さ (およびこれから推測できる情報) は漏れるが、他には何もない。ある種の RTP パケットは、予測可能なデータ (例えば、SID) を含む可能性があるため、既知のプレーンテキスト攻撃 (現在の慣例) に耐えるように設計された暗号を使用することが重要である。

9.4 RTP ヘッダの機密性

SRTP では、ヘッダ圧縮を可能にするために RTP ヘッダがクリアで送信される。これは、ペイロードタイプ、同期ソース識別子、タイムスタンプなどのデータを盗聴者が利用できることを意味する。さらに、RTP は将来ヘッダの拡張を可能にするので、どのような機密情報も「漏洩」する可能性は予見できない。

SRTP は低コストの方法であり、ヘッダ圧縮により帯域幅を削減できる。採用するセキュリティプロトコルについての決定は、エンドポイントのポリシーに従う。実際にヘッダを保護する必要があり、周囲の環境によってそうすることが許されているなら、IPsec [RFC2401] のような代替案も検討すべきである。

9.5 RTP ペイロードとヘッダの完全性

SRTP メッセージは完全性と発信元の識別のために攻撃され、これらのリスクはセクション 9.5.1 で議論されている。これらの攻撃から保護するために、各 SRTP ストリームは、HMAC-SHA1 [RFC2104] によって、80 ビットの出力タグと 160 ビットのキー、または同等の強度を持つメッセージ認証コードで保護されるべきである [SHOULD] 。
安全な RTP は、このセクションで説明された状況を除いて、メッセージ認証なしで使用されるべきではない [SHOULD NOT] 。AES カ

ウンタ・モードと f8 を含む暗号化アルゴリズムはメッセージ認証を提供しないことに注意することが重要である。SRTCP は、弱い (または NULL) 認証では使用してはならない [MUST NOT] 。

SRTP は、弱い認証 (例えば、32 ビット認証タグ) 、または認証なし (NULL 認証アルゴリズム) で使用してよい [MAY] 。これらのオプションにより、SRTP を使用して、*弱い認証または NULL 認証が許容されるセキュリティリスクである場合に機密性を提供することができて、*強力なメッセージ認証を提供することは実用的ではない。

これらの条件については、セクション 7.5 で説明する。弱い認証または NULL 認証が使用されるためには、両方の条件が成立しなければならない [MUST] ことに注意する。弱いまたは拒否された認証オプションの実行に関連するリスクは、セクション 9.5.1 で説明したリスクを考慮して、特定のアプリケーションまたは環境で使用する前にセキュリティ監査によって検討する必要がある。

RTP アプリケーションが無視できる程度の偽造の影響が少ない場合は、弱い認証は許容される。アプリケーションがステートレスである場合、単一の偽造された RTP パケットの影響は、その特定のパケットのデコードに限定される。この条件の下では、認証タグのサイズは、SRTP 受信者によって RTP アプリケーションに渡されるパケットのごくわずかな部分だけが偽造であることを保証しなければならない [MUST] 。この部分は、偽造されたパケットの制御が与えられたならば、攻撃者が RTP アプリケーションの出力に大きな影響を及ぼさない場合には無視できる (セクション 7.5 の例を参照) 。

攻撃者が暗号文を修正して分かりやすい値に復号することができない場合は、脆弱または NULL 認証が受け入れられてよい[MAY]。1 つの重要なケースは、多くのコーデックでは、入力信号を知らない攻撃者が制御された方法で出力信号を操作できないため、攻撃者が RTP プレーンテキストデータを取得することが困難な場合である。

多くの場合は、攻撃者がプレーンテキストの実際の値を決定するのは難しいかもしれない。例えば、生のオーディオまたはビデオ信号を知るために、隠されたスヌープ装置が必要になることがある。攻撃者が使用するコーデックでその信号をエンコードするのに十分な情報がないため、攻撃者の信号は攻撃対象の信号と同等かそれ以上の品質を持つ必要がある。また、電話などの対話型アプリケーションでは、プレーンテキストの予測が特に困難な場合がある。

RTP アプリケーションが RTP データに基づいてデータ転送またはアクセス制御の決定を行う場合、弱い認証または NULL 認証を使用してはならない [MUST NOT] 。そのような場合、攻撃者は、受信者にデータを攻撃者に転送させることで機密性を破壊する可能性がある。このような攻撃の実際の例については、 [B96] のセクション 3 を参照。

攻撃者がパケットを保存し、セッションの後半にそれらを再生するリプレイ攻撃が受信者に無視できない影響を及ぼす可能性がある場合、NULL 認証を使用してはならない [MUST NOT] 。リプレイ攻撃の成功例は、一定期間監視カメラの出力を記憶し、その後監視を避けるためにその出力を監視ステーションに注入することである。

暗号化はこの攻撃から保護するものではなく、非 NULL 認証はそれを無効にするために必要である。存在するメッセージ偽造が問題である場合、すなわち、受信データの精度が無視できないほど重要である場合、NULL 認証を使用してはならない [MUST NOT] 。

9.5.1 脆弱または NULL メッセージ認証のリスク

弱い認証または NULL 認証の使用を考慮したセキュリティ監査では、メッセージ認証アルゴリズムを使用しない場合に可能な次の攻撃を覚えておくことが重要である。

プレーンテキストを予測できない攻撃者は、送信者と受信者の間で送信されたメッセージを常に変更して、ランダムなプレーンテキスト値に復号化することができる。あるいはランダムなプレーンテキスト値に解読するためのパケットのストリームを受信者に送ることができる。この攻撃は本質的にサービス拒否攻撃であるが、メッセージ認証がない場合、RTP アプリケーションは真の値とビット単位で関連づけられた入力を持つ。そのようなデータが有効なビデオデータとして受け入れられると、一部のマルチメディアコーデックや一般的なオペレーティングシステムがクラッシュすることがある。このサービス拒否攻撃は、攻撃者がパケットを破棄、遅延、または再注文することによるものよりもはるかに大きな脅威である可能性がある。

プレーンテキストを予測できない攻撃者は、受信者がそれを受け入れることを確実にして、以前のメッセージを再生することができる。ステートレスコーデックを使用するアプリケーションは、この種の攻撃に対して堅牢である可能性があるが、他のより複雑なアプリケーションでは、これらの攻撃はさらに深刻である。

プレーンテキストを予測できる攻撃者は、選択した任意の値に復号化するように暗号文を変更できる。加算ストリーム暗号では、攻撃者は常に個々のビットを変更することができる。

解読されたが認証されていないプレーンテキストに対してデータ転送またはアクセス制御の決定が行われた場合、攻撃者は認証の欠如のために機密性を破壊する可能性がある。これは、受信者がデータを攻撃者に転送することに騙され、間接的な機密性の侵害につながる可能性があるためである ([B96] のセクション 3 を参照) 。これは、復号化されたプレーンテキストに対してデータ転送の決定が行われるためである。プレーンテキストの情報は、ESP [RFC2401] トンネルモード (それぞれ、転送モード) でプレーンテキストがどのサブネット (またはプロセス) に転送されるかを決定する。

Secure RTP をメッセージ認証なしで使用する場合、アプリケーションが解読されたプレーンテキストに基づいてデータ転送またはアクセス制御の決定を行わないことを確認する必要がある。パディングを必要とするいくつかの暗号モード (例えば、標準暗号ブロック連鎖 (CBC)) は、あるパディングタイプが完全性でない場合に使用する際、機密性に対する攻撃に対して非常に敏感である。

攻撃 [V02] は、これが CBC モードと共に使用される場合、図 1 を参照して説明した標準的な RTP パッディングのケースであることを示している。SRTP への後での変換の追加は、適切な完全性保護なしでこのパディングを使用するリスクを注意深く考慮しなければならない [MUST] 。

9.5.2 暗黙のヘッダ認証

f8 モードの IV 形成は、メッセージ認証が使用されていなくても、RTP ヘッダの暗黙的認証 (IHA) を提供する。IHA が使用されると、RTP ヘッダの値を変更する攻撃者は、受信者の復号化プロセスでランダムなプレーンテキスト値を生成する。この保護はメッセージ認証と同等ではないが、一部のアプリケーションでは有効である。

10. 前方誤り訂正メカニズムとの相互作用 (RFC3711 セクション 10)

SRTP による前方誤り訂正 (FEC) (例えば RFC2733) 処理を使用するときのデフォルト処理は、送信側で SRTP 処理を行う前に FEC 処理を実行し、受信側で FEC 処理を行う前に SRTP 処理を実行することである [SHALL] 。 この順序付けの変更 (それを取り消す、または SRTP 暗号化と SRTP 認証の間に FEC を置く) は、帯域外で合図されなければならない [SHALL] 。

11. シナリオ (RFC3711 セクション 11)

SRTP は、さまざまなシナリオで RTP / RTCP トラヒックのセキュリティプロトコルとして使用できる。 SRTP には、特にキーの使用に関するいくつかの設定オプションがあり、アプリケーションの使用方法に応じてアプリケーションのパフォーマンス全体に影響を与える可能性がある。 したがって、SRTP の使用は、使用されるシナリオとアプリケーションの種類に依存する。 以下では、SRTP の使用例を簡単に説明し、オプションの推奨設定に関するガイドラインを示す。

11.1 ユニキャスト

典型的な例は、音声呼またはビデオオンデマンドアプリケーションである。

1 つの双方向 RTP ストリームを 1 つの RTP セッションとして考える。 両当事者は、セクション 9.1 の原則に従って 2 つの方向で同じマスターキーを共有することができる。 キー導出の第 1 ラウンドは、マスターキーを以下のいずれかまたはすべてのセッションキーに分割します (提供されたセキュリティ機能に従って) 。

SRTP_encr_key, SRTP_auth_key, SRTCP_encr_key, and SRTCP_auth key.

(簡単にするために、我々はまた、導出されているソルトの議論を省略します。) このシナリオでは、それはほとんどの場合、デフォルトの寿命を持つ単一のマスターキーを持ってすればよい。 これにより、キーの十分な長寿命と、ほとんどの実用的な目的のための最小限のキーセットが保証される。 また、この場合、RTCP 保護を円滑に適用することができる。 これらの前提の下で、MKI の使用は省略することができる。 それぞれの方向のパケットレートの大きな差と組み合わせたキー導出には複数のセッションキーの同時格納が必要な場合があるため、ストレージが問題であれば、低レートのキー導出を使用することを推奨する。

複数の RTP セッションで同じセッションをユニキャストシナリオに拡張することができる。各セッションには個別のマスターキーがある。

11.2 マルチキャスト (1 送信者)

(保護されていない) RTP の場合と同様に、送信者が処理する必要のある SRTCP 受信者レポートの可能性が非常に高いため、大きなグループでスケーラビリティの問題が発生する。 SRTP では、送信者は各受信者、より正確には受信者レポートを保護するために使用される SRTCP の状態 (暗号化コンテキスト) を保持する必要がある。 オーバーヘッドは、グループのサイズに比例して増加する。 特に、再キーイングには特別な懸念が必要である。下記を参照してください。

最初に小さなグループの受信者を考えてみます。 受信者間でマスターキーを配布すると、いくつかの設定が可能です。 単一の RTP セッションが与えられた場合、受信者はセクション 9.1 と同じマスターキーを共有して、それぞれの RTCP トラフィックをすべて保護する可能性がある。 この共有マスターキーは、発信側の SRTP トラフィックを保護するために送信側が使用するものと同じものにすることができる。 あるいは、受信者間でのみ共有され、SRTCP トラフィック専用に使用されるマスターキーでもよい。 どちらの方法でも、受信者は互いに信頼する必要がある。

SRTCP およびキーストレージを考慮すると、送信者があまりにも多くのセッションキーを格納する必要がないように、低速 (またはゼロ) キー導入 (必須の初期キーを除く) を使用することを推奨する (各 SRTCP ストリームは、 SRTCP ソースが異なる時間に送信するので、所定の時点でキーを持つ) 。 したがって、SRTP に対してキーの導出が必要な場合、SRTP の暗号コンテキストを SRTCP 暗号コンテキストとは別に保つことができるので、SRTCP にはキー導出率を、SRTP にはゼロ以外の値を持たせることができる。

再キーイングのための MKI の使用は、ほとんどのアプリケーションで推奨される (セクション 8.1 を参照) 。

マスターキーを共有する複数の SRTP / SRTCP ストリーム (同じ RTP セッション内) がある場合、2 ^ 48 SRTP パケット/ 2 ^ 31 SRTCP パケットの上限は、ストリームの 1 つが最大数に達する前に マスターキーを共有するすべてのストリームで再キーイングを開始しなければならない [MUST] 。 (厳密なセキュリティの観点からは、最大に達するストリームだけを再キーする必要があるが、ストリームはもはやマスターキーを共有しない。これは意図である。) 送信側のローカルポリシーは、 いずれのストリームにおいても最大パケット限界に達しないようにする。 再キーイングのための MKI の使用は推奨されている。

1 つの送信者を伴う大規模なマルチキャストでは、小グループマルチキャスト保留の場合と同じ考慮事項がある。 このシナリオの最大の問題は、RTCP 受信者レポートを送り返して、各受信者に維持する必要がある状態 (暗号化コンテキスト) が原因で、送信側に追加される負荷である。 少なくとも RTCP ソースごとに再生ウィンドウを維持する必要があるかもしれない。

11.3 再キーイングとアクセス制御

再キーイングは、アクセス制御 (例えば、メンバーがマルチキャスト RTP セッション中に削除されたとき) 、または純粋な暗号の理由 (例えば、そのキーがその存続期間の終わりにある) のために起こり得る。 SRTP のデフォルト変換を使用する場合、同じマスターキーで保護されているストリームのいずれかのインデックススペースが使い尽くされる前に、マスターキーを交換する必要がある。

どのようにキー管理の再キーSRTP 実装が範囲外であるかは明らかだが、マルチキャストグループのキーを管理する簡単な方法があることは明らかである。たとえば、1 つの送信者のマルチキャストでは、新しいキーがいつ必要なのかを判断するのは、通常、送信者の責任である。送信者は、受信者がいつでもセッションに参加したり離れることができるように、パケットの最大数がいつ送信されたかを追跡することができる 1 つのエンティティであり、パケット損失および遅延などが存在する可能性がある。他の方法を用いてもよい。ここでは、キー交換はコストのかかる操作であり、1 回の交換で数秒かかることを考慮する必要がある。したがって、マスターキーが使い尽くされるかまたは期限切れになる前に、帯域外キー管理が開始され、その結果、受信者と共有される新しいマスターキーが生成される。いずれにしても、新しいキーに切り替えるときに同期を維持するために、グループポリシーでは、MKI と <From, To> を使用するかどうかをセクション 8.1 で説明している。

アクセス制御の目的のために、 <From, To> 期間は、パケットレートに依存して、所望の精度で設定される。 大規模なグループシナリオでは、SRTCP では高速再キーイングが問題になる。 前述のように、SRTCP インデックスではなく SRTP インデックスを使用してマスターキーを決定する際に潜在的な問題がある。 特に、マスターキーの切り替え中に短期間、SRTCP パケットが通信相手の SRTP の現在のマスターキーの下にない場合がある。 したがって、このようなシナリオでキー入力に MKI を使用すると、より良い結果が得られる。

11.4 基本シナリオのまとめ

これらのシナリオの説明では、主に再キーイングと大規模なマルチキャストに関連する SRTP の使用に関するいくつかの推奨事項が強調されている。

- <From, To> 機能を使用して高速再キーイングを使用してはならない。 特に、SRTCP パケットが再キーイング時間の近くに到着した場合に、正しい SRTCP キーを検索する際に問題が生じる可能性がある。 この場合、MKI を使用すべきである [SHOULD] 。

- 同じ RTP セッション内の複数の SRTP ストリームが同じマスターキーを共有する場合、中程度のレートの再キーイングでも同じ問題が発生する可能性があり、MKI を使用すべきである [SHOULD] 。

- セキュリティが向上しているが、複数のストリームで使用されているキーの数を最小限に抑えようとすると、ゼロ以外のキー導出率は推奨されない。

12. IANA についての考慮事項 (RFC3711 セクション 12)

RTP 仕様では、セッション記述プロトコル (SDP) などの上位レベル制御プロトコルで使用されるプロファイル名のレジストリが転送メソッドを参照するために設定されている。 このプロファイルは「RTP / SAVP」という名前で登録する。

SRTP は、キー管理プロトコルが伝える暗号化変換を使用する。 IANA に変換の暗号化変換または変換を登録するのは、それぞれの特定のキー管理プロトコルのタスクである。 キー管理プロトコルは SRTP ではなくこれらのプロトコル番号を伝え、各キー管理プロトコルは必要な番号付けスキームと構文を選択する。

SRTP の主要な管理プロトコルの仕様はここでは範囲外である。 しかし、セクション 8.2 では、デフォルトおよび必須変換に対して定義する必要のあるパラメータについてのガイダンスが提供されている。

Appendix B テストベクトル

B.3 主なキー導出テストベクトル

このセクションでは、カウンタ・モードで AES-128 を使用するデフォルトのキー導出機能のテストデータを示す。以下では、16 オクテットのセッション暗号化キーと 14 オクテットのセッションソルトを必要とする AES-128 カウンタ・モード暗号、および 94 オクテットのセッション認証キーを必要とする認証機能の初期キー導出について説明する。以下では、これらの値を暗号キー、暗号ソルト、および認証キーと呼ぶ。これは初期のキー導出であり、キー導出率はゼロに等しいので、インデックス DIV キー導出率の値はゼロである (実際には、6 オクテットのゼロのストリング) 。

キー導出関数への入力は、16 オクテットのマスターキーと 14 オクテットのマスターソルトである。

マスターキー: E1F97A0D3E018BE0D64FA32C06DE4139
マスターソルト: 0EC675AD498AFEEBB6960B3AABE6

最初に暗号キーがどのように生成されるかを示す。 AES-CM の入力ブロックは、マスターソルトを暗号化キーラベル 0x00 とインデックス DIV キー導出率の値の連結で排他的論理和をとり、次に右側に 2 つの null オクテットでパディングすることによって生成される (2^16 の乗算操作の実装、セクション 4.3.3 を参照) 。結果の値は、マスターキーを使用して AES-CM-暗号化され、暗号キーが取得される。

インデックス DIV キー導出率の値: 000000000000
ラベル: 00
マスターソルト: 0EC675AD498AFEEBB6960B3AABE6

排他的論理和: 0EC675AD498AFEEBB6960B3AABE6 (x, PRF input)

x*2^16: 0EC675AD498AFEEBB6960B3AABE60000 (AES-CM input)

暗号キー: C61E7A93744F39EE10734AFE3FF7A087 (AES-CM output)

次に、暗号化ソルトがどのように生成されるかを示す。AES-CM の入力ブロックは、マスターソルトと暗号化ソルトラベルの連結を排他的論理和することによって生成される。その値は上記のように埋め込まれ暗号化される。

インデックス DIV キー導出率の値: 000000000000
ラベル: 02
マスターソルト: 0EC675AD498AFEEBB6960B3AABE6

排他的論理和: 0EC675AD498AFEE9B6960B3AABE6 (x, PRF input)

x*2^16: 0EC675AD498AFEE9B6960B3AABE60000 (AES-CM input)

30CBBC08863D8C85D49DB34A9AE17AC6 (AES-CM ouptut)

暗号化ソルト:　30CBBC08863D8C85D49DB34A9AE1

認証キーがどのように生成されるかを示す。 AES-CM の入力ブロックは上記のように生成されるが、認証キーラベルを使用する。

インデックス DIV キー導出率の値:　000000000000
ラベル:　01
マスターソルト: 0EC675AD498AFEEBB6960B3AABE6

排他的論理和:　0EC675AD498AFEEAB6960B3AABE6　(x, PRF input)

x*2^16:　0EC675AD498AFEEAB6960B3AABE60000 (AES-CM input)

以下に、認証キーと対応する AES 入力ブロックを記載します。

認証キー	AES 入力ブロック
CEBE321F6FF7716B6FD4AB49AF256A15	0EC675AD498AFEEAB6960B3AABE60000
6D38BAA48F0A0ACF3C34E2359E6CDBCE	0EC675AD498AFEEAB6960B3AABE60001
E049646C43D9327AD175578EF7227098	0EC675AD498AFEEAB6960B3AABE60002
6371C10C9A369AC2F94A8C5FBCDDDC25	0EC675AD498AFEEAB6960B3AABE60003
6D6E919A48B610EF17C2041E47403576	0EC675AD498AFEEAB6960B3AABE60004
6B68642C59BBFC2F34DB60DBDFB2	0EC675AD498AFEEAB6960B3AABE60005