Military Technical College
Kobry El-Kobbah,
Cairo, Egypt

7th International Conference
on Electrical Engineering
ICEENG 2010

Enhancement Packet Authentication and integrity in SRTP Protocol

By

Mazen Tawfik Mohammed* Alaa Eldin Rohiem** Ali El-moghazy***

Abstract:

The Real-time Transport Protocol (RTP) is a media transport protocol. RTP is primarily designed to satisfy of the needs for multimedia transmission [1]. The Secure Real-time Transport Protocol (SRTP) provides confidentiality, message authentication, and replay protection for RTP traffic. However, there are risks of weak message authentication in SRTP Protocol [2]. With a weak Message authentication code it is easy for attacker to modify the SRTP packets. The main purpose of this paper is to propose an alternative scheme to provide a stronger authentication and integrity. The proposed scheme is implemented using Microsoft open source project for conference. The implementation is tested using StsGui NIST Statistical suite and Cryptool software [3] for security and Wireshark [4] for performance. The test results show that the proposed modification enhances the security with minor effect on the quality of service (QoS).

Keywords:

Real-time Transport Protocol (RTP), Secure Real-time Transport Protocol (SRTP), Authentication and Integrity.

* Military technical College, Cairo, Egypt
** Military technical College, Cairo, Egypt
*** Military technical College, Cairo, Egypt

1. Introduction:

The Real-time Transport Protocol RTP is the Internet Engineering Task Force (IETF) standard [1], which is intended to provide end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video over multicast or unicast network services. RTP protocol group consists of two parts; RTP and RTCP. RTP is designed to carry data that has real-time properties, while RTCP is used to monitor the quality of service and convey information about the participants in on-going session. There are several security protocols at different layers such as Internet Protocol Security (IPSec) [5], Secure Socket Layer (SSL)/Transport Layer Security (TLS) [6], and Secure Real-Time Transport Protocol (SRTP).

TLS is not suitable for real-time traffic and is only applied to TCP connections [7], and IPSec has a high effect on QoS. However, SRTP is a good protocol for real-time traffic. SRTP is the Internet Engineering Task Force (IETF) standard [2], which provides confidentiality, message authentication, and replay protection for RTP traffic. The SRTP RFC3711 specifies AES [8] encryption of the RTP payload and a message authentication hash of the header and the encrypted payload using HMAC-SHA1 [9] to achieve enhanced security. The Real-time Transport Protocol (RTP) is susceptible to several attacks [10], including third-party snooping of private conversations, injection of forged content, and introduction or modification of packets to degrade voice quality.

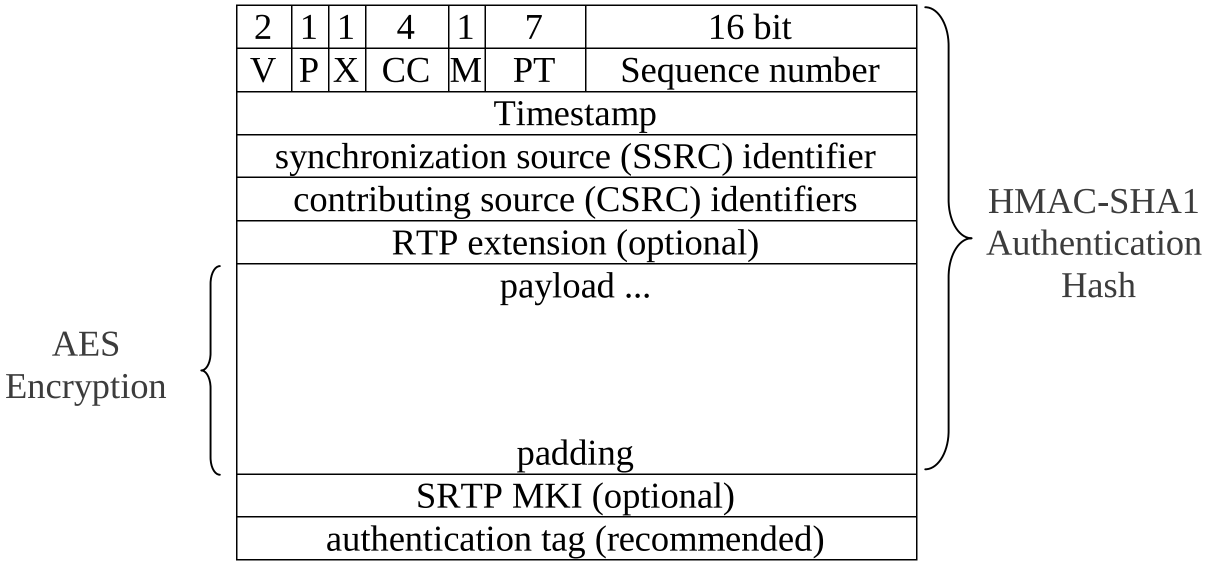
Stronger RTP authentication could be achieved using Internet Protocol Security (IPSec), the cost here is the added latency. Also it must be applied on a hop-by-hop basic. This paper introduces an alternative scheme, which could enhance the security in SRTP. The rest of this paper is organized into 4 sections. Section 2 introduces for SRTP authentication and SRTP weakness. Sections 3 explain the scope of the new scheme; Section 4 is concerned with testing the proposed implementation and comparing it with the standard SRTP, while section5 is the conclusion.

2. SRTP Overview and Weakness:

In this section we describe the SRTP packet format, processing at transmitter and receiver and weakness.

a) SRTP Packet format:

Figure (1) shows the SRTP packet format, the packet consists of header, encrypted payload and the authentication code.



2	1	1	4	1	7	16 bit
V	P	X	CC	M	PT	Sequence number
Timestamp						
synchronization source (SSRC) identifier						
contributing source (CSRC) identifiers						
RTP extension (optional)						
payload ... padding						
SRTP MKI (optional)						
authentication tag (recommended)						

Figure (1): *The format of an SRTP packet*

The header fields are:

- Version (V): indicates the version of RTP used (now version 2).
- Padding (P): indicates the padding, additional information on the header (payload encrypted case).
- Extension (X): indicates the presence of the header extension.
- CSRC Count (CC): The contributing source (CSRC) count contains the number of CSRC identifiers that follow the fixed header.
- Marker (M): It is intended to allow significant events such as frame boundaries to be marked in the packet stream.
- Payload (PT): this field identifies the format of the RTP payload.
- Sequence number: the sequence increments by one for each RTP data packet sent, and may be used by the receiver to detect packet loss, and to restore packet sequence. The initial value of the sequence number should be random.
- Timestamp: contains the time SRTP data are sent and is used to overcome jitter and synchronization problems.
- Synchronization source SSRC identifier: This identifier should be chosen randomly, is used to distinguish each RTP stream from another on the same session (overcome RTP stream Conflict).
- Master Key Identifier (MKI): The MKI identifies the master key from

which the session keys were derived (the MKI is optional).

- Authentication tag: configurable length, recommended. The authentication tag is used to carry message authentication data for RTP header and the encrypted portion of the SRTP packet.

Figure (1) shows that the entire header and encrypted payload are hashed via the HMAC-SHA1 algorithm to produce message authentication code; the authentication code must be computed for each packet.

b) SRTP Packet Processing:

The default defined authentication mechanism for SRTP is HMAC-SHA1. The authentication tag (code) is computed by the sender and appended to the data. The receiver side computes the authentication tag using the algorithm defined in the cryptographic context and compares it to the tag of the received message. The data is authentic if both the tags are valid otherwise it is invalid.

b.1) On sender side:

Figure (2) shows SRTP packet processing on sender side

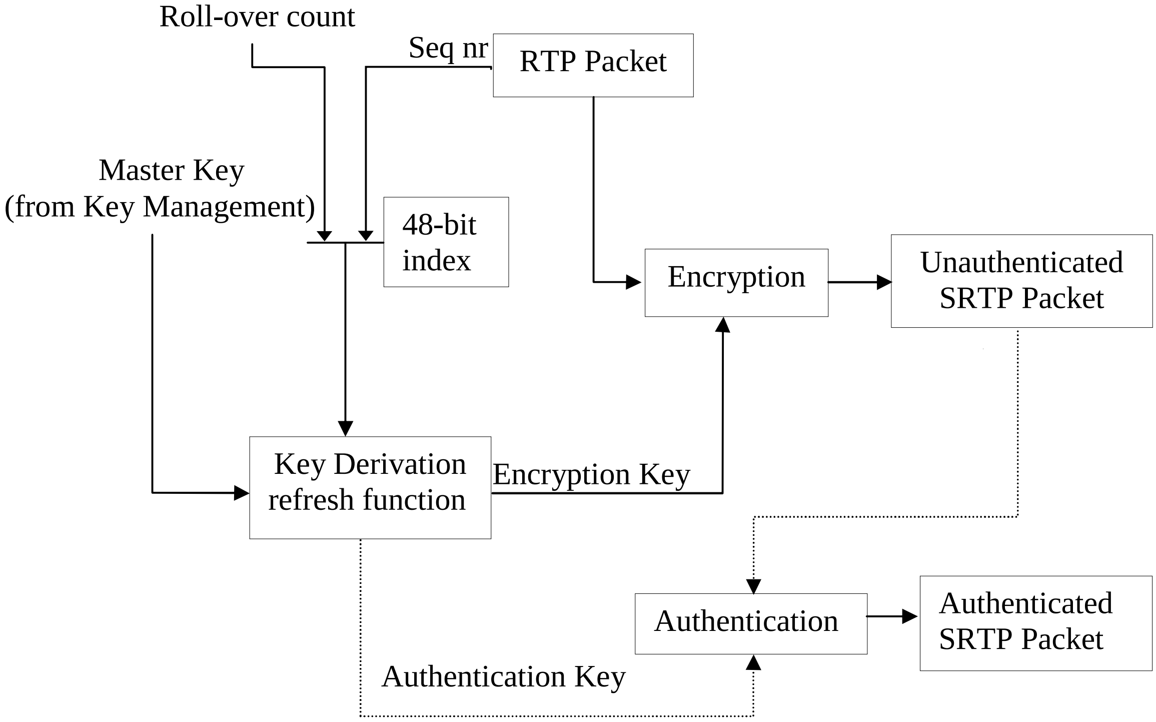


Figure (2): *SRTP packet processing on sender side*

Figure (2) shows that the sender determine the index of the SRTP packet using the rollover counter and the sequence number, then encryption key and authentication key are derived using the index and the master key (and master salt), then encrypt the RTP payload, for message authentication, compute the authentication tag for the encrypted portion of the packet, and then append the authentication tag to the packet. If necessary, update the ROC and send the packet.
Rollover counter is a 32 bit length, which records the number of times the sequence number has been reset to zero after passing through 65,535.

b.2) On receiver side:

Figure (3) shows SRTP packet processing on receiver side

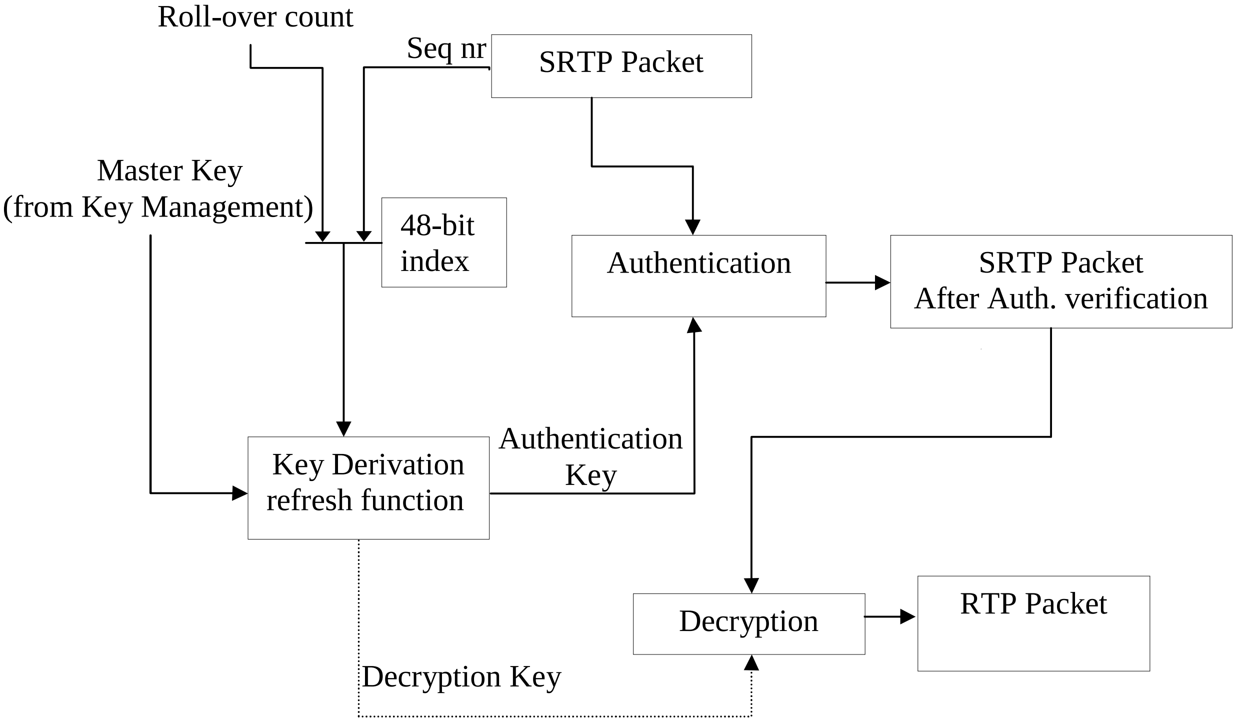


Figure (3): *SRTP packet processing on receiver side*

Figure (3) shows that in the receiver side the packet index is calculated using the rollover counter and sequence number, and then encryption key and authentication key are derived using the index and the master key. For message authentication, perform verification of the authentication tag; and then decrypt the encrypted portion of the packet. If necessary, update the ROC.

c) SRTP weakness:

In SRTP a full 80-bit authentication-tag should be used, and a shorter tag or even a zero-length tag (null message authentication) may be used under certain conditions. This causes several risks in SRTP protocol [RFC3711].

An attacker who cannot predict the plaintext is still able to modify the message sent between the sender and the receiver, by modifying the payload and /or header then recalculating the corresponding tag. The receiver can't detect this integrity violation.

3. The Proposed Authentication and Integrity Scheme:

In this section we describe the proposed scheme for enforcing SRTP authentication and integrity.

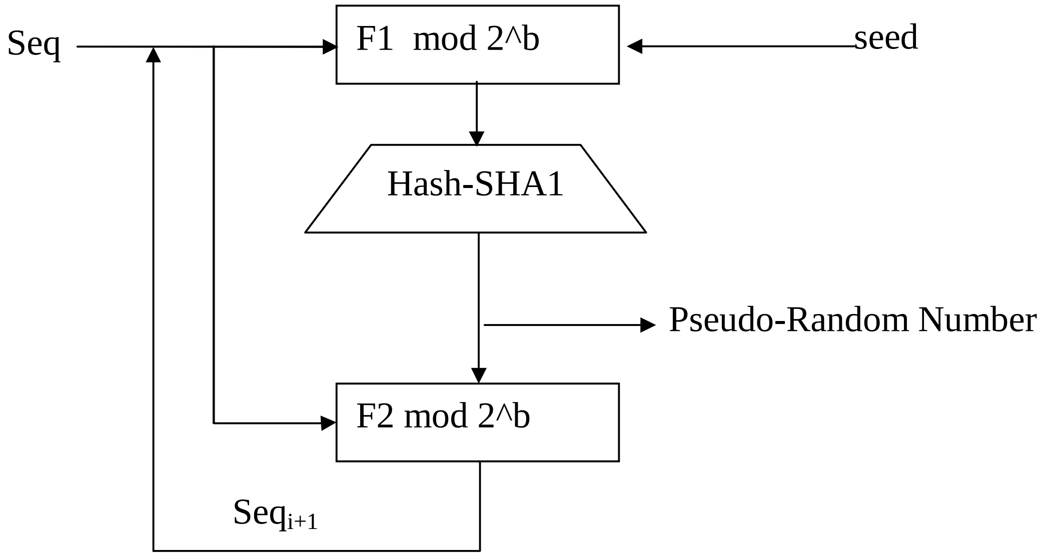
a) Basic Idea Of Proposed Scheme:

The basic Idea of proposed scheme is to make the authentication code unpredictable.

Both endpoints must generate the pseudorandom number (PRNG) based on sequence number and seeds. Before any SRTP packet is transmitted, the seed for the PRNG is exchanged in a secure manner and then authentication code is inserted to each SRTP packet Payload at a random position according to the output of PRNG. The authentication code will be encrypted with payload using AES algorithm, the position of authentication code will not be known to attacker. This enforces integrity without affecting confidentiality.

b) Pseudo Random generator:

Pseudo-Random Number Generators: A pseudo-random number generator, or PRNG, is a random number generator that produces a sequence of values based on a seed and a current state. Given the same seed, a PRNG will always output the same sequence of values. The theoretical considerations behind the choice of the pseudorandom number generator (PRNG) are a good randomness and fast generation. The Pseudo-Random Number Generator must be the FIPS-140 [11] approved. Figure (4) shows the used algorithm for pseudo-random number generator based on cryptographic hash [12].



Where :

Seq: sequence number of RTP packet.
b=160.
F1=(Seq XOR Seed).
F2=(PRN +1).

***Figure (4)**:Pseudo-random number generator*

c) Packet Processing In the Proposed Scheme:

The encryption and authentication process take place as define below:

c.1) on the sender side:

1) Determine the sequence of the SRTP packet.
2) Generate random number based on sequence number of the SRTP Packet and the seed, then calculate x=PRN mod (payload size - authentication code size).
3) Generate MAC-SHA1 using authentication key.
4) Insert the MAC-SHA1 in the payload. The position of MAC-SHA1 for plain text will be defined from step (2).
5) Encrypt the mixed output using AES algorithm

Figure (5) shows packets processing on the sender side.

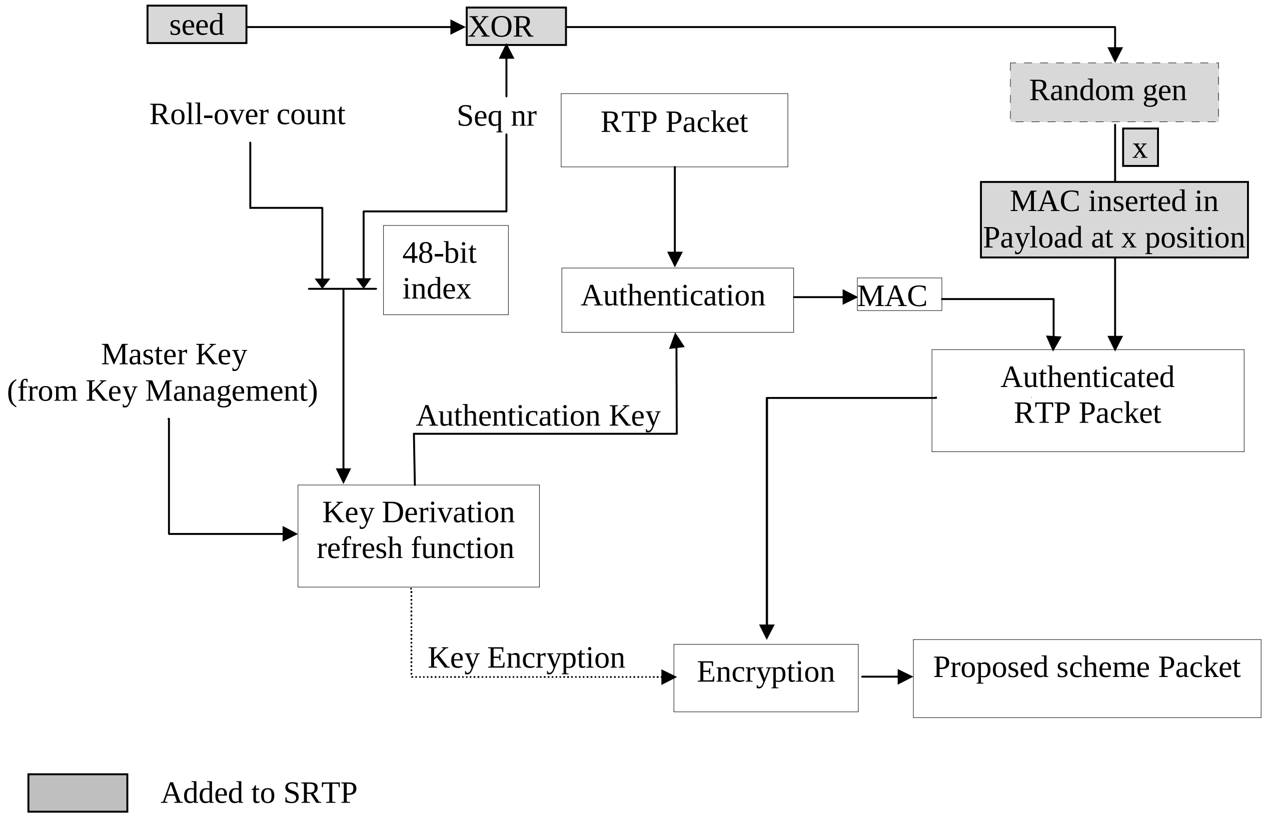


***Figure (5)**:Packets processing on the sender side*

c.2) on the receiver side:

On the receiver side the steps will performed vice versa.

1) Determine the sequence of the received SRTP packet.
2) Decrypt the mixed output (payload and authentication code) using AES algorithm.
3) Generate pseudo-random number based on sequence number of the received SRTP Packet and the seed, then calculate x=PRN mod (payload size - authentication code size), which define the position of MAC-SHA1.
4) Isolate the MAC-SHA1 from plain text.
5) Generate MAC-SHA1 for plain text (header and payload).
6) Compare the received MAC-SHA1 with generated MAC-SHA1 to check integrity.

Figure (6) shows packets processing on the receiver side.

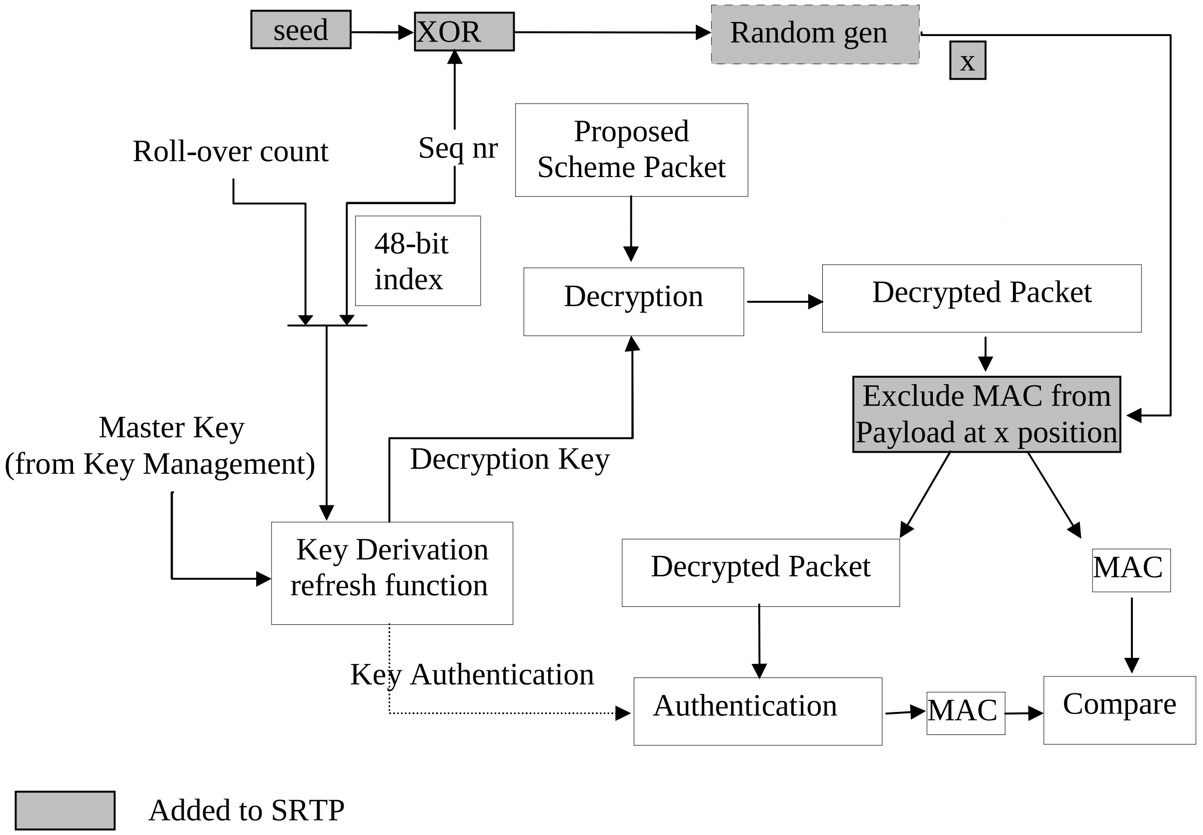


***Figure (6)**:Packets processing on the receiver side*

4. Security and Performance Analysis of Proposed Scheme:

a) Test Bed Configuration:

The test bet is composed of two endpoints the first endpoint consists of 2.16 GHz (CPU) with 3 Gbytes RAM and 1 Giga Ethernet for network card, running Windows Vista operating system. The second endpoint consists of 3 GHz (CPU) with 1 Gbytes RAM and 1 Giga Ethernet for network card, running Windows Xp operating system.

The proposed scheme is implemented by Visual C#. net 2008. The proposed scheme use two buffers, one for send (or receive) and the second for processing. The Payload size used during the test is 1440 bytes.

b) ***Statistical analysis:***

The implemented scheme is test by using (StsGui) suite and Cryptool software to perform the statistical tests. The results are shown here after.

b.1) Statistical test for Pseudo-Random Generator:

The statistical NIST tests required for Pseudo-Random Number Generator are passed.

b.2) Statistical test for Payload:

The statistical NIST parameterized and non-parameterized tests for payload are passed.

c) ***Quality of service (QoS) analysis:***

The performance in terms of quality of service (QoS) for real-time applications is measured on the basic of the delays. The total acceptable delay for a VoIP packet is 150ms [13].

The delays are classified as follows:

c.1) The Call setup delay (T_{cs}) that happens before the actual call and the call setup delay generally consists of:

a. The signaling delays caused by signaling protocol.
b. Initial key exchange delay.

c.2) The delays during call.

The delays possible during call are:

a. The encryption delay (T_{enc}).
b. The authentication delay (T_{aut}).
c. The decryption delay (T_{dec}).
d. The network delays (T_{net}).

The Per-Packet delay for SRTP calculated as below:

Call Delay = $T_{enc}+2*T_{aut}+ T_{dec}+ T_{net}$

The Per-Packet delay for Proposed scheme calculated as below:

Call Delay=Per-Packet delay for SRTP + Pseudo-Random generator delay

The delay of pseudorandom number generator depends on the used algorithm, shown in Figure (4). Now we will compare the per-packet (encryption & decryption & authentication) delays caused when using SRTP and the proposed authentication scheme. The other delays are fixed. Table (1) shows the total delay for five samples, each sample is 500 SRTP packets, and the table (2) shows the total delay for five samples, each sample is 500 Proposed scheme packets. The test is done using Wireshark software.

Table (1): *The total delay for SRTP*

	SRTP sample					Average
	1	2	3	4	5	
Max delay (ms)	143.09	142.07	141.37	141.13	142.32	141.996
Max Jitter (ms)	465.31	466.65	465.76	465.47	465.36	465.71
Mean Jitter (ms)	449.30	448.59	449.41	448.42	449.55	

Table (2): *The total delay for proposed scheme*

	Proposed Scheme sample					Average
	1	2	3	4	5	
Max delay (ms)	145.08	141.72	141.61	141.34	142.16	142.382
Max Jitter (ms)	466.97	465.66	465.29	465.26	465.42	465.72
Mean Jitter (ms)	449.34	449.94	449.31	448.41	448.41	

The minor increase in delay for proposed scheme is due to the time needed for generating Pseudo-Random Number.

5. Conclusions:

This paper presents an alternative scheme to provide a strong SRTP authentication and integrity. The proposed system is based on "hiding" the authentication code in the payload of packet, The test results show that the proposed modification enhances the security with minor effect on the quality of service (QoS).

References:

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

[2] M. Baugher, D. McGrew, E. Carrara, M. Naslund, and K. Norrman, The Secure Real-time Transport Protocol, IETF RFC 3711, March 2004.

[3] Educational Tool for Cryptography and Cryptanalysis, for Windows, http://www.cryptool.com.

[4] The Network Protocol Analyzer for Windows and Unix, June 2008, http://www.wireshark.org.

[5] S. Kent, R. Atkinson, Security Architecture for the Internet Protocol, IETF RFC 2401, November 1998.

[6] T. Dierks, E. Rescorla, The Transport Layer Security (TLS) Protocol, IETF RFC 4346, April 2006.

[7] Kuhn D., Walsh T., and Fries S., Security Considerations for Voice Over IP Systems, Recommendations of the National Institute of Standards and Technology, ITU publications, June 2005.

[8] National Institute for Standards and Technology (NIST), Advanced Encryption Standard (AES), FIPS Pub 197, November 2001.

[9] H. Crawczyk, M. Bellare, and R. Canetti, HMAC: Keyed-Hashing for Message Authentication, IETF RFC 2104, February 1997.

[10] M.Adams, M.Kwon, Vulnerabilities of the Real-Time Transport (RTP) Protocol for Voice over IP (VoIP) Traffic, Proceedings of IEEE Consumer Communications and Networking Conference. Held at Harrah's Las Vegas, Nevada: 9- 12 January 2009.

[11] National Institute for Standards and Technology (NIST), Security Requirements for Cryptographic Modules, FIPS Pub 140-2, May 2001.

[12] National Institute for Standards and Technology (NIST), Digital Signature Standard (DSS), FIPS Pub 186-2, January 2000.

[13] B. Goode, Voice Over Internet Protocol (VOIP). Proceedings of the IEEE, VOL. 90, NO. 9, Sept. 2002.