2009 Third International Conference on Network and System Security

An Evaluation of Secure Real-time Transport Protocol (SRTP) Performance for VoIP

Andre L. Alexander, Alexander L. Wijesinha, and Ramesh Karne
Department of Computer & Information Sciences
Towson University
Towson, MD, USA
awijesinha@towson.edu

Abstract—**The Secure Real-Time Transport Protocol (SRTP) is an Internet standards-track security profile for RTP used to provide confidentiality, integrity and replay protection for RTP traffic. We study the performance of SRTP when it is used to secure VoIP conversations. Experiments are conducted using snom and Twinkle softphones running on Windows and Linux platforms respectively and a bare PC softphone running with no operating system installed to provide a baseline. Pre-defined SRTP transforms based on AES counter mode encryption with a 128-bit key and HMAC-SHA-1 with a 32-bit authentication tag, as well as 192 and 256-bit AES keys and an 80-bit authentication tag are tested. Measurement of internal processing times for each operation in the SRTP protocol indicates that authentication processing is more expensive than encryption regardless of key or tag size. A comparison of jitter and delta (packet inter-arrival time) for secured and unsecured VoIP traffic reveals that the addition of SRTP protection to VoIP traffic over RTP has a negligible effect on voice quality. VoIP throughput with SRTP is about 2% more than with RTP alone since the insignificant increase in delay is offset by the small increase in packet size.**

Keywords- SRTP; VoIP; Performance; Security; Softphone

I. INTRODUCTION

VoIP is now used extensively by businesses, campus networks and individuals for low-cost communication. VoIP performance is affected primarily by network delay, jitter (delay variation), and packet loss, excessive levels of which may degrade voice quality. On the Internet for example, queuing delays at routers may increase network delay and jitter and cause packets to be dropped. However, even under ideal network conditions, intrinsic processing delays and jitter introduced at end devices such as phones and gateways can also impact VoIP performance. In particular, the additional overhead due to securing VoIP conversations may have an adverse effect on performance and voice quality.

The primary security considerations for VoIP are encryption of voice conversations, authentication and integrity of voice data, and protection against replay attacks. SRTP (Secure Real-time Transport Protocol) is an Internet standards-track profile of RTP (often used over UDP to carry VoIP data) that addresses these security aspects. We study the performance of SRTP when it is used for VoIP security. Our experiments are conducted using Windows-based snom, Linux-based Twinkle, and bare PC softphones, which do not have an operating system. The goal is to determine the overhead due to SRTP and its impact on VoIP performance.

We use the softphones to make calls in a small test LAN with no other traffic, with and without SRTP. We determine values of intrinsic jitter and delay (measured by values of packet inter-arrival time or delta), and VoIP throughput (there was no packet loss). The experiments are conducted using pre-defined SRTP cryptographic transforms with a 128-bit AES counter mode encryption key and a 32-bit HMAC-SHA-1 authentication tag. For the bare PC softphone, we are additionally able to obtain values of call quality measures for different sizes of the AES encryption key and authentication tag. We also compare internal timings for individual SRTP operations using the bare PC softphone. Our main results are the following: 1) SRTP overhead has little or no effect on voice quality regardless of AES encryption key and authentication tag sizes; 2) authentication is more expensive than encryption in terms of processing time; and 3) VoIP throughput increases by 2% with SRTP due to the few extra bytes of authentication tag i.e., intrinsic delays due to SRTP are insignificant.

II. SRTP OVERVIEW

SRTP is a profile of RTP that is designed to provide security for RTP and its control protocol RTCP with low overhead [1]. It can be used for encryption, message authentication/integrity and replay protection of RTP and RTCP traffic. While SRTP mandates message authentication for RTCP and adds new fields to an RTCP packet, we do not consider SRTP performance with respect to RTCP in our study since the overhead due to securing the periodic but infrequent RTCP messages is negligible. The pre-defined cryptographic transforms for

978-0-7695-3838-9/09 $26.00 © 2009 IEEE
DOI 10.1109/NSS.2009.90

SRTP are AES in counter mode or f8 mode for encryption and HMAC-SHA-1 for message authentication. SRTP encryption, which precedes authentication, consists of generating a pseudo-random keystream for each RTP packet and XORing the RTP data (excluding the RTP header) with the keystream.

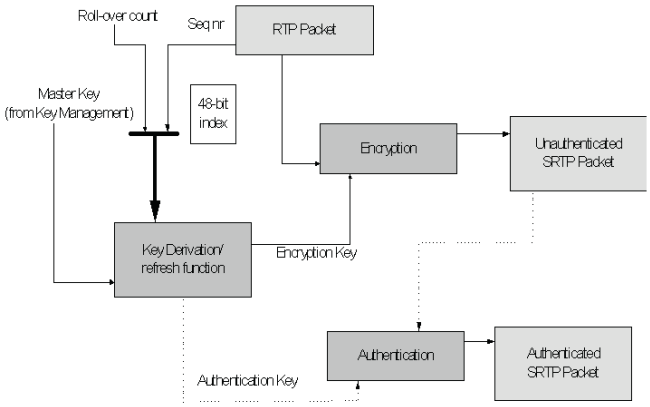

Figure 1. SRTP processing

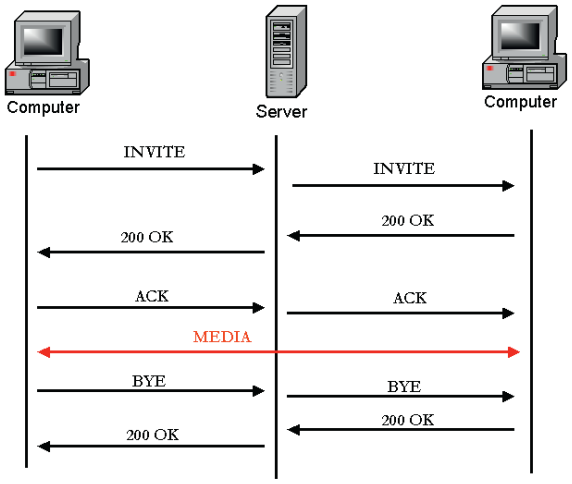

Figure 2. SIP message exchange

In AES counter mode encryption, which is employed by the softphones we use, successive 128-bit integers formed by unit increments of the IV are AES encrypted with the session encryption key and concatenated to form the keystream segment. The IV depends on the session salting key, the SSRC from RTP and the packet index. SRTP processing requires an implicit index for packets derived from the 16-bit RTP sequence number and a 32-bit rollover counter (ROC), which indicates the number of times the RTP sequence number has wrapped around due to reaching its maximum value. Session keys for encryption and message authentication/integrity protection are derived from a single master key using the SRTP key derivation function. SRTP also enables periodic refreshing of session keys and uses salting keys to enhance security. Replay protection requires that the receiver maintain a replay list and window. Fig. 1 shows the main steps in SRTP processing.

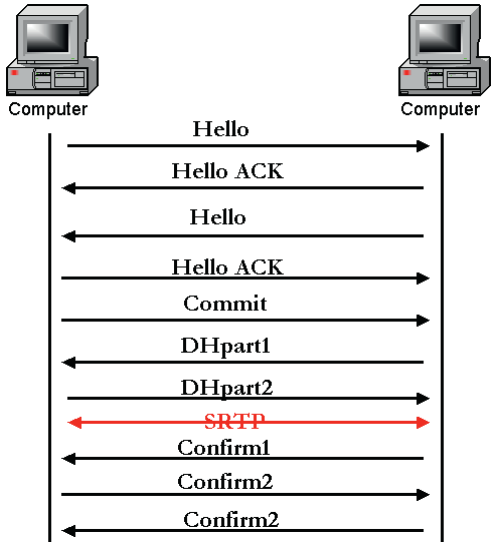

Figure 3. ZRTP message exchange

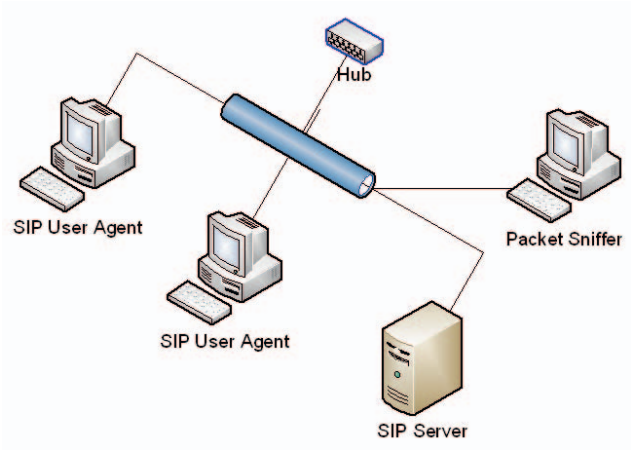

Figure 4. Test LAN

III. KEY MANAGEMENT

Secure VoIP calls require the exchange and management of keys for protection of the media sessions. The SRTP specification provides guidelines for selection of a key management system and mentions several standards but does not mandate a particular system. A variety of key exchange protocols are currently used by applications/providers with SRTP including ZRTP [2],

SDES [3], MIKEY [4] and TLS [5]. In our experiments, the snom and bare PC softphones use SDES/SIP, and the Twinkle softphone uses ZRTP for key exchange.

Since key exchange and management are not a focus of this study, we will only give a brief overview of the SDES/SIP and ZRTP message exchanges. The SDES/SIP message exchange to set up a secure VoIP call is shown in Fig. 2. The messages are the same as for a normal SIP INVITE exchange, except that they also include exchange of the master and master salt keys and cryptographic transforms via SDES utilizing the SDP offer/answer model. Since SDES uses the inline tag within SDP, the latter does not require any protocol modifications. Although used in some softphones, the SDES key exchange in this form is insecure since the SIP packets are sent in the clear. This problem can be addressed by using a TLS handshake over TCP (or DTLS over UDP) to protect the SDES key exchange over SIP/SDP. Fig. 3 shows the ZRTP message exchange. ZRTP provides a tag within the SDP protocol for notification to the client that it is able to support ZRTP. It then utilizes the media channel of the VoIP call for key establishment. Compared to SDES/SIP, ZRTP requires 5 extra packets, which are sent over RTP, with an average size of 201 bytes.

IV. Related Work

Previous and current work on SRTP primarily focuses on key exchange methods and ways to address drawbacks of the protocol. In [6], the requirements for a protocol that manages keys and parameters for SRTP and interoperates with SIP are described. Furthermore, several existing approaches including SDP security descriptions, MIKEY, ZRTP and DTLS-SRTP, an extension of DTLS to manage keys in SRTP, are compared. In [7, 8], the vulnerability of SRTP to denial-of-service flooding due to the high overhead of HMAC-SHA-1 authentication is addressed and an alternate lightweight authentication scheme SRTP+ is proposed. In [9], security protocols for VoIP and their impact on call quality are examined by measuring the mean opinion score (MOS). Our study differs from the latter in that we 1) compare jitter and delta values with and without SRTP using snom, Twinkle and bare PC softphones; and 2) determine the time for the various internal operations in SRTP using a bare PC softphone.

V. Experiments

The test LAN used for experiments consists of four Dell Optiplex GX-260 PCs with a 2.4 Ghz processor and 1 GB memory connected to a 100 Mbps Ethernet as shown in Fig. 4. The softphones contain SIP user agents and SRTP. We use a snom softphone v5.3 [10] running Windows XP SP2, a Twinkle softphone version 1.4.2 [11] running Linux Ubuntu 8.04 kernel 2.6.24-16, and a bare PC softphone [12] with no operating system. The latter serves as a baseline for comparing VoIP performance and measuring times associated with various internal SRTP operations. The OpenSER 1.3.4-1 SIP server [13] is used to register user agents and set up (proxy) VoIP calls between the softphones. The Wireshark 1.0.3 packet sniffer [14] captures packets, displays message exchanges and reports performance data.

VoIP call quality with and without SRTP is determined by obtaining delta (packet inter-arrival time) and jitter values from Wireshark. These values were computed based on 10,000 VoIP packets transferred in each direction between the softphones (i.e., about 3.5 minutes of voice traffic). The softphones used SRTP with a 128-bit AES encryption key and a 32-bit HMAC-SHA-1 message authentication tag. The bare PC softphone implementation of SRTP also allowed 192-bit and 256-bit encryption keys and an 80-bit authentication tag.

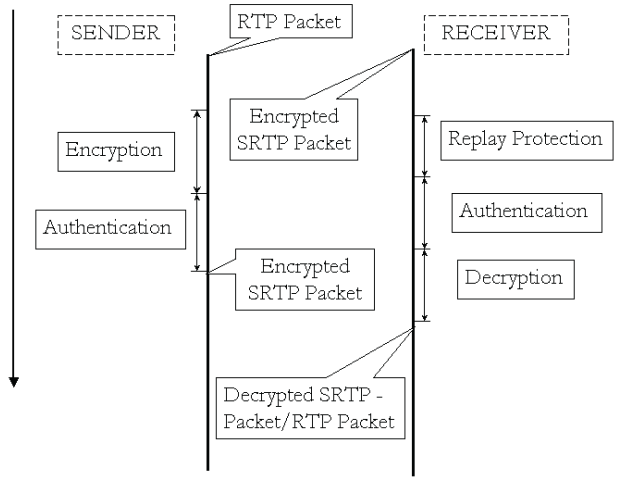

Figure 5. SRTP timing points

Timing points as shown in Fig. 5 were inserted into the SRTP code on the bare PC softphone to get the processing times of major functions in SRTP including key derivation, encryption, decryption, replay protection and authentication. Key derivation produces the session encryption, authentication, and salting keys, while encryption and decryption use AES in counter mode as described earlier. To prevent replay attacks, the receiver checks the index of each packet using a replay list of processed RTP packets within a window of size 64. Packets are authenticated by using HMAC-SHA-1 with a 160-bit key and the result is truncated to obtain an 80-bit or 32-bit authentication tag that is appended to the packet. We also measured the time to process network headers in incoming and outgoing SRTP packets i.e., the time to transfer packets between the Ethernet and SRTP processing levels.

VI. Results

The results below are for SRTP with all softphones using the G.711 codec and 20 ms voice packets consisting of 160 bytes. Since AES processes 16-byte blocks at a time, a total of 10 processing loops are necessary for each voice packet.

A. *Processing Times*

The processing times for various internal SRTP functions on the bare PC softphone with 128, 192, or 256-bit AES keys and a 32 or 80-bit HMAC/SHA-1 authentication tag are shown in Figs. 6-11. The most expensive internal step in the SRTP protocol is authentication processing. In contrast, the encryption and decryption processes consume much less time. It can also be seen that the times for the key derivation and replay processing steps are negligible. However, processing network headers on outgoing packets has higher cost than any of the other steps.

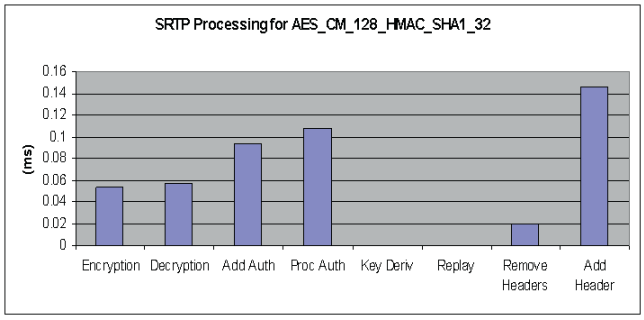

Figure 6. SRTP timing for 128-bit encryption key and 32-bit authentication tag

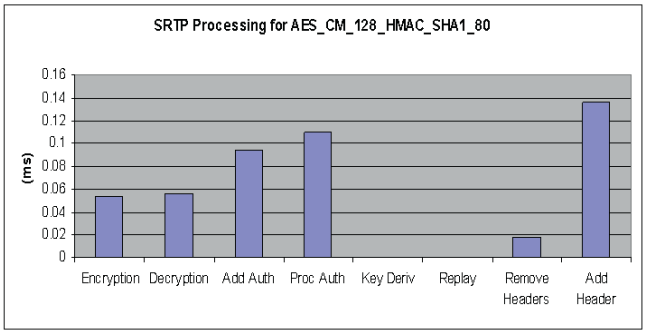

Figure 7. SRTP timing for 128-bit encryption key and 80-bit authentication tag

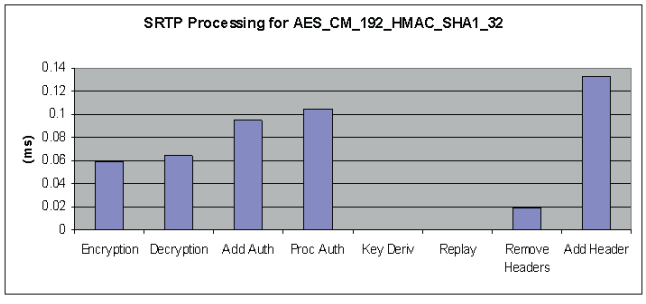

Figure 8. SRTP timing for 192-bit encryption key and 32-bit authentication tag

Processing time increases by 10% when using a 192-bit AES key versus a 128-bit key, and by 20% when using 256-bit AES key versus a 128-bit key. However, since the actual amount of processing time for all AES key sizes is very small, key size has no observable effect on call quality or VoIP throughput as is confirmed by the results in the next section. It can also be seen that processing times are about the same regardless of authentication tag size. This is because 160 bits are produced by HMAC-SHA-1 prior to truncating to a 32-bit or 80-bit authentication tag and the increase in processing time to compare the larger tag is insignificant compared to the nearly constant processing time of HMAC-SHA-1. Overall, the results clearly indicate that SRTP processing adds negligible overhead (less than 1 ms) to RTP processing.

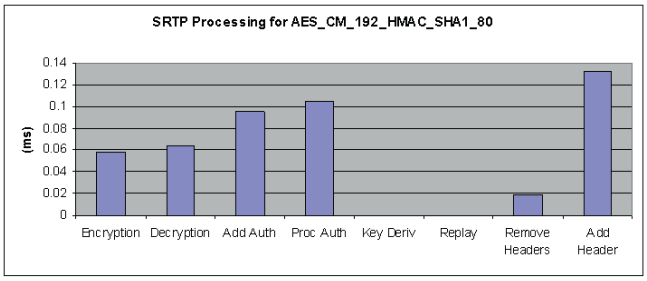

Figure 9. SRTP timing for 192-bit encryption key and 80-bit authentication tag

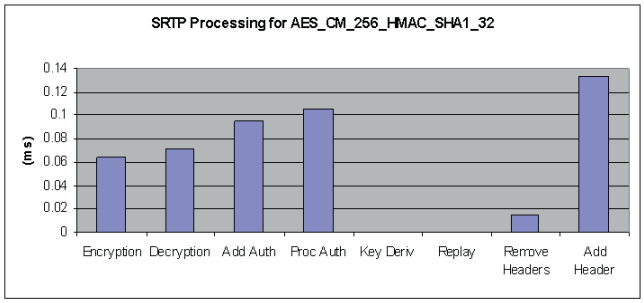

Figure 10. SRTP timing for 256-bit encryption key and 32-bit authentication tag

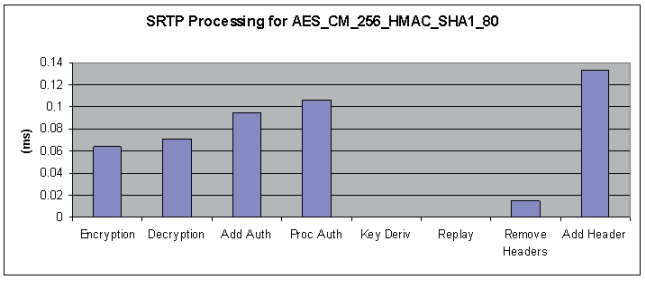

Figure 11. SRTP timing for 256-bit encryption key and 80-bit authentication tag

B. VoIP Performance

VoIP performance with and without SRTP was evaluated by comparing maximum and mean delta as well as maximum and mean jitter values on the snom, Twinkle, and bare PC softphones. These experiments used a 128-bit AES key and a 32-bit authentication tag.

Maximum delta values are shown in Fig. 12. Maximum delta without security is close to the ideal 20 ms value for the bare PC softphone and 30 ms for the snom and Twinkle softphones. However, while the increase in maximum delta due to SRTP is less than 1 ms for the snom and bare PC softphones, it is over 40 ms for the Twinkle softphone. This increase in maximum delta for the Twinkle softphone is likely due to ZRTP exchanging its keys in the media channel. Mean delta values for all three softphones with SRTP (Fig. 13) are close to 20 ms.

Maximum and mean jitter values are shown in Figs. 14 and 15 respectively. For the snom softphone, maximum or mean jitter with or without SRTP is the same (13 ms). For the Twinkle softphone, maximum and mean jitter is 5 ms and 4 ms without security, and increases by 6 ms and 2 ms respectively with SRTP. Again, this performance drop in the Twinkle softphone is possibly due to the effects of ZRTP using the media channel. In contrast, maximum and mean jitter for the bare PC softphone with or without SRTP is close to zero.

The above results for the bare PC softphone indicate that its streamlined processing of voice packets is able to reduce intrinsic delay and jitter with or without SRTP. Yet it is also evident that since delta and jitter values for all three softphones are within generally accepted limits, SRTP overhead has little or no effect on VoIP performance.

We also tested SRTP interoperability and VoIP performance when communicating between different softphones. This was done by measuring maximum delta, and maximum and mean jitter values on the respective softphones for calls between a snom softphone and a bare PC softphone using a 128-bit AES key and a 32-bit authentication tag. Maximum delta and maximum and mean jitter values with or without SRTP for bare PC to snom calls are shown in Figs. 16-18 and can be compared with the corresponding values in Figs. 12, 14, and 15 respectively.

Maximum delta for the voice packet stream from the snom softphone is the same with or without SRTP but double that for snom to snom calls. However, maximum delta values for the stream from the bare PC softphone with or without SRTP are not significantly different compared to bare PC to bare PC calls. Maximum jitter values with or without SRTP are also the same but slightly higher for the stream from the snom softphone compared to snom to snom calls, but again, differences in maximum jitter values for the stream from the bare PC softphone are very small. Mean jitter values with or without SRTP for the stream from each softphone are unchanged for bare PC to snom calls. The increased values of maximum delta and maximum jitter for the stream from the snom softphone are possibly due to the difference in timing between the softphones when processing voice packets. More studies are needed to investigate these timing differences.

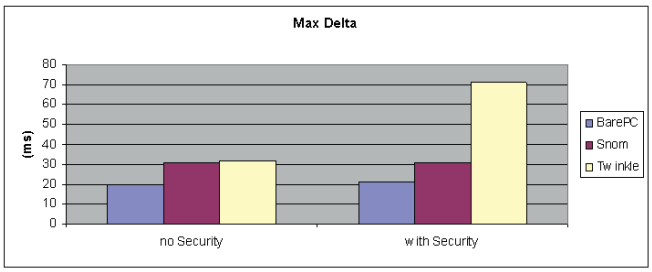

Figure 12. Maximum delta with/without SRTP

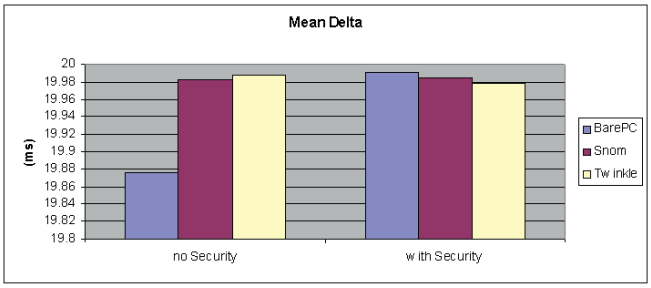

Figure 13. Mean delta with/without SRTP

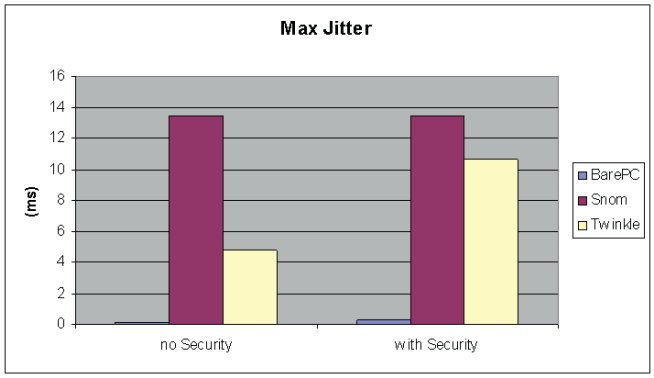

Figure 14. Maximum jitter with/without SRTP

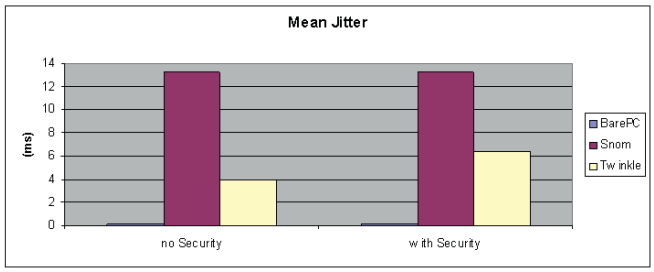

Figure 15. Mean jitter with/without SRTP

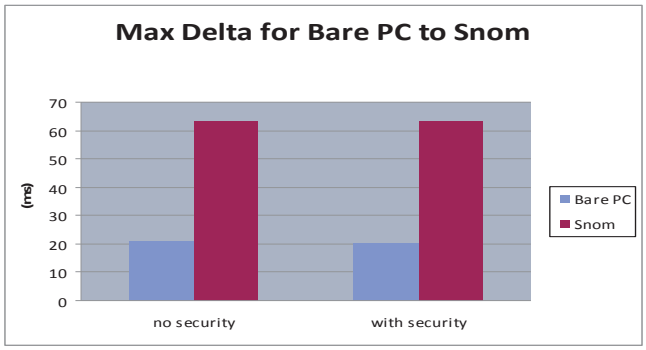

Figure 16. Maximum delta for bare PC to snom with/without SRTP

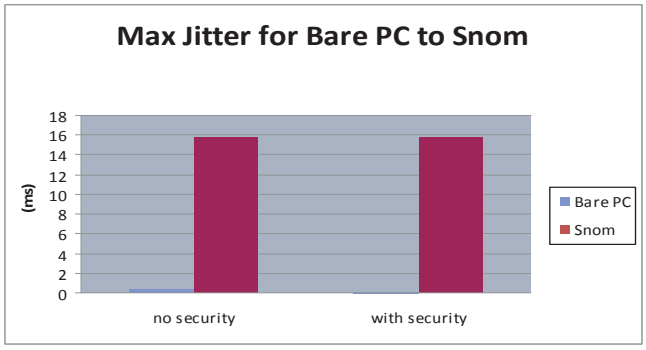

Figure 17. Maximum jitter for bare PC to snom with/without SRTP

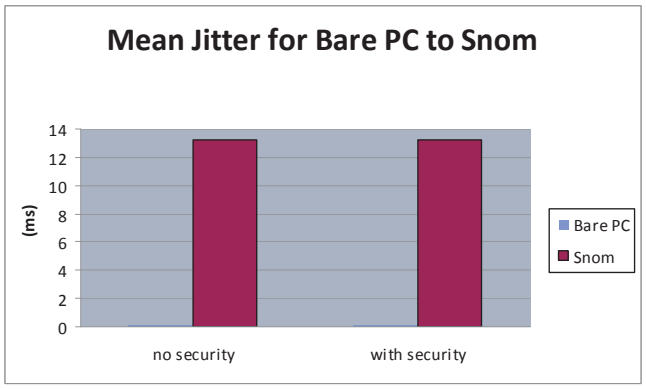

Figure 18. Mean jitter for bare PC to snom with/without SRTP

To evaluate the impact on VoIP performance with SRTP due to changing the AES key size, we measured maximum delta, and maximum and mean jitter values on a bare PC softphone with 192-bit or 256-bit AES keys and a 32-bit authentication tag (we were unable to test the snom softphone as it did not appear to support alternate AES key sizes). The results are compared with those for 128-bit AES keys (and a 32-bit authentication tag) in Figs. 19-21. The values of maximum delta and maximum jitter show little variation, and do not seem to have a simple relation to key size (the 192-bit key size has the best values and the least variation but the differences are very small). Also, the results for the two softphones are not identical. However, mean jitter is nearly constant for both bare PC softphones regardless of key size. Since the processing overhead for all authentication tag sizes is the same as explained above, the results using an 80-bit authentication tag would not be significantly different.

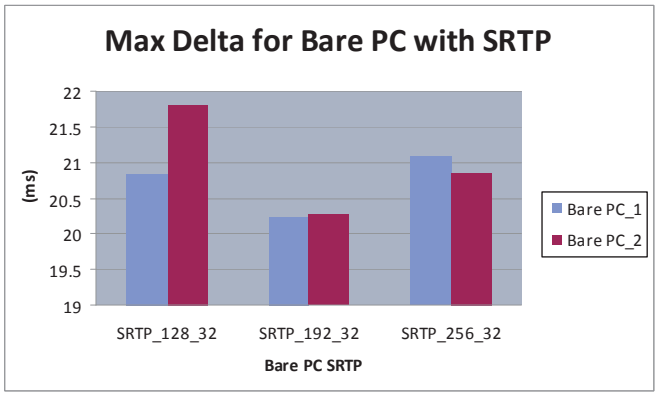

Figure 19. Maximum delta for various AES key sizes and a 32-bit authentication tag

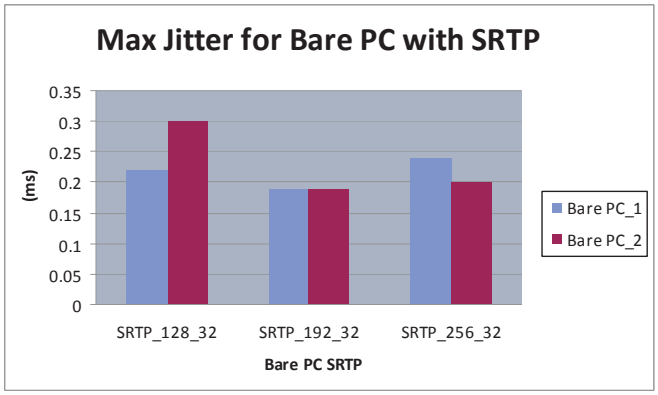

Figure 20. Maximum jitter for various AES key sizes and a 32-bit authentication tag

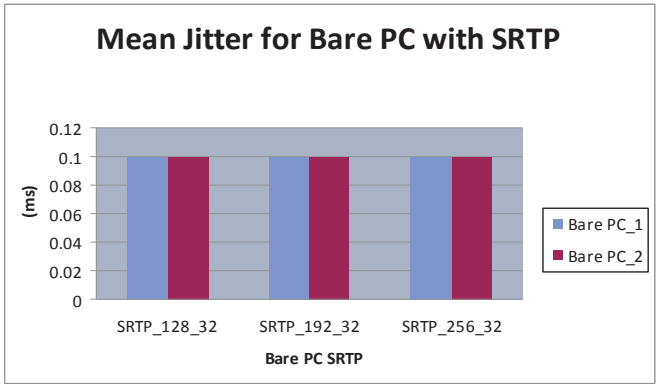

Figure 21. Mean jitter for various AES key sizes and a 32-bit authentication tag

VoIP throughput for all three softphones without SRTP is 81.6 kbps without SRTP, and 83.23 kbps with SRTP

when using a 128-bit AES key and a 32-bit authentication tag. Since SRTP encryption does not increase the size of the voice packet, the only increase in size is due to the 32-bit (or 80-bit) authentication tag. In an Ethernet, the total packet size including all network headers but excluding the CRC is 214 bytes without SRTP, and 218 bytes (or 224 bytes) with SRTP. Thus, the 2% increase in throughput with SRTP in our case simply reflects the 4-byte increase in packet size due to the authentication tag i.e., the increase in processing time due to SRTP is negligible and does not alter the throughput. Furthermore, all three softphones have the same throughput since their mean delta values are the same.

VII. CONCLUSION

We studied VoIP performance with SRTP using snom, Twinkle and bare PC softphones. Jitter and packet inter-arrival times (delta) with and without SRTP for these softphones, and internal processing times for SRTP operations on the bare PC softphone were measured. Processing overhead due to SRTP authentication is expensive compared to AES encryption but no operation degrades VoIP performance. Maximum delta and maximum and mean jitter with or without SRTP for the bare PC softphone, which has no operating system, are smaller than for the snom and Twinkle softphones. This implies that VoIP performance may be improved with lean protocol implementations, simple tasking, and other bare PC softphone optimizations. Mean delta values for all three softphones are close to the ideal value. Overall, the results indicate that SRTP adds negligible overhead to VoIP processing and has no observable effect on VoIP quality.

REFERENCES

[1] M. Baugher, D. McGrew, M. Naslund, E. Carrara and K.Norrman, "The secure real-time transport protocol (SRTP)," RFC 3711, March 2004.

[2] P. Zimmermann, A. Johnston, and J. Callas, "ZRTP: Media Path Key Agreement for Secure RTP," Internet-Draft, March 2009.

[3] F. Andreason, M. Baugher, and D. Wing, "Session description protocol (SDP) security descriptions for media streams," RFC 4568, July 2006.

[4] D. Ignjatic, L. Dondeti, F. Audet, P. Lin, "MIKEY-RSA-R: An Additional Mode of Key Distribution in Multimedia Internet KEYing (MIKEY)," RFC 4738, November 2006.

[5] T. Dierks and C. Allen, "The TLS protocol version 1.0," RFC 2246, January 1999.

[6] D.Wing, S. Fries, H. Tschofenig, and F. Audet, "Requirements and analysis of media security management protocols," RFC 5479, April 2009.

[7] S. Garg, N. Singh, and T. Tsai, "SRTP+, An efficient scheme for RTP packet authentication," Retrieved Nov. 2, 2008 from pubs.research.avayalabs.com/pdfs/ALR-2004-001-paper.pdf.

[8] S. Garg, N. Singh, and T. Tsai., "Schemes for enhancing the denial-of-service tolerance of SRTP," pp. 409-411, 1st Int. Conf. on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM 05), 2005.

[9] S. Spinsante, E. Gambi, E. Bottegoni, "Security solutions in VoIP applications," IEEE International Symposium on Consumer Electronics (ISCE 2008), pp. 1-4, 2008.

[10] Kamailio (OpenSER) SIP server, http://sourceforge.net/projects/openser/

[11] Twinkle, http://www.xs4all.nl/~mfnboer/twinkle/index.html.

[12] G. H. Khaksari, A. L. Wijesinha, R. K. Karne, L. He, and S. Girumala, "A peer-to-peer bare PC VoIP application," 4th IEEE Consumer Comm. and Networking Conf. (CCNC 2007), pp. 803-807, 2007.

[13] Kamailio (OpenSER) SIP server, http://sourceforge.net/projects/openser/

[14] Wireshark, http://www.wireshark.org.