情報保護學會論文誌
第17卷 第2號, 2007. 4

QoS Evaluation of Streaming Media in the Secure Wireless Access Network

Jong-woo Kim,† Seung-wook Shin, Sang-duck Lee, Seung-jo Han‡
Chosun University

보안 무선엑세스 네트워크에서 스트리밍 미디어의 QoS 평가

김 종 우,† 신 승 욱, 이 상 덕, 한 승 조‡
조선대학교

ABSTRACT

With the increasing growth of Internet and wireless IP networks, Multimedia systems need to be envisaged as information resources where users can access anywhere and anytime. However, efficient services in these multimedia systems are open and challenging research problem due to user mobility, limited resources in wireless devices and expensive radio bandwidth. To implement multimedia services over heterogeneous network, the IP header compression scheme can be used for saving bandwidth. In this paper, we present an efficient solution for header compression, which is modified form of ECRTP. It shows an architectural framework adopting modified ECRTP when IP tunneling network using GRE over IPSec is implemented. We have conducted simulations in order to analyze the effects of different header compression techniques while delivering real-time services to the wireless access network through secured IP Network. The impacts on performance have been investigated through a series of experiments.

Keywords : *Wireless Access Network, IP tunneling, ECRTP*

Ⅰ. Instroduction

With the increasing growth of Internet andwireless IP network, multimedia systems need to be envisaged as information resources where users can access for anywhere and anytime. However, efficient services in these multimedia systems are open and challenging research problem due to user mobility, limited resources in wireless devices and expensive radio bandwidth.

As networks evolve to provide more bandwidth, the applications, services and consumers of those applications all compete for that bandwidth. For network service operators, it is important to offer high Quality of Services (QoS) in order to encourage clients to use their network as much as possible. As for wireless networks with their high bit error rates (BER) and high latency, it is difficult to attain those high bandwidths required. When all these factors are taken into account it means that available resources must be used as efficiently as possible. For these multimedia services, the payload of IP packets is al-

접수일: 2006년 6월 26일 채택일: 2007년 1월 24일
* 이 논문은 2006년도 조선대학교 학술연구비의 지원을 받아 연구되었음
† 주저자, mmm@7.co.kr
‡ 교신저자, sjbhan@chosun.ac.kr

most the same size or even smaller than the header. It is possible to compress those headers and thus save bandwidth and use expensive resources efficiently.

In this paper, we present an efficient solution for header compression. Modified enhanced header compression scheme is used to carry IP header compressed packets over an IP tunnel. The main intention of our investigation is to show the effect of different header compressions in wired/wireless network with implementation of IP tunneling.

Ⅱ. Real-time Transport Protocol

Multimedia applications often use RTP, UDP, and IP as protocols. Real-time transport protocol (RTP) [1] is an IP-based protocol providing support for the transport of real-time data such as video and audio streams. Thus RTP provides end-to-end delivery services for data with real-time characteristics. The services provided by RTP include time reconstruction, loss detection, security and content identification. RTP is primarily designed for multicast of real-time data, but it can be also used in unicast. It can also be used for interactive services such as Internet telephony.

RTP itself does not provide any mechanism to ensure timely delivery however. It needs support from lower layers that actually have control over resources in switches and routers. RTP does not address resource reservation and does not guarantee quality-of-service for real-time services. RTP itself however, does not provide all of the functionality required for the transport of data and, therefore, applications usually run it on top of a transport protocol such as UDP.

RTP is designed to work in conjunction with the auxiliary control protocol, Real Time Control Protocol (RTCP), to get feedback on quality of data transmission and information about participants in the on-going and to provide minimal control over the delivery of the data. RTCP provides support for real-time conferencing of groups of any size. [2]

Ⅲ. IP Tunneling

Due to the interest in emerging scenarios such as wireless access networks over public IP environments, some tunneling technologies have been introduced. Currently used tunneling protocols are Layer 2 Tunneling Protocol (L2TP) Tunnel, Layer 2 Forwarding (L2F) Tunnel, IP Security (IPSec) Tunnel, and Generic Route Encapsulation (GRE) Tunnel. [3].

IPSec [4] is a suite of protocols "designed to provide security services, such as access control, data integrity, authentication, confidentiality (encryption), and replay protection to the IP layer as well as the layers above. [5] Security Association (SA) is uniquely identified by a Security Parameter Index (SPI), an IP destination address, and a security protocol. Authentication Header (AH) [6] and Encapsulating Security Payload (ESP) [7] are secure protocols provided by IPSec to form SAs. These protocols can be used alone or in combination. Each supports two modes of use: transport mode and tunnel mode. Each SA defines the algorithms for encryption, authentication, hash and key exchange (attributes) for protecting a particular path.

Generic Route Encapsulation (GRE) [8] is a tunneling protocol that encapsulates traffic with new packet headers to ensure delivery to specific destinations. The network is considered private because traffic normally enters a tunnel only at the beginning and endpoint of the tunnel. Although limiting traffic access in this manner may deem the network private, it does not provide message confidentiality or integrity. There are several potential benefits of using GRE and IPSec on the same router. GRE can be used to encapsulate non-IP traffic in IP packets. The GRE tunnel packet is an IP unicast packet, so the GRE packet can be encrypted using IPSec.

Ⅳ. Voice over IP

Voice over IP (VoIP) [9] is one of the fastest

growing Internet applications today. While a VoIP device can accept a digital call directly, a coder-decoder (CODEC) must convert analog calls before they can be transported over a packet-switched network. To transmit voice data over a data network, the voice data must first be digitized and then compressed into small units known as packets. The compression algorithm determines the size and the transmission interval of these packets. The most popular coding standards for telephony and voice packet are: G.711, G.723.1, and G.729.

4.1 Impact of IPSec on voice traffic

Two main factors affect voice traffic when IPsec is used. The first one is the increased packet size because of the headers added to the original IP packet, namely the ESP header for confidentiality and the new IP header for the tunnel. The second one is the time required to encrypt payload and headers and the construction of the new ones. Figure 1 depicts the voice packets without and with IPSec.

An IP packet with voice data will have IP header, UDP header, and RTP header for a total of 20+8+12=40 octets. The size of the voice data depends on the codec used; it can be 15-30 bytes.

It can be seen that for VoIP traffic, where IPsec could more than double the size of a G.729 voice packet, as shown in Figure 1. The Layer 3 data rate for a G.729 call (at 50 pps) is 24 kbps (60 bytes x 8 bits x 50 pps). IP GRE tunnel overhead adds 24 bytes per packet. IPsec ESP adds another 52 bytes.

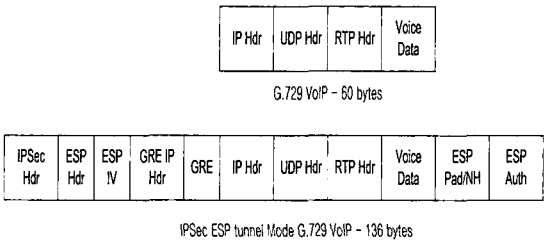


Figure 1. Packet Format for only G.729 VoIP and G.720 VoIP with IPSec ESP tunnel mode

The combined additional overhead increases the rate from 24 kbps (clear voice) to just less than 56 kbps (encrypted voice).

The packet size increase has a negative effect not only on bandwidth usage but also impacts on the transmission delay, router internal delays, queueing delay, thus affecting jitter and overall packet delay. The transmission delay increases proportionally with the packet size and is constant for every router. Internal router delays are considered in the generic IPsec delay. Queueing delay is sensitive to packet size as well, which is evident with low bandwidth links.

V. Header Compression Schemes

In many services and applications, such as Voice over IP (VoIP) or messaging, the payload of the IP packet is almost the same size or even smaller than the header. Over the end-to-end connection, comprised of multiple hops, these protocol headers are extremely important but over just one link (hop-to-hop) these headers serve no useful purpose. It is possible to compress those headers, thus save the bandwidth and use expensive resources efficiently. IP header compression also provides other important benefits, such as reduction in packet loss and improved interactive response time. IP header compression schemes have always been an important part of saving bandwidth over bandwidth limited links. IP header compression [10] also provides other important benefits, such as reduction in packet loss and improved interactive response time.

An important principle in the design of header compression standards has been robustness. There are two aspects to this: robustness to packet loss, and robustness to misidentified streams. Network links, especially wireless links can lose or corrupt packets and the header compression scheme must be able to function in the presence of such damage. The most important requirement is that damage to the compressed bit stream not cause undetectable corruption in the uncompressed stream. If a packet is damaged, the fol-

lowing packets will either be decompressed correctly or discarded. The decompressor should never produce a corrupted packet.

5.1 Compressed Real-time Transport Protocol

RTP header compression (CRTP) [11] was designed to reduce the header overhead of IP/UDP/RTP datagram by compressing the three headers. The IP/UDP/RTP headers are compressed to 2-4 bytes most of the time. CRTP was designed for reliable point to point links with short delays. For lossy links and long round trip delays, CRTP does not perform well. After a single lost packet several sequential packets are lost within the round trip time. Thus, CRTP is not suitable for wireless links, which have typically a very high, variable bit error rate (BER).

CRTP is designed to compress IP/UDP/RTP flows. CRTP uses four packets formats: full header, Compressed UDP, Compressed RTP and context state.

After a full header packet has been sent to establish the context, the transition to Compressed RTP packets may occur. Each Compressed RTP packet indicates that the decompressor may predict the headers of the next packet on the basis of the stored context. Compressed RTP packets may update that context, allowing for common changes in the headers to be communicated without full header packet being sent. Compressed RTP is depicted in Figure 2.

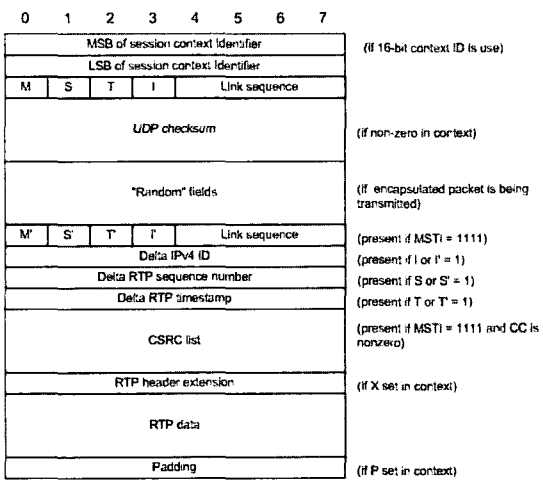


Figure 2. Format for Compressed RTP Packet

5.2 Enhanced CRTP

The Enhanced CRTP (ECRTP) [12] for links with high delay, packet loss and reordering feature includes modifications and enhancements to CRTP to achieve robust operation over unreliable point-to-point links. Thus ECRTP was developed to overcome the problems of CRTP as CRTP does not perform well over links with long round trip time that lose and reorder packets.

ECRTP extends CRTP by repeating context updates and by sending absolute values along with delta values when encoding monotonically thereby increasing header fields for increased robustness. It inserts a header checksum when UDP checksum is missing, to improve error recovery and fail checks for the compression.

The packet format Full Header had some changes. The first two length-fields in the IP/UDP/RTP header and possible encapsulating headers are changed in the ECRTP compressor. The fields are used to send information about the flow to the decompressor.

In CRTP, the compressor sends a context identifier, a sequence number and a generation number. In ECRTP, the same fields are sent but the formats of the fields are changed. The C bit is included in the length fields indicating the new header checksum, replacing the missing UDP checksum over the compressed link. A check for a zero UDP checksum when parsing through the whole header must be done.

This mechanism improves the header compression performance, especially for highly error - prone links and long round trip times [13]. ECRTP is used as a header compression scheme for real-time traffic. And ECRTP is thus suitable for many applications in the scenarios such as IP tunneling and other virtual circuits. ECRTP uses local retransmissions to more efficiently recover from wireless link errors. [14]

5.3 Modification in ECRTP

In order to reduce header overhead, we have modified the packet format of ECRTP header compression scheme.

In all Compressed RTP, ECRTP packets have 2 bytes of either the UDP checksum or the compressor inserted Header Checksum. The average header size can be reduced if we send these checksums in some packets only. Robustness will be achieved if there is some way of conveying a correct decompression of these packets to the compressor.

For instance, if only three checksum for every 16 packets were sent, then, assuming at least one of these packets is acknowledged, it can be expected the average header size to drop by about 1.6 bytes, given the fact that Compressed RTP packets are sent most often. Furthermore, this clearly implies a reduction in implementation complexity.

The Compressed RTP header includes the whole IP/UDP/RTP. Compressed RTP with individual RTP fields is shown in Figure 3. With respect to the original packet structure of Compressed RTP, the T bit is replaced by the C bit, which represents whether or not a Header Checksum is included in this packet. The S and I bits are set to 0. Hence, it is distinguishable from the Compressed UDP F=1 packet, where the corresponding bits are I and T, at least one of which, is 1. It is distinguishable from Compressed UDP F=0, because the corresponding bits are I and dI, both of which are 1, since the packet is a refresh packet.

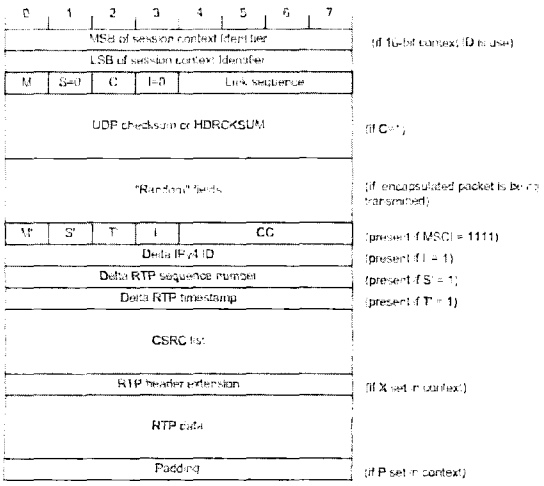


Figure 3. Format for Modified ECRTP Packet

VI. Related Works

The article [15] analyzes the impact of CRTP performance over cellular environment using real-time traffic such IP telephony and shows that CRTP does not cope with packet loss very well. The article [16] has analyzed the transmission of voice over secure communication links on implementing IPsec and presented an efficient solution for packet header compression, called cIPSec, for real-time traffic. The RFC 4170 [17] describes a method to improve the bandwidth utilization of RTP streams over network by providing compression, multiplexing, and tunneling over a network when multiple RTP streams are carried over that path. However, more studies are required to investigate impact of IP header compression in real-time services over wired/wireless networks while implementing IP tunneling.

VII. Experimental Validation

Figure 4 shows the conceived system architecture for a secure wired/wireless network, which comprises of a service provider, IP backbone network and wireless access networks. And in the system architecture, GRE over IPSec ESP is considered for IP tunneling.

In order to validate the conceived architectural model, we have simulated a scenario that includes the real-time application services such as audio streaming to the wireless access network over public IP network using an OPNET Modeler [18], which is a discrete event-driven simulator tool capable of modeling both wireless and wired networks.

7.1 Scenarios

Practical applications of multimedia services for

mobile users are likely to occur in scenarios where a wireless access network is extended over public IP backbone network (i.e. the Internet).

The basic architecture for the audio streaming service in secure wired/wireless IP network is shown in Figure 5. In Figures 6 and 7, the service provider and wireless access network are depicted. The service provider consists of audio streaming servers. And the wireless access network consists of 10 wireless clients using wireless IEEE 802.11b devices.

In the simulation, we conducted a series of experiments in turn, the results obtained, in terms of packet loss rate, average header size and probability loss in

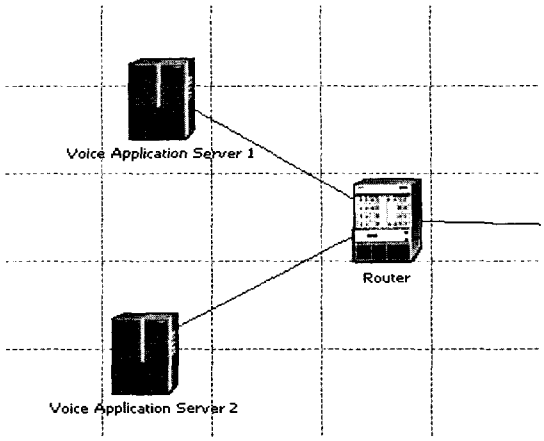


Figure 6. Service Provider for providing voice service by using OPNET Modeler

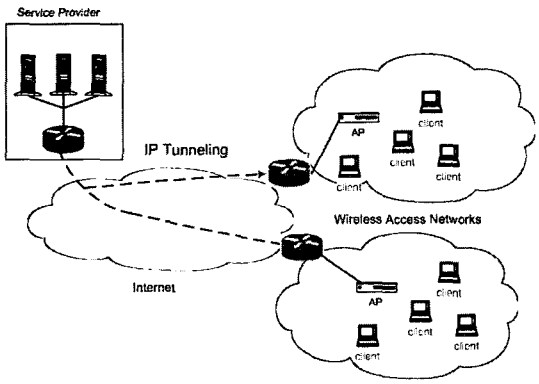


Figure 4. System architecture for streaming media in wireless access network through public IP network

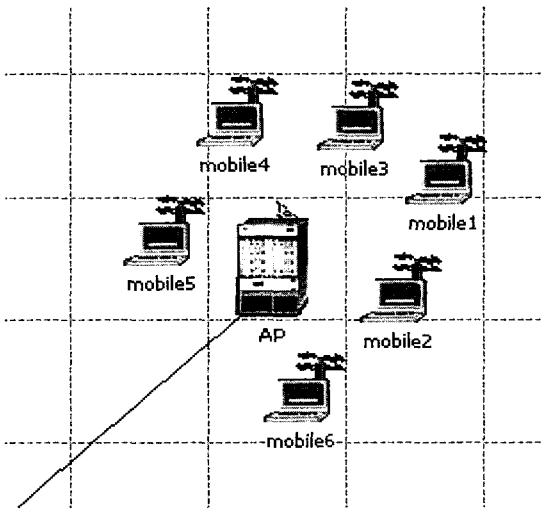


Figure 7. Wireless Access Network with mobile clients by using OPNET Modeler

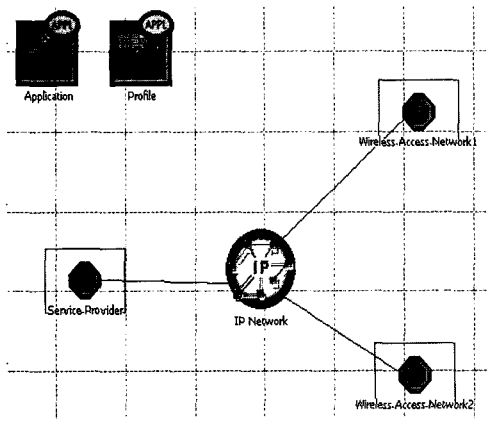


Figure 5. Simulation Model for streaming media in wireless access network through public IP network using OPNET Modeler

context as a function of bit error rate (BER) have been investigated. A comparison with results obtained with ECRTP allows for evaluation of the performance of modified ECRTP in a realistic environment.

7.2 Implementation of IPSec model in OPNET

The GRE/IPSec Model is based on the IETF RFC specifications and associated data encryption specifications. Various components of the IPSec protocol including Encryption and Authentication com-

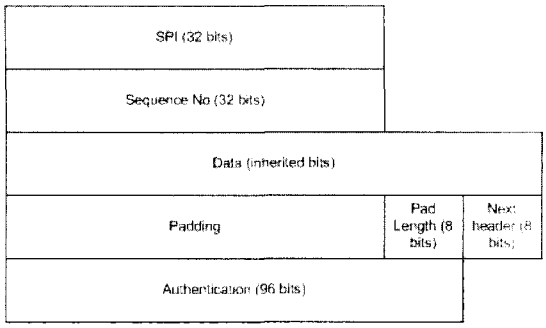


Figure 8. A packet format for ESP

ponents were coded to implement this model in OPNET Modeler. The encryption and authentication throughputs were obtained from Crypto++ 5.2.1 Benchmarks [19], which were used within the model to compute authentication and encryption delays. The GRE/IPSec model was then incorporated into the IP layer of the TCP/IP protocol stack.

ESP in transport mode was used for this model. The transport mode is used when security on a link must be transparent to a gateway (if one exists) along the link from source to destination. ESP packet model is shown in Figure 8.

The throughputs of the encryption and authentication algorithms were obtained from Crypto++ 5.2.1 Benchmarks [19] with computer Configuration: Pentium 4- 2.1GHz processor speed with Window XP SP1.

The Triple-DES algorithm is a symmetric block cipher algorithm. Symmetric means that the key used to encrypt a message must be the one used to decrypt it. Block cipher means that the message is broken down into blocks of a specified number of bits with each block encrypted and decrypted separately. [20] The throughput of 3DES is 1.231 Mbps.

A Message authentication code (MAC) is used to provide connectionless integrity checks for information. Hash MAC (HMAC) is an integrity check mechanism that depends on an underlying hash algorithm and an authentication key. RFC 2104 [21] defines a generic HMAC function denoted by HMAC-h-t, where 'h' denotes specific hash algorithm e.g. MD5, SHA-1, and 't' represents the length of integrity check value (ICV).

For this model, the specific HMAC used is HMAC-SHA1-96. SHA-1 as defined in FIPS PUBS 180-2 [22] takes input data of any length and outputs a 160 bit ICV. It is possible to truncate the ICV to some specific length without compromising the authentication property of IPSec. The ICV used within this model was truncated to 96 bits as recommended in RFC 2404. [23] SHA-1 throughput is 8.497 Mb/s.

Brief IPSec outbound and inbound processing Functions are as follows:

```
Packet * encap (int x, Packet* inPk )
{
..
..
encrypt_size/encrypt_throughput - calculate encryption delay
auth_size/auth_throughput - calculate authentication delay
Out_Delay - total outbound delay
..
}
```

The module sends packets to the IP layer delayed by Out_Delay.

```
Packet *  decap (int x, Packet* inPk)
{
..
..
decrypt_size/decrypt_throughput - calculate decryption delay
auth_size/auth_throughput - calculate authentication delay
In_Delay - total inbound delay
..
..
}
```

The module sends packets to the transport layer delayed by In_Delay.

7.3 Setup Assumptions

In our experimental analysis, we selected G.729 as the voice codec and worked on the assumption that the voice characteristics in most fields remained constant. The IP-ID and RTP Sequence Number fields were assumed to increase by one from packet to packet. Silence suppression was assumed - so, the RTP Timestamp will jump by a constant amount, except at the end of a silence interval. Speech is bursty, with talk-spurts followed by silence periods. No packets are sent during these silence intervals. We have considered the IP public network ie. Internet with 5% packet discard ratio and average packet latency of 0.5 sec.

When specifying the QoS, following performance metrics are taken into account: End-to-End Delay (Latency) - The average time it takes for a packet to travel the network from a sending to receiving device; and Packet Loss - The percent of transmitted packets that never reach the intended destination.

It is of paramount importance to audio steaming application that QoS can be guaranteed from end-to-end. The three main network problems that affect voice quality are packet loss, and delay.

One of the important network design aspects of implementing voice service involves the calculation of bandwidth requirements. We wanted to see the bandwidth requirements for voice traffic in different scenarios. In our case, we selected voice CODEC G.729 having sampling rate of 20ms, voice payload of 20 bytes and 50 packets per sec (pps). In case of no IPSec and compression, the bandwidth requirement is calculated as (20 voice traffic + 40 IP/RTP/UDP + 6 L2 encapsulation) x 50 x 8 = 26.4kbps. The required bandwidth can be reduced for the given scheme by using RTP Header Compression (CRTP). As mentioned earlier, CRTP operates hop-by-hop and compresses the 40 byte IP/UDP/RTP headers to 2-4 bytes. So the link bandwidth requirement when using CRTP is 12.4kbps.

On implementing IPSec/GRE, encryption increases the bandwidth requirements of voice traffic, impacting the provisioning of service policies on output interfaces. Encrypting that packet using IPSec Tunnel mode for IP GRE increases the required bandwidth to 56.8kbps since it adds 76 byte of IPSec/GRE headers. While using CRTP or ECRTP, the bandwidth requirement can be reduced to 42.5 kbps. Since modified ECRTP reduces average header size, the bandwidth requirement can be slightly reduced to 42kbps.

Another network design aspect of implementing voice service involves the calculation of the end-to-end, one-way delay (latency). The ITU standard G.114 states that a latency of 150 msec is acceptable for high voice quality. For most networks, a delay of 200 msec will provide acceptable voice quality.

When considering the one-way delay of voice traffic, one must take into account the delay added by the different segments and processes in the network. Some components in the delay budget need to be broken into fixed and variable delay. Some important components of this latency are:

CODEC delay- Each compression algorithm has certain built-in delay. Choosing different CODECs may reduce the latency, but reduce quality or result in more bandwidth being used. This includes algorithmic delay, processing delay and packetization delay. Queueing delay - As voice packets enter an output queue and wait for the preceding frame (voice or data) to be played out. Serialization delay - Fixed delay required to clock a voice frame onto the network interface. Backbone (network) delay - This is the delay incurred when traversing the backbone network. In general, to minimize this delay, try to minimize the router hops that are traversed between end-points.

Jitter buffer delay - Because speech is a constant bit-rate service, the jitter from all the variable delays must be removed before the signal leaves the network.

In our experiment, we have considered G.729 as a CODEC, and link speed of 256kbps. So the end-to-end latency without IPSec and header compression can be calculated as (25 CODEC delay + 4

queuing delay + 1.5 serialization delay + 50 backbone network delay + 10 wireless transmission delay + 40 de-jitter buffer delay) ≈ 130.5ms.

The latency is reduced significantly by using RTP Header Compression (CRTP). The end-to-end delay while using CRTP is 102.5ms.

On implementing IPSec/GRE, encryption increases the end-to-end delay as it adds encryption/decryption delay of 20ms. The latency is increased to 167ms. While using CRTP, the latency can be reduced to 137ms, whereas with ECRTP, latency is about 135ms. With modified ECRTP the latency can be further reduced to 132.5ms.

7.4 Simulation Results

In order to analyze the simulation results in term of packet loss rate for various header compression schemes, we have devised the simulation scenario in which IPv4 audio streams are characterized.

Figure 9. shows the packet loss rate as a function of BER with and without header compressions while implementing

IP tunneling. Using header compression schemes,

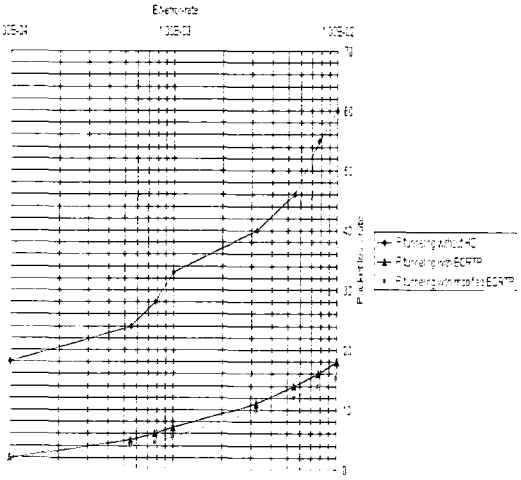


Figure 9. Packet loss rate versus bit error rate (BER) for IP tunneling without any header compression and with ECRTP and modified ECRTP

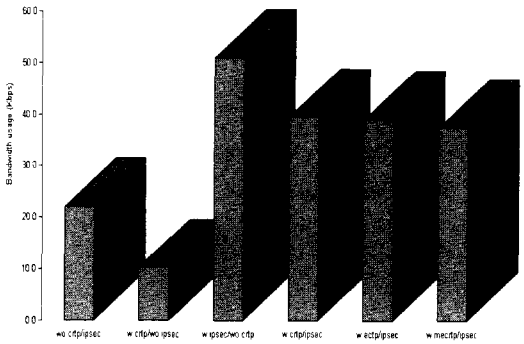


Figure 10. Bandwidth usage for various scenarios - without IPSec and header compression, without IPSec and with CRTP, without header compression and with IPSec; with CRTP and IPSec; with ECRTP and IPSec; and with modified ECRTP and IPSec

the packet loss rates for ECRTP and modified ECRTP have been significantly reduced in compare to uncompressed (raw) voice data. It can be seen that at a BER of 10-3, the packet loss rate for ECRTP is slightly higher than that for modified ECRTP.

Fig. 10 shows the bandwidth used in 6 scenarios - without IPSec and header compression; without IPSec and with CRTP; without header compression and with IPSec; with CRTP and IPSec; with ECRTP and IPSec; and with modified ECRTP and IPSec. It can be seen that in case of implementing IPSec/GRE in the network, the bandwidth usage for modified ECRTP is the lowest.

Figure 11. shows the average end-to-end delay in 6 scenarios - without IPSec and header compression; without IPSec and with CRTP; without header compression and with IPSec; with CRTP and IPSec; with ECRTP and IPSec; and with modified ECRTP and IPSec. It can be seen that in case of implementing IPSec/GRE in the network, the bandwidth usage for modified ECRTP is the lowest.

Ⅷ. Conclusion and Future Works

This paper provides a framework for multimedia

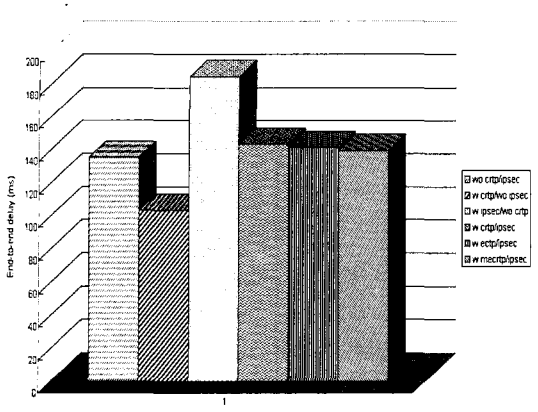


Figure 11. Average End-to-end Delay for various scenarios - without IPSec and header compression, without IPSec and with CRTP, without header compression and with IPSec; with CRTP and IPSec; with ECRTP and IPSec; and with modified ECRTP and IPSec

services such as m-learning through the public IP backbone network to the wireless access network using IP tunneling. The simulation scenario shows that ECRTP and modified ECRTP can work well over wired/wireless IP network with IP tunneling. With a modified ECRTP, the bandwidth requirement can be significantly reduced while implementing IPSec/GRE on the network. Simulation results show that the modified ECRTP compression scheme reduces the end-to-end one-way latency in compare to both CRTP scheme and ECRTP scheme.

In future work, we will concentrate on a detailed investigation on multimedia streaming, both voice and video over secure VoIP networks based on Session Initiation Protocol (SIP).

Furthermore, comparative study with other robust header compression scheme such as Robust Header Compression (ROHC) shall be preformed. [24].

References

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

[2] C. Perkins, RTP: Audio and Video for the Internet, Addison Wesley, 2003

[3] G. Schafer, Security in Fixed and Wireless Networks, John Wiley & Sons, Inc, 2003.

[4] R. Atkinson and S. Kent, "Security Architecture for Internet Protocol," IETF RFC 2401, Nov 1998.

[5] N. Doraswamy, and D. Harkins, "IPSec: the new security standard for the Internet, Intranets, and Virtual Private Networks," Prentice-Hall, Mar 2003

[6] R. Atkinson and S. Kent, "IP Authentication Header," IETF RFC 2402, Nov. 1998.

[7] R. Atkinson and S. Kent, "IP Encapsulating Security Payload (ESP)," IETF RFC 2406, Nov 1998.

[8] D. Farinacci,T. Li,S. Hanks, D. Meyer, and P. Traina, "Generic Routing Encapsulation (GRE)" IETF RFC 2784, March 2000.

[9] U. Black, Voice Over IP, Prentice Hall PTR, 1999.

[10] M. Degermark, B. Nordgren, and S. Pink, "IP Header Compression," IETF RFC 2507, Feb. 1999.

[11] S. Casner and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links," IETF RFC 2508, Feb.1999.

[12] T. Koren,S. Casner,J. Geevarghese, B. Thompson, and P. Ruddy, "Enhanced Compressed RTP (CRTP) for Links with High Delay, Packet Loss and Reordering" IETF RFC 3545, July 2003.

[13] K. Svanbro, H. Hannu, L.-E. Jonsson, and M. Degermark, "Wireless Real - time IP Services Enabled by Header Compression," Proceedings of the IEEE Vehicular Technology Conference (VTC), vol. 2, Japan, 2000, pp. 1150 - 1154.

[14] W.T. Chen, D.W. Chuang, and H.C.Hsiao, "Enhancing CRTP by retransmission for wireless networks"
Proceedings of the Tenth International Con-

ference on Computer Communications and Networks, 2001, pp. 426 - 431.

[15] M. Degermark, H. Hannu, L. Jonsson, and K. Svanbro, "Evaluation of CRTP Performance over Cellular Radio Links," IEEE Personal Communications, Vol. 7, No. 4, pp. 20-25, August 2000.

[16] R. Barbieri, D. Bruschi, and E. Rosti, "Voice over IPsec: Analysis and Solutions", Proceedings of 18th Annual Computer Security Applications Conference, USA, Dec. 2002, pp 261-270.

[17] B. Thompson, T. Koren, D. Wing, "Tunneling Multiplexed Compressed RTP (TCRTP)," IETF RFC 4170, Nov 2005

[18] OPNET Modeler Simulation Software, http://www.opnet.com

[19] Crypto++ 5.2.1 Benchmarks, http://www.eskimo.com/~weidai/benchmarks.html

[20] A.J. Menezes, P.C. van Oorschot, and S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997.

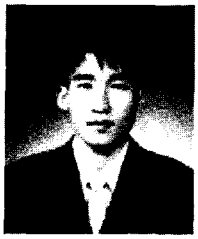
[21] H. Krawczyk, M. Bellare, R. Canetti, HMAC: Keyed-Hashing for Message Authentication, RFC 2104, Feb 1997

[22] National Institute of Standards and Technology (NIST), Secure Hash Signature Standard (SHS), FIPS PUB 180-2, Aug 2002

[23] C. Madson and R. Glenn, "The Use of HMAC-SHA-1-96 within ESP and AH," IETF RFC 2404, Nov. 1998.

[24] C. Bormann, C. Burmeister, M. Degermark, et al, "Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed," IETF RFC 3095, Jul 2001.

〈著者紹介〉



김 종 우 (Jong-woo Kim) : 주저자
1998년 2월: 조선대학교 전자공학과 학사
2000년 8월: 조선대학교 대학원 전자공학과 석사
2005년 2월: 조선대학교 대학원 전자공학과 박사
2005년 4월~2006년 2월: 조선이공대학 인터넷 정보과 전임강사
2005년 2월~현재: (주)일우통신 이사
<관심분야> 통신보안시스템설계, 네트워크 보안, DRM, 무선 네트워크 보안
mmm@7.co.kr

이 상 덕 (Sang-duck Lee)
1997년 2월: 조선대학교 전자공학과 학사
1999년 2월: 조선대학교 대학원 전자공학과 석사
2000년 2월~현재: 조선대학교 대학원 전자공학과 박사과정
2005년~현재: 조선대학교 정보통신공학과 외래교수
<관심분야> 네트워크 및 시스템보안, 임베디드 시스템, DRM
dandylsd@daum.net

이 상 덕 (Sang-duck Lee)
1997년 2월: 조선대학교 전자공학과 학사
1999년 2월: 조선대학교 대학원 전자공학과 석사
2000년 2월~현재: 조선대학교 대학원 전자공학과 박사과정
2005년~현재: 조선대학교 정보통신공학과 외래교수
<관심분야> 네트워크 및 시스템보안, 임베디드 시스템, DRM
dandylsd@daum.net

한승조(Seung-jo Han) : 교신저자
1980년: 조선대학교 전자공학과
1982년: 조선대학교전자공학과 석사
1994년: 충북대학교 전자계산학과 박사
1986년 6월~1987년 3월: 뉴올리언스대학 객원교수
1995년 2월~1996년 1월: 텍사스대학 객원교수
2000년 12월~2002년 2월: 버클리대학 객원교수
2005년 11월~ 현재 조선대학교 정보전산원장
1998년 3월~현재: 조선대학교 전자정보통신공학부 정교수
<관심분야> 정보보안, 컴퓨터네트워크, DRM, S/W 불법복지방지시스템
sjbhan@chosun.ac.kr