中央研究院
資訊科學研究所
Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-05-012

An Application-layer Security Control for Real-time Video Streaming

Chia-Hui Wang, Jan-Ming Ho

September 2005 || Technical Report No. TR-IIS-05-012

http://www.iis.sinica.edu.tw/LIB/TechReport/tr2005/tr05.html

The work was done/partially done while the author was visiting the Institute of Information Science, Academia Sinica, Taiwan in 2004.

An Application-layer Security Control for Real-time Video Streaming

Chia-Hui Wang, Jan-Ming Ho
Department of Computer Science and Information Engineering, Ming Chuan University
wangch@mcu.edu.tw
Institute of Information Science, Academia Sinica.
hoho@iis.sinica.edu.tw

Abstract- *In the shared Internet, real-time video streaming service is now prevalent and popular. Real-time video streaming services such as video conferencing, surveillance videos and live videos may preserve privacy and commercial values. Thus, it's very important to secure real-time video streaming services from potential eavesdropper.*

However, security has been aware of inadequacy for real-time video streaming applications because it contends for lots of resources.

In this paper, we propose an effective application-level security control to protect the real-time video streaming. This method will economically transpose the data block in the sender's buffer by a given secrete key of database through a receiver's buffer occupancy feedback control. Authenticated receiver can restore the scrambled video data from the receiver's buffer by the secrete key to playback the original video. However, without the secrete keys to restore the data, eavesdropper will not be able to playback the video. It's hard to break the encryption even if eavesdropper tries to store the scrambled video data for processing later.

We evaluate the proposed scheme's effect to the playback QoS of real-time video streaming and secure ability through theoretical analysis and some experimental data. Furthermore, this method will be applied on a test-bed of Internet video surveillance services to demonstrate the resource-saving and highly secure capabilities.

Keywords: Video Streaming Security, Secrete Key, Transposition Cipher, Feedback Control.

This research was supported in part by National Science Council, Project No. NSC92-2815-C-424-001-E and Project No. NSC92-2218-E-424-001.

1. Introduction

Based on the great absorption and acceptability of multimedia, diversified multimedia applications are playing a very important role on the prevalent Internet. Multimedia network applications such as e-learning, digital library, video on demand, video conference and video surveillance change human daily life. Besides, the infrastructure of broadband network is almost furnished, so the effective delivery of diversified media content is the key to the success in Internet business.

Due to the open architecture of Internet, lots of companies, research organizations and even individuals can easily work together to make their own business progress through the public Internet community. But, the open architecture also makes the data communications vulnerable to security and privacy. Computer hackers take advantage of the ubiquitous connectivity of Internet through wired or wireless devices [13] to easily start attacks. They have successfully stolen others' private data through the Internet.

Current Internet security solutions decouple the security component from the original application design by using existing security protocol or dedicated hardware devices. At the network application layer, Secure Socket Layer (SSL) is common used to protect the common data communication in the Internet.

Because the Internet multimedia communications may also preserved privacy and confidential in the Internet, it's important to secure real-time video streaming services from potential eavesdropper. But, most developers for real-time video streaming applications choose to ignore security provision completely because security services such SSL may content a lot of resources to reduce the performance of real-time video streaming applications. Therefore, we propose an application-layer security mechanism for real-time video streaming service. This security mechanism economically transposes the data blocks in sender's packet to effectively encrypt the video stream without degrading the performance real-time video streaming services.

We will first introduce related researches in the following section. In section 3, the proposed security scheme will be presented in detail. The analysis and experimental results are investigated in section 4. The conclusions and future work are presented in the final section.

2. Real-time Video Streaming and Security

In this section, we will introduce the some researches in the techniques of real-time video streaming and video security in section 2.1 and section 2.2 respectively.

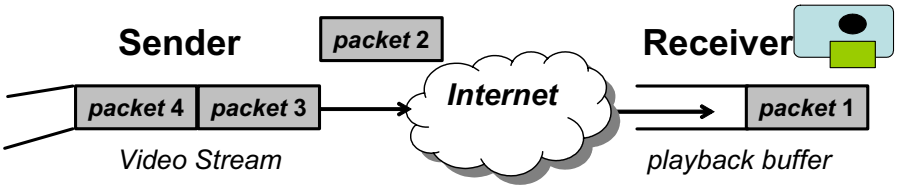

Figure 1. Application-level real-timevideo streaming.

2.1. Real-time Video Streaming

As shown in Figure 1, while video stream is sent out to network, the video stream should be divided into packets and each packet is tagged with its sequence number. Multimedia applications will deliver these packets sequentially. Receivers will also playback the arrival packets sequentially according to the packet's sequence number.

However, the bandwidth of Internet is shared by all kinds of applications to achieve statistical multiplexing gain. It introduces considerable uncertainty in workload and resource requirements. While delivering packets through a shared network, the unpredictable delay jitter may introduce underflow or overflow in a limited playback buffer even if the network is error-free at a time period.

Due to delay variation in the shared network, the receiver may playback no data while the playback buffer is empty. It will seriously degrade the playback quality. That is the reason why real-time video streaming requires strictly timing delivery and most developers provides no security at all for real-time video streaming applications.

The real-time streaming applications require end-to-end quality-of-service (QoS) with jitter-free playback of audio and video. Thus a good end-to-end flow control mechanism is needed to maintain high throughput and keeping average delay per packet at a reason level for such time-critical video streaming applications.

Rate-based flow control can provide end-to-end deterministic and statistical performance guarantees over modern packet-switching networks [14]. Its short propagation delay has brought wide deployment of the rate-based flow control in real-time video streaming. Usually, the rate adjustment is performed by the intensive feedback controls [12] from receiver to achieve playback QoS.

Our proposed security scheme is based on the previous work [8] of [12] rate-based flow control with buffer occupancy feedback.

2.2. Video Security

Multimedia network applications are growing vigorously in the Internet. Most of them need effective protection of security and privacy in the open Internet. Many encryption schemes such as SET, SSL et al. [10]

are proposed to provide protections for data communication in the Internet. However, these schemes need to contend resources to encrypt every single bit of the raw data.

These encryption methods are not applicable to real-time video streaming applications, which need strictly sending, receiving and playback timing constraints. Nowadays most proposed video streaming security mechanisms are strongly dependent on the coding scheme of media. So, we classify them as media-level protection scheme.

One kind of the media-level protection is called partially encryption [3][4][5][6]. Either intra-frames or frame headers are encrypted through a secret key. Each intra-frame of the video streaming preserves whole image picture information. Besides, the frame header also preserves the important attributes of the related video frame. So receivers without the secrete key to decrypt the intra-frame or frame header cannot playback the partially encrypted video data.

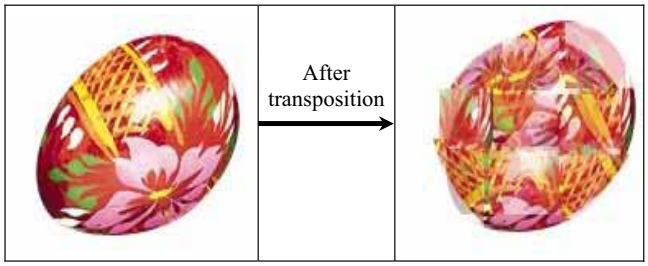

Figure 2. A sample image and its transposed image.

Another kind of media-level protection scheme makes transpositions on the blocks of a singe image picture. As shown in Figure 2, a sample image is on the left and its scrambled image is on the right. It's hard to perceive the content in the scrambled data. We believe the complexity to recover the scrambled image without the transposition key is much the same as to solve the jigsaw puzzle.

The above-mentioned video security schemes are strongly dependent on the compression or coding schemes for different media. While providing media-level video protection scheme, applications need to spend time to parse the bit sequence of the video stream to find out the location of the intra-frames or frame header.

Now, we propose a security scheme in the application level to transpose some blocks among the video packets according to the feedback of the buffer occupancy in receiver's playback buffer. Without degrading the real-time video streaming constraints, our proposed application-layer security scheme does not have to parsing the bit sequence for different streaming media in the Internet.

The proposed security scheme for real-time video streaming applies the similar encryption idea from jigsaw puzzle to transpose some block in video packets. The scrambled blocks in video packets will result in the error

propagation [11] in video rendering. So, eavesdropper can not playback the encrypted video without the secrete key to recover the original bit sequence in video streaming.

3. Application-layer Security Scheme for Real-time Video Streaming

As mentioned in Section 2.1, the video data will be divided into packets with sequence number before they are sent to the network. Internet applications apply the UDP protocol to deliver these video packets. Then, receiver will receive these packet and playback them sequentially according to their sequence numbers.

We propose an application-level security scheme for the real-time video streaming to simply transpose data blocks in the packet without prior knowledge about the media coding scheme. We believe that our security scheme is effective because of the error propagation in video compression. After the data blocks are transposed in video packets, the video bit sequence is scrambled and the video error may propagate to the next synchronization point in the video bit stream. Therefore, eavesdropper can not playback the video stream in the public Internet if every video packet is scrambled. Now, let's first take a look at the idea of transposition cipher applied in the following subsection.

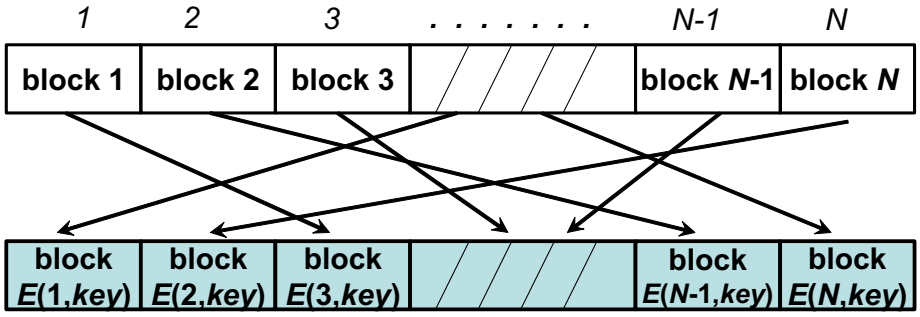

Figure 3. A sample to encrypt a video packet through transposition function $E(i, key)$.

3.1. The Transposition Cipher Scheme

In Figure 3, the upper figure shows the original block sequence in a packet and the lower figure indicates the new block sequence after transposition through the function $E(i, key)$. The transposition function $E(i, key)$ will give a number j (i.e. $j = E(i, key)$) of data block and the block j will be given a new position i.

Usually, the secrete key in transposition cipher is represented by a set of sequence number to represent the new block sequence. For example, we have 5 blocks in a packet (i.e. N is 5), the original block sequence is <1, 2, 3, 4, 5>. If secrete key *key* is <4, 2, 5, 3, 1> and the secrete key simply indicates the new sequence. The authenticated receiver can use this key to decrypt the encrypted packet and to recover the original block

sequence in video packet. So, the time complexity of transposition cipher is related to the number of permutations blocks. In the above-mentioned example, we have 5 moves of block position for encryption and decryption by respectively.

Therefore, if the number of block moves is larger (i.e. the number of moves is not larger than *N*), the transposition is more time-consuming. To lessen the security loading to the real-time video streaming service, we will change the secrete key to a new secrete key with less number of block moves in our proposed security scheme.

The system architecture of proposed security scheme for real-time video streaming is shown as Figure 4. After finishing transposition from encrypt function *E*, application in the sender will deliver the encrypted packet to the network. The authenticated user will receive the secrete key through the secure channel such as SSL, et al. to recover the block sequence in encrypted packet by the inversion function of *E*.

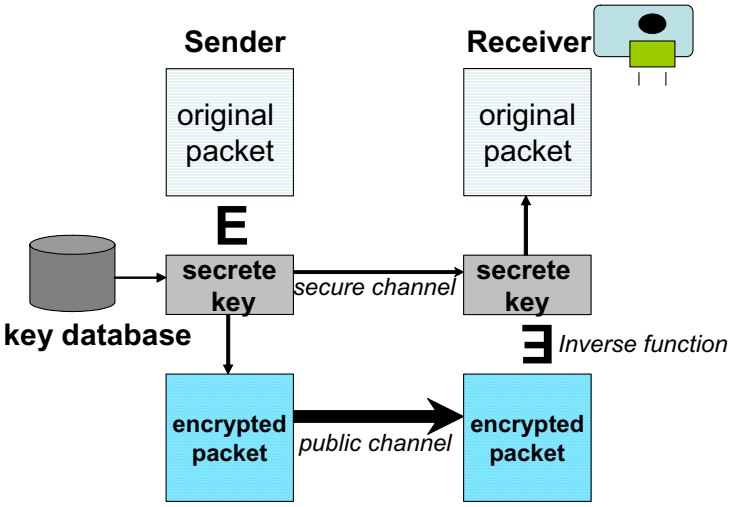

Figure 4. Security architecture for real-time video streaming.

Let's take a simple cryptanalysis for the transposition cipher. If a packet is divided into *N* data blocks and we have a secrete key to encrypt the packet and sent it to the public network. Now the hacker has the packet and tries to restore the encrypted packet by enumerating and validating all the possible block sequence in the packet. We assume that the time to validate a possible block sequence is *T*. Then, the average time of hacker's breaking is shown in the following formula.

$$T \times \frac{\sum_{i=1}^{N!} i}{N!} = T\frac{(1+N!)}{2} \qquad (1)$$

For example, if a packet is divided by 16 data blocks (i.e. *N*=16) and the validation time is 1 *ms*. The average time to find out the correct sequence is 10461394944.0005 seconds and it's about 332 years. So, if the

N is large, to recover the encrypted packet is very hard. It's known that the larger number *N* in block sequence we apply, the more time we need to move blocks in the packet. But, the small *N* blocks in a packet is vulnerable for hacker to break. Therefore, we apply the feedback control of receiver's buffer occupancy to dynamically change the secrete key.

Besides, the functionality of the key database provides the data of secrete key while the proposed security system needs to dynamically changing the secrete key. The proposed security system will pick up new key from the key database while trying to equilibrate the resource contention between the security and the real-time video streaming services. In the following subsection, we will present the transposition control with feedback of buffer occupancy to dynamically change key.

3.2. The Transposition Control with Feedback of Buffer Occupancy

Real-time video streaming requires strictly timing of delivering and rendering the video packets. While the receiver's buffer running overflow and underflow, the playback quality of real-time video streaming will be jeopardized. Therefore, the receiver's buffer occupancy of arrival packets indicates the maximum delay tolerance for the next arrival packet. Nevertheless, the security scheme such as transposition cipher needs to spend the time to move as many data blocks as possible to increase the strength of protection.

Because of the resource contention between real-time streaming services and security, exploiting a dynamic key changing can be a good approach to balance the security and real-time video streaming. However, feedback control ideas, those gain momentum as a promising foundation has been used successfully in computer systems with complex and unpredictable workloads. Therefore, we will apply the feedback control idea to balance the loading between security and real-time video streaming in the shared Internet.

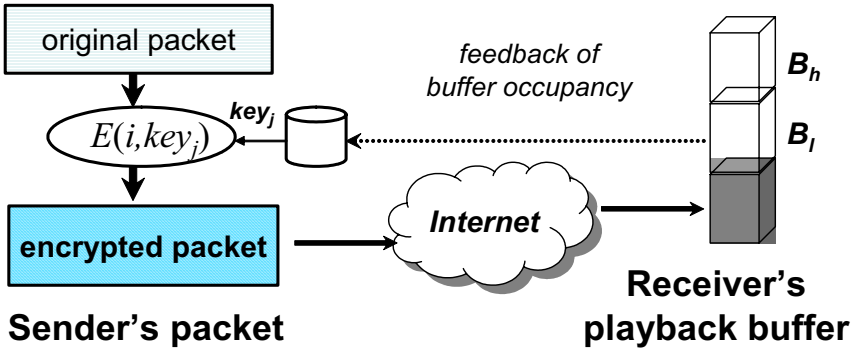

Figure 5. Proposed transposition control with feedback of buffer occupancy.

The proposed transposition control with feedback of buffer occupancy is illustrated in Figure 5. Firstly, while the occupancy in a limited buffer running above the high watermark B_h, sender will change the current secrete key to a new one according to the feedback buffer occupancy, which will decompose more blocks in packet to transpose to boost the security strength.

Secondly, while the receiver's buffer running below the low watermark B_l, we will change current secret key to a new key with less movement of data blocks to encrypt the packet. Finally, while the buffer occupancy running within B_l and B_h, applications can change current secrete key to a new one with same number of block movement to strengthen the security.

The basic idea for the proposed security scheme is to adaptively change secrete key while cannot transpose as more data blocks as possible to maintain the security strength because the real-time playback constraints in the receiver. The receiver can be acknowledged by a new key sending through a reliable secure channel. As shown in Figure 6, there are two protocol data units for sending new key in secure channel and sending transposed video packet respectively.

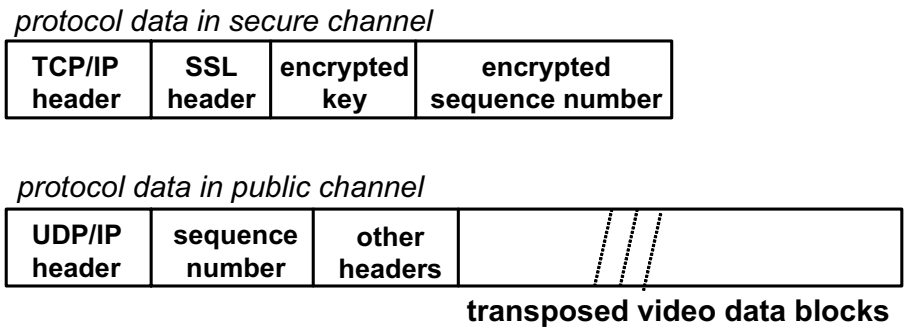

Figure 6. Protocol data in transposition control with feedback of buffer occupancy

Now, we list a sample table in Table 1 to indicate the number of permutations (i.e. keys) for different number N of data blocks to encrypt the packet. We assume that a video streaming application is running our security scheme. While the number N is running at 12 and receiver's playback buffer is running below the low watermark B_l, the sender will pick up a new key with smaller N (i.e. 11).

Then, if the playback buffer is running higher than the high watermark B_h, the sender will again pick up a new key with larger N (i.e. 12) to strengthen security without impact to real-time video streaming. Moreover, if the playback buffer is running within low watermark and high watermark, sender can apply a new key with the same N (i.e. 12) to enhance the security, while those keys have been applied before were small values of N.

N	*N*!	*N*	*N*!
16	2.09228E+13	**8**	40320
15	1.30767E+12	**7**	5040
14	87178291200	**6**	720
13	6227020800	**5**	120
12	479001600	**4**	24
11	39916800	**3**	6
10	3628800	**2**	2
9	362880	**1**	1

Table 1. A sample table for different number *N* of data blocks in a packet and its number of transpositions

4. Performance and Experiments

In this section, we will examine the performance of the proposed security method by theoretic analysis. Besides, we present a preliminary experimental result for transposition cipher to preliminarily evaluate the performance for this security method.

4.1. Theoretic Performance Evaluation

We will not only evaluate our proposed security scheme by proving if it is vulnerable to break, but also examine if the security is too complex to be applied to real-time video streaming. We firstly define a formula to represent the whole original video streaming as shown in formula (2):

$$stream=b_{0,1}b_{0,2}....b_{0,N_0}b_{1,1}b_{1,2}....b_{1,N_1}....b_{0,N_0}b_{i,1}b_{i,2}....b_{i,N_i}b_{i+1,1}\ .... \quad (2)$$

In formula (2), N_i represents the total number of block in a packet with sequence number i in the video stream.

Sender may select a secrete key *key* according to the buffer occupancy feedback of playback buffer. Then, the permutation function $E(i, key)$ will indicate a block number and the indicated block will be moved to the new position i. The secrete key may vary due to the buffer occupancy feedback. The new encrypted stream is listed as follows:

$$nstrm=b_{0,E(1,key)}b_{0,E(2,key)}....b_{0,E(N_0,key)}b_{1,E(1,key)}b_{1,E(2,key)}....b_{1,E(N_1,key)}....b_{i,E(1,key)}b_{i,E(2,key)}....b_{i,E(N_i,key)}b_{i+1,E(1,key)}..... \quad (3)$$

Let's take cryptanalysis for the transposition control with feedback of buffer occupancy. We define the size of the video stream is totally P packets and the maximum number of data blocks to compose a packet is N_{max} (i.e. the maximum number of blocks to transpose). For example, the N_{max} is 16 in Table 1. Therefore, we have the range of N_i shown in the formula (4).

$$2 \le N_i \le N_{max} \quad (4)$$

Besides, the number of different secrete keys which has been applied in video streaming services is defined as K.

Because the changes of secrete key are all applied to packets, the number K of key changes is not larger than the number P of total packets in the video stream.

$$1 \le K \le P \tag{5}$$

Then through formula (2) to (5), we can estimate the possible range for the total combinations of the transposition cipher with feedback control of buffer occupancy. We define the total combination as S and it will depend on the number K of key changes and the number N_i of blocks to encrypt a packet with sequence number i. The range of the combinations S is listed in formula (6):

$$2^K \le S \le (N_{max}!)^K \tag{6}$$

Now, we will demonstrate the performance of the proposed security scheme by taking a sample from a Video On Demand service. We assume the length of a video file of MPEG-1 is 640MB and the packet length is 1KB. Then, the total number of packets are 640000 (i.e. P=640000). Besides, N_{max} is assumed as 16 and we pick up a very small number 64 of key changes (i.e. K=64) in 640000 packets. Then, according to formula (6), the total number of combinations S of transpositions will be ranged from 2^{64} to $(16!)^{64}$. If the validation time T for every possible transposition is 1 *ns*, the average time to recover the entire video stream will cost at least 292471 years for the eavesdropper to break without the knowledge of the key changes.

4.2. Preliminary Experimental Result

We simply perform an experiment on the time cost of transposition cipher to preliminarily demonstrate that the propose security scheme won't seriously degrade playback quality of real-time video streaming service.

Computer	→Desktop AMD-Athlon XP 2000+ AMD-Athlon XP 1500+ INTEL Celeron 1800MHz/500MHz →Notebook IBM R30 (Intel Mobile Celeron 896MHz)
Language	C

Table 2. Experimental equipments and language tool.

We wrote a program to simulate the data transpositions, which is shown in Figure 3, by different number of blocks in a packet. Without loss of generality, the number N of blocks ranged from 2 to 128 with binary exponents. Besides, we recorded the time spent for different number N of blocks apply in a packet. The length of a packet was 1024 bytes. This test was equally performed on different machines and their averages were shown in Table 2.

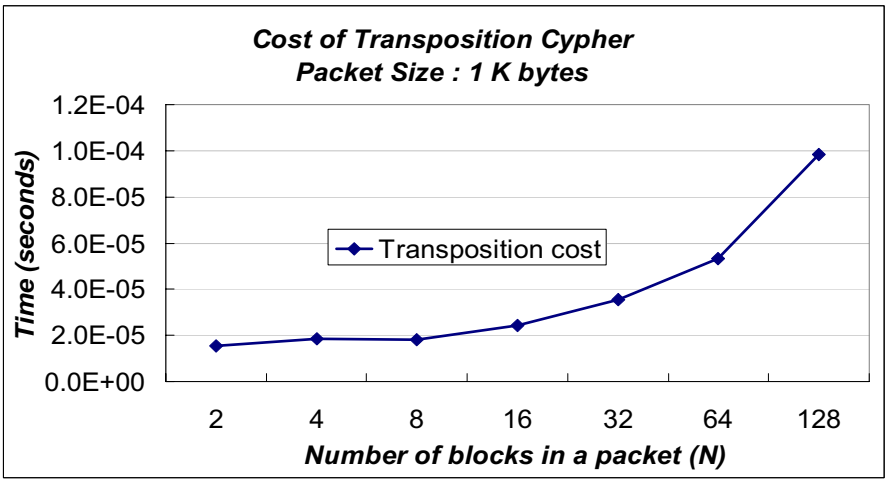

Figure 7. Transposition cost of different *N* blocks in a packet (1K bytes)

The preliminary results are shown in the Figure 7. The average time of this transposition cipher ranges from 15.3 *ns* to 98.2 *ns* due to different *N* applied. They are very small values. Therefore, we believe that our proposed security scheme using transposition control with feedback control of buffer occupancy won't seriously degrade the real-time video streaming service.

5. Conclusions and Future Work

The objective of this research is trying to provide an economically and effectively security scheme for real-time video streaming in the public network. This application-level security scheme is not dependent on the coding scheme of the media. Internet applications can apply this security method to deliver the media stream which preserves privacy and commercial value.

Our security method won't spend much resource such as bandwidth, CPU and memory. Thus, it won't degrade the playback QoS of video streaming. If all the encrypted video packets are copied from the Internet eavesdroppers, they can not playback the video packets without the secure secret key. Even if encrypted packets are stored by the hackers, they cannot recover the video in a short time. While the video are recovered by the hackers, the video may not preserve its original value.

This research considers the characteristics of delivering real-time video streaming and propose a security scheme. We have examined the performance of this proposed method by theoretic analysis and preliminary experimental results. In the near future, we will deploy this method to a video surveillance application [7] to practically prove the effectiveness of this security scheme. Moreover, this method will be applied to the Video on Demand system [12] and Video Conferencing system [9]. We will further justify their performance while

this security scheme is applied to those video streaming services, which even require more strictly real-time constraints.

References

[1]. Fred T.Hofstetter,Third Edition, Multimedia Literacy, McGraw-Hill Irwin,2001.

[2]. Han-Chieh Chao, T.Y.Wu and Jiann-Liang Chen, “Security-enhanced packet video with dynamic multicast throughput adjustment”, International Journal of Network Management, 11:147-159, 2001.

[3]. Yongcheng Li, Zhigang Chen, See-Mong Tan, Roy H. Campbell, “Security-enhanced MPEG player”, International Workshop on Multimedia Software Development, 1996. Proceedings., 25-26 March 1996.

[4]. Kunkelmann, T., Reinema, R.” A scalable security architecture for multimedia communication standards”, IEEE International Conference on Multimedia Computing and Systems '97. Proceedings. Pages:660 - 661, 3-6 June 1997.

[5]. Lei Tang, “Methods for encrypting and decrypting MPEG video data efficiently” Proceedings of the fourth ACM international conference on Multimedia, February 1997.

[6]. Tosun, A.S., Feng, W.-C., “Efficient multi-layer coding and encryption of MPEG video streams”, IEEE International Conference on Multimedia and Expo, 2000. ICME 2000. Pages:119 - 122 Volume: 1 , 30 July-2 Aug. 2000.

[7]. Chia-Hui Wang; Chang, R.-I.; Jan-Ming Ho, ”An effective communication model for collaborative commerce of web-based surveillance services,” E-Commerce, 2003. CEC 2003. IEEE International Conference on, 24-27 June 2003, Page(s): 40 -44. NSC 92-2213-E-424-002.

[8]. Chia-Hui Wang, Ray-I Chang, Jan-Ming Ho, Shun-Chin Hsu, "Rate-Sensitive ARQ for Real-Time Video Streaming," GLOBECOM’03, IEEE 2003 Global Communications Conference, 1 - 5 December 2003. NSC 91-2213-E-001-026 and NSC92-2218-E-130-004.

[9]. Jan-Ming Ho, Jie-Yong Juang, Chia-Hui Wang, “Extended Services of ASIS VConf.,” Fourth International Symposium on Real-Time And Media Systems (RAM 98), Taiwan, Sept. 1998, pp.271-278.

[10]. William Stallings, “Cryptography and Network Security, Principles and Practices”, Prentice Hall, 2003.

[11]. I. Rhee, S.R. Joshi, ”Error Recovery for Interactive Video Transmission over the Internet,” IEEE Journal on Selectied Areas in Communications, Vol.18, No.6, June 2000.

[12]. C. H. Wang, J. Ho, S. Hsu, et.al., ”Design and Implementation of an Interactive True VOD System on ADSL with Scarce Resources at the Set-top Box,” International Computer Symposium on Workshop on Computer Networks, Internet, and Multimedia, Taiwan, 2000.

[13]. Chui Sian Ong, Klara Nahrstedt and Wanghong, "Qualily of Protection for Mobile Multimedia Applications”, IEEE International Conference on Multimedia and Expo, 2003. ICME 2003. Pages: II137-II140, 2003.

[14]. Hui Zhang, Domenico Ferrari, “Rate-Controlled Service Disciplines”, Journal of High Speed Networks, 3(4), 1994.

[15]. H. Schulzrinne, S. Casner, R. Fredrick, and V. Jacobson, “RTP: a transport protocol for real-time applications,” Tech. Rep. RFC 1889, Internet Engineering Task Force, Jan. 1996.

[16]. A. Menezes, P. van OOrschot, and S. Vanstone, “Handbook of Applied Cryptography”, CRC Press, 1996.

[17]. Chia-Hui Wang, Jan-Ming Ho, Ray-I Chang, Shun-Chin Hsu, “A Control-Theoretic Mechanism for Rate-based Flow Control of Real-time Multimedia Communication,” Multimedia, Internet, Video Technologies 2001 (MIV 2001) of WSES/IEEE International Multi-conference, Sept., 2001.