The Application of Data Source Authentication Protocol to Video Multicast

Shijie Zhao
School of Computer Science and Technology
Beijing Institute of Technology
Beijing, China 100081
bitdongba@gmail.com

Zhirun Liu
School of Computer Science and Technology
Beijing Institute of Technology
Beijing, China 100081
lzreee@126.com

Liehuang Zhu*
School of Computer Science and Technology
Beijing Institute of Technology
Beijing, China 100081
*liehuangz@bit.edu.cn

Rufeng An
School of Computer Science and Technology
Beijing Institute of Technology
Beijing, China 100081
jairus2008@126.com

***Abstract*—An efficient packet-injection resistant data source authentication protocol for group communication(EPJRSA) was proposed to authenticate data source. It can recognize which packets are injected by attacker and delete them. But it cannot recover the lost packets. So an advanced protocol based on EPJRSA protocol is proposed to solve this problem. This advanced EPJRSA protocol is used to build a video broadcast system. The system has the ability of packet loss and injection resistant. And it can guarantee the security of both video data and authentication information at the same time. Experiments are set to verify the correctness of the system.**

Keywords- data source authentication; application on video multicast; packet-lost; packet-injection

I. INTRODUCTION

With the improvement of Interment technology, videos can be accessed and transported in an open way through networks. This brings the explosive growth of users, meanwhile the giant challenge to video security. Video security problems are intimately concerned all the time. In this environment, video signal can be broadcasted as IPTV via Internet. Once the signal is tampered illegally, it will cause serious influence, and the number of users influenced will increase exponentially. Meanwhile, the video terminal will not only confined to television， but also can be PC, mobile phone, MP4. For these reasons, the security problems of video become more and more important and wildly concerned [1-2]. Data source authentication technology is the main course of group security research. Using advanced real-time packet-loss resistant data source authentication protocol to authenticate data source, can efficiently resist packet loss and injection. This can ensure that the data packets clients received are sent by the legal server. It can ensure that attackers cannot forge or modify the data.

UDP transfer protocol has the advantages of fast transfer speed and short transfer delay. But it cannot resist packet-loss.

Park, Chong and Siegel [3] presented a packet-loss resistant protocol SAIDA, which can efficiently resist the packet-loss within its allowable loss range. However, this protocol can't help with the packet-injection attack.

Hong Tang and Liehuang Zhu [4] proposed an efficient packet-injection resistant protocol named EPJRSA. Using merkle hash tree, it can find packets injected before verifying the signature. When the packet-injection is smaller than 50%, the injection attack can be resisted. Combining with information dispersal algorithm, the advanced protocol can resist packet loss and injection at the same time.

Our system uses advanced EPJRSA protocol to encode the video broadcast data. It can not only do real-time authentication to video data source, but also can guarantee the security of video information, and play the original video, which is a safe video broadcast.

II. THE ADVANCED EPJRSA PROTOCOL

The original EPJRSA protocol is the basement of the system.

Divide the video into packets by frame. And use the EPJRSA protocol to encode the packets. Set the data to be processed be D.

1) Define two parameters : N and Ns.
2) Divide D into N packets, and group this N packets into Ns groups. Group.
3) Construct the merkle hash tree of D.
4) Compute the signature and construct the packets to be sent.

Assuming N =16, Ns =4, then we'll have 4 different merkle hash trees to be constructed. Each hash tree regards as a group. For example, the merkle hash tree of Group 3 is as **Figure 1**:

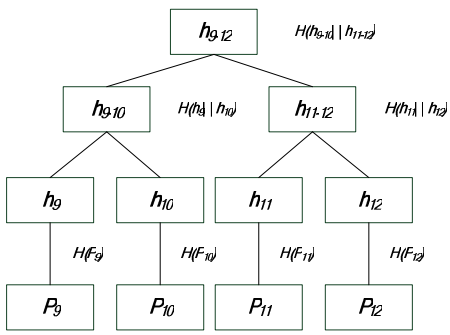

Figure 1. An example of merkle hash tree

The signature [5] and the authentication information are computed as follow:

$$\sigma = SIGN_{sender}(h^1 \| h^2 \| \ldots \| h^{N_S})$$

$$F = h^1 \| h^2 \| \ldots \| h^{N_S} \| \sigma$$

$$S = (s_1, s_2, \ldots, s_N) = Disperse(F, m, N)$$

Set P_i' be the authentication information packet of data packet P_i, then $P_i' = \{i, P_i, s_i, D_i\}$.

D_i is made up of the hash values of all the brother nodes which come from the root node to the leaf node of the hash tree corresponded to P_i.

As in Figure 1, P_9 corresponds to $D_9 = \{h_{10}, h_{11-12}, h_{13-16}\}$, $P_1' = \{9, P_9, s_9, h_{10}, h_{11-12}, h_{13-16}\}$.

That above is the EPJRSA protocol. Some changes are made to it. Then it can resist both packet-loss [6] and packet-injection.

In order to ensure the security of video information at the same time, we add the original video data D to the front of F. Use Information Disperse Algorithm(*IDA*) as $Disperse()$.

$$\sigma' = SIGN_{sender}(h^1 \| h^2 \| \ldots \| h^{N_S})$$

$$F' = D \| h^1 \| h^2 \| \ldots \| h^{N_S} \| \sigma$$

$$S' = (s_1', s_2', \ldots, s_N') = IDA(F', m, N)$$

Then we set the format of packet to be sent as follow:

$$P_i' = i \| P_i \| D_i \| s_i .$$

D_i is made up of the hash values of all the brother nodes which come from the root node to the leaf node of the hash tree corresponded to P_i.

III. Application to Video Multicast

In order to ensure the correctness of the advanced EPJRSA protocol and the system [7], we set up another two protocols for comparison, SAIDA, EPJRSA.

We have three ends in the multicast communication model [8], the server, the adversary, the clients. The system diagram is as **Figure 2**.

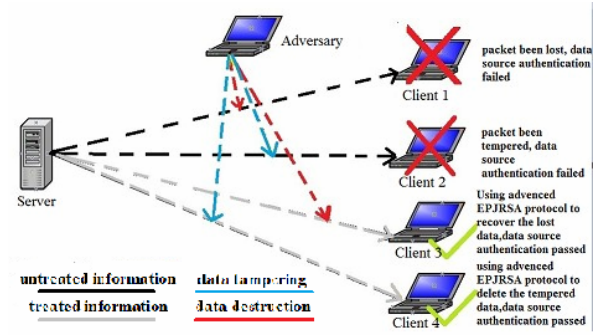

Figure 2. The system diagram

The black line stands for untreated information, the gray one means treated information, the blue for data tampering, and the red data destruction. If the information haven't been treated, when they are tempered or destructed, clients cannot authenticate the data source. After treating the information, using advanced EPJRSA protocol, clients can recover the original data and complete the authentication.

The server preprocesses the video data stream and computes the authentication information. It can choose SAIDA, EPJRSA or the advanced EPJRSA protocol to encode the video data and the authentication information. It also can choose to encode the whole video source file or only encode the authentication information for decreasing the cost of transmission. And then broadcast them. The server management interface is as **Figure 3**.

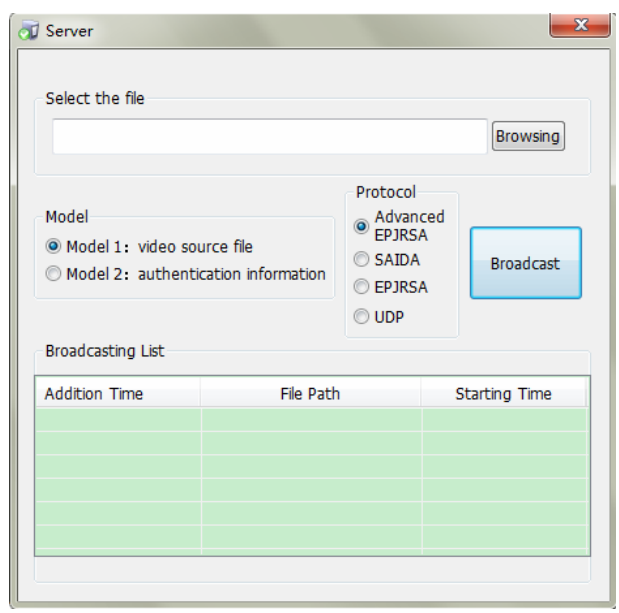

Figure 3. Server Management Interface

The adversary intercepts the data stream and

launches malicious attack(e.g. deleting or injecting some packets randomly according to the rate of packet-loss and the rate of packet-injection), then resend them to the client. The software interface of adversary end is as **Figure 4**.

The client receives data stream [9-11], deletes the forged packet, and recovers the original data stream by using SAIDA, EPJRSA or advanced EPJRSA protocol, and then verifies the signature and play the accurate video. Based on the protocol chosen, it shall get different result. The client software module is as **Figure 5**.

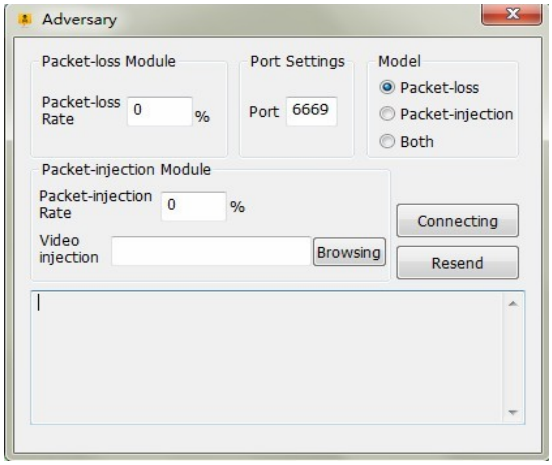

Figure 4. Adversary Interface

Figure 5. Client Software Module

A. *The link libraries we invocated*

- We invoke the Crypto++ library to realize the IDA algorithm. It includes the Rabin's Information Dispersal Algorithm and the Rabin's Information Recovery Algorithm.
- We invoke the Miracl library to realize the HASH function and the RSA signature scheme.
- We invoke the JM program to encode the videos from .yuv to .264 [12] at the server, and to decode the videos from .264 to .yuv at the client. This ensures the instantaneity of the video multicast communication.

B. *The preprocessing to the video*

To reduce the delay of the clients, we preprocess the videos before broadcasting them. Divide a video into several smaller videos by a video-divide program. This can ensure that the clients do not have to wait for receiving the whole video when he wants to play it. After receiving several small videos, they can be played one by one with our non-stop-playing system.

IV. Performance Test of The Video Multicast System

Here in this system, preset $m=N/2$ in $IDA(F,m,N)$ that means it can recover the data when the percent of packet-loss is lower than m.

We have designed five experiments below for comparison:

A. *The SAIDA protocol can resist packet-loss*

Set the rate of packet-loss forty-percent and the rate of packet-injection zero-percent in the adversary end. And choose the SAIDA protocol at the server and the client.

When receiving the data packets, clients recover the original packets by the SAIDA protocol. And then they will authenticate the data origin. The result will be success, and the video can be played as **Figure 6**:

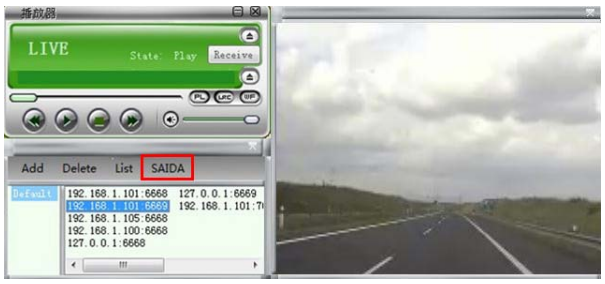

Figure 6. Packet-loss test on SAIDA model

B. *The SAIDA protocol cannot resist packet-injection*

The adversary end sets the rate of packet-loss zero-percent, and the rate of packet-injection forty-percent. Choose the SAIDA protocol at the server and the client.

The result will be failed, and the video cannot be played as **Figure 7**:

Figure 7. Packet-injection test on SAIDA model

C. *The EPJRSA protocol can resist packet-injection*

The adversary end sets the rate of packet-loss zero-percent, and the rate of packet-injection forty-percent. Choose the EPJRSA protocol at the server and the client.

The result will be success, and the video can be played as **Figure 8**:

Figure 8. Packet-injection test on EPJRSA model

D. *The EPJRSA protocol cannot resist packet-loss*

The adversary end sets the rate of packet-loss forty-percent, and the rate of packet-injection zero-percent. Choose the EPJRSA protocol at the server and the client.

The result will be failed, and the video can be played as **Figure 9**:

Figure 9. Packet-loss test on EPJRSA model

E. *The Advanced EPJRSA protocol can resist packet-loss and packet-injection at the same time*

The adversary end sets the rate of packet-loss forty-percent, and the rate of packet-injection forty-percent. Choose the advanced EPJRSA protocol at the server and the client.

The result will be success, and the video can be played as **Figure 10**:

Figure 10. Packet loss and injection test on advanced EPJRSA model

V. CONCLUSIONS

We use the advanced EPJRSA protocol to build up a video broadcast system. This system is made up of server, adversary end and client. The adversary end can stimulate the packet- loss and illegal attack while transferring. Comparing the advanced EPJRSA protocol with EPJRSA protocol, the result states that the system can resist the adversary's attack, do the real-time authentication of broadcast data source, and play the legal video. This also guarantees the correctness of video source and information and protect the benefit of both video servers and users.

REFERENCES

[1] V Lehtovirta, F Lindholm, Security Key Management In IMS-Based Multimedia Broadcast And Multicast Services (MBMS) US Patent 20,120,027,211, 2012

[2] D Phan, D Pointchecal, M Strefler, Security notions for broadcast encryption, Ctyptography and Network Security, 2011-Springer

[3] Park Y, Cho Y. The eSAIDA stream authentication scheme[A]. Proceedings of ICCSA 2004[C]. Assisi, Italy, 2004. 799-807.

[4] Tang H, Zhu L H, Li J, Cao R Q, Efficient packet-injection resistant data source authentication protocol for group communication[J].Journal on Communications, Vol.29 No.11A, November 2008. Pp.96-100.

[5] Park J M, Chong E K P, Siegel H J. Efficient Multicast Packet Authentication Using Signature Amortization[A]. Proceedings of 23rd IEEE Symposium on Security and Privacy[C]. Oakland, California, USA, 2002. 227-240.

[6] Perrig A, Ganetti R, Tygar J, et al. Efficient authentication and signing of multicast streams over lossy channel[A]. Proceeding of 21st IEEE Symposium on Security and Privacy[C]. Berkeley, California, USA, 2000. 56-73.

[7] Wang C K, Chen A. Immediate data authentication for multicast resource constrained networks[A]. Proceedings of ACISP 2005[C]. Brisbane, Australia, 2005. 133-121.

[8] Liu D, Ning P. Broadcast authentication for distributed sensor networks[J]. ACM Transactions in Embedded Computing System, 2004. 3(4):800-836.

[9] Miner S, Staddon J. Graph-based authentication of digital streams[A]. Proceedings of 22nd IEEE Symposium on Security and Privacy[C]. Oakland, California, USA, 2001. 233-246.

[10] Challal Y, Bouabdallah A, Bettahar H. H2A: hybrid hash-chaining scheme for adaptive multicast source authentication of media-streaming[J]. Computer & Security, 2005. 24(1):57-68.

[11] Lysyanskaya A, Tamassia R, Triandopoulos N. Multicast authentication in fully adversary networks[A]. Proceedings of 24th IEEE Symposium on Security and Privacy[C]. Berkeley, CA, USA, 2003. 241-253.

[12] Bi H J, Wang J. New generation video compassion standard——H.264/AVG(The second edition) [M]. Beijing: The People's Posts and Telecommunications Press,2009.