How to Authenticate Real Time Streams Using Improved Online/Offline Signatures

Chong-zhi Gao[1] and Zheng-an Yao[2]

[1] Information, Machinery and Electronics College,
Guangzhou University,
GuangZhou 510000, China
[2] College of Mathematic and Computational Science,
Sun Yat-Sen University,
GuangZhou 510275, China

Abstract. Providing authentication protocols for real time streams is a challenging task. This is because the authentication rate is very important for real time streams, whereas it is usually a bottleneck. Using improved online/offline signatures and hash chain techniques as tools, our proposed protocol greatly reduces the online computational and communicational cost and thus is more applicable to authenticate real time streams.

Keywords: Stream Authentication, Real Time Streams, Online/Offline Signatures.

1 Introduction

A digital stream is a (potentially infinite) sequence of bits that a sender transmits to a receiver. With the growth of the Internet and the popularization of electronic commerce, there are more and more applications that need data transmissions such as live video/radio broadcastings and real time stock quote systems. In the data transmission, people are usually concerned with the following issues:

1) *Privacy:* the sender keeps information secret from those who are unauthorized to see it.
2) *Integrity:* the receiver can ensure that information has not been altered by any unauthorized parties.
3) *Authenticity:* the receiver can corroborate that the received information is sent by a certain party.
4) *Non-repudiation:* the receiver can prove to a third party that the information is sent by a certain party.

Over the years, researchers have proposed various techniques to achieve these objects. For example, public encryption schemes[21, 3, 13] were proposed to ensure the privacy of data and signature schemes[2, 21, 20] were proposed to ensure the authenticity.

Y.G. Desmedt et al. (Eds.): CANS 2005, LNCS 3810, pp. 134–146, 2005.

1.1 Authentication for Real Time Streams

In this paper, our goal is to provide authenticity as well as integrity and non-repudiation for real time streams. A real time stream is quite different from a non-real time stream since the sender cannot be expected to obtain the entire stream before or on sending the stream. Thus, the sender can only buffer few packets while transmitting these streams. In addition, the authentication rate must be higher than the stream generation rate while this is not required for non-real time streams.

1.2 Related Work

A trivial authentication method is to sign each packet[7]. The sender first splits the stream into packets and signs each packet one by one. The receiver then verifies these signatures after he/she receives the packets and their corresponding signatures. However, this method has its disadvantage since every packet requires a sign/verify computation and thus the computational cost is quite heavy. In addition, adding a signature to each packet will greatly increase the communication overhead.

In 1997, Gennaro and Rohatgi [7] proposed two paradigms for stream authentications. In the paradigm for streams that can be known in advance by the sender, they use hash chain techniques and signature techniques to authenticate streams. In this paradigm, although a signature is amortized over several packets, the computational cost is still high since a signature operation is very inefficient. In the paradigm for streams that can not be known in advance, they employ one time signatures introduced in [12, 14]. This paradigm results in a large communication overhead since the signature size and the key size of one-time signatures are very large.

In 1998, Wong and Lam [23] proposed a tree chaining technique to authenticate streams. Their construction is robust to any number of losses in streams. However, the communication overhead per packet is quite large (even larger than the size of a digital signature) and thus is not practical.

Miner and Staddon [15] proposed a graph-based authentication protocol in 2001. In their transmission model, each packet is assumed to be lost independently with the same probability and the protocol is designed based on this probability.

There are other authentication protocols for streams such as Perrig et al.'s EMSS and TESLA scheme[18], Wu et al.'s object-based scheme[24] and Pannetract and Molva's EC scheme[17]. Some of them, e.g., the TESLA scheme in [18] and the objected based scheme in [24], do not offer non-repudiation.

1.3 Contribution

In previous works, using ordinary signature schemes such as RSA[21], DSA[16], FFS[6, 5] or eFFS[23] will result in heavy computational cost[7, 15, 24, 17] or a large communication overhead[23]. However, the so-called online/offline