Design of Fully Deniable Authentication Service for E-mail Applications

Lein Harn and Jian Ren

***Abstract*— Secure Electronic Mail (e-mail), such as PGP and S/MIME, uses digital signature to provide message authentication, which also provides the undesired non-repudiation evidence of the message sender. In this paper, we introduce a fully deniable e-mail authentication service. Our design can be easily integrated into the current PGP and S/MIME to provide message authentication without non-repudiation evidence. This feature can protect personal privacy of the message sender in most personal communication.**

***Index Terms*— E-mail authentication, deniable authentication, non-repudiation.**

I. Introduction

ELECTRONIC mail (e-mail) is one of the most important and widely used network applications. It has been used in communications between individuals, business organizations and governmental agencies around the world. The vulnerability of underlying network demands secure e-mail solutions.

In a secure e-mail application, the following two security services must be considered:

- *Message confidentiality*: Message confidentiality assures the sender that the message can be read only by the intended receiver.
- *Message authenticity*: Message authenticity assures the receiver that the message was sent by a specified sender and the message was not altered *en route*.

Currently, there are two widely used secure e-mail solutions, Pretty Good Privacy (PGP) [1] and S/MIME [2]. Both solutions utilize a combination of conventional symmetric-key techniques and modern asymmetric-key (i.e. public-key) techniques to provide message confidentiality and message authentication. The recent research on securing emails have been largely focused on the design of new cryptographic protocols to enhance confidentiality [3]–[6]. Although the objective of message authentication can be achieved by using digital signatures, it also creates a potential privacy threat. The receiver can pass the message and the corresponding digital signature to a third party without the permission of the sender. The digital signature can be verified by any third party. This design inherently provides non-repudiation evidence to the message sender which is not required and even not desired in most e-mail applications.

Manuscript received October 27. 2007. The associate editor coordinating the review of this letter and approving it for publication was C.-K. Wu. This research was supported in part by the National Science Foundation under Award CNS-0716039.

L. Harn is with the Department of Computer Science and Electrical Engineering, University of Missouri - Kansas City, MO 64110, USA (e-mail: harnl@umkc.edu).

J. Ren is with the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA (e-mail: renjian@egr.msu.edu).

Digital Object Identifier 10.1109/LCOMM.2008.071793.

There are two types of deniability: plausible deniability and full deniability [7], [8]. For plausible deniability, the sender can only deny transmission of a particular message, however, he is unable to deny the fact that he has communicated with another user. While full deniability allows the message sender to totally deny that he has communicated with another user.

In this paper, we propose a fully deniable secure e-mail service using the cryptographic functions supported by both PGP and S/MIME. The main idea is that in our design, only the message sender and the message receiver have the ability to generate the transmitted message. Therefore, the message receiver knows that the message was generated by the message sender if the receiver did not generate it.

Deniable authentication is straightforward for interactive Diffie-Hellman (DH) key agreement [9]. A non-interactive deniable authentication using RSA is introduced in [7]. The advantage of our design is that it works for any public-key algorithms. The message sender and message receiver have the flexibility to use any public-key algorithms, such as the combination of RSA encryption and DL-based digital signature. Designated verifier signature (DVS) [10] can also provide deniable authentication service. However, DVS is still an on-going research [11], while our design uses existing public-key algorithms supported by PGP and S/MIME.

The rest of the paper is organized as follows. In the next section, we give some preliminaries for this paper. Our proposed cryptographic design of secure email is introduced in section III. The features and security of our design is given in section IV. We conclude in section V.

II. Preliminary

A. Review of PGP and S/MIME Solutions

In PGP and S/MIME applications, each user is assumed to have two pairs of public and private keys selected for long-term use. One pair of keys is used for message encryption and the other pair is used for digital signature. It is assumed that the public keys of all communication partners have already been securely stored in each user's public-key ring.

Message confidentiality using digital envelope: A *digital envelope* is a technique used by the sender to transmit the message in such a way that only the intended receiver can read the content of the message. The sender first randomly selects a secret session key and uses this secret key to encrypt message. Then, the sender encrypts this secret session key with the receiver's public key using any public-key encryption algorithm. After receiving the encrypted message, the receiver first uses its private key corresponding to the public key to uncover this secret session key. Then, the receiver uses the secret session key to decrypt the ciphertext.

Message authentication using digital signature: PGP and S/MIME both use digital signature to provide message authentication. The message sender uses its private signing key to generate a digital signature on the message digest. The digital signature is attached along with the message and is sent to the receiver. The receiver uses the sender's public key to verify the digital signature. One potential security problem in using digital signature to provide message authentication is that, without consent from the message sender, the receiver can pass the message and its digital signature to a third party. Since the digital signature can provide non-repudiation evidence that can be verified by anyone, this poses a *security threat* to the sender's privacy.

Message confidentiality and message authentication: PGP and S/MIME provide this service by using both, digital signature and digital envelope for the message.

B. RSA digital signature scheme

RSA signature scheme was introduced in [12]. It includes

- *Key generation algorithm*: let p and q be two large primes, $n = p \times q$, d be a private key satisfying $e \times d = 1 \bmod \phi(n)$, where $\phi(n) = (p-1) \times (q-1)$. The public key is (e, n) and the private key is (d, p, q).
- *Signature generation algorithm*: the signature σ of a message digest m is defined as $\sigma = m^d \bmod n$.
- *Signature verification algorithm*: the verifier checks the equation $m = \sigma^e \bmod n$.

C. ElGamal digital signature scheme

ElGamal signature scheme was introduced in [13]. It includes

- *Key generation algorithm*: let p be a large random prime, g be a generator, both made public. For a random private key $x \in \mathbb{Z}^*_{p-1}$, the public key is $y = g^x \bmod p$.
- *Signature generation*: to sign a message digest m, one chooses a random $k \in \mathbb{Z}^*_{p-1}$, computes $r = g^k \bmod p$, and solves s from the linear equation $m = xr + ks \bmod (p-1)$. The signature is defined as the pair $\sigma = (r; s)$.
- *Signature verification*: to verify the signature, one checks the equation $g^m = y^r \cdot r^s \bmod p$.

III. Our New Design of Deniable Authentication

We propose a new design to provide deniable authentication. Similar to PGP and S/MIME, we assume that each user has two pairs of public and private keys. One pair of keys is used for message encryption and the other is used for message authentication. These keys are for long-term use. We also assume that the public keys of all communication partners are securely stored in each user's public-key ring.

In this paper, we only describe the cryptographic design. The details of message format and delimiters are left open for possible integration with current implementations of PGP and/or S/MIME. We describe our design in the scenario when Alice (sender) wants to send an e-mail message m to Bob (receiver). Let (x_A, y_A) be the private/public key pair of Alice for message signing and (x_B, y_B) be the private/public key pair of Bob for message encryption.

The communication processing between Alice and Bob, shown in Fig. 1, includes two steps:

1) Alice randomly selects a one-time secret key k and constructs a digital envelop $c = pE_{y_B}(k)$ by encrypting k using Bob's public key y_B. Then computes $\sigma = \mathsf{sign}_{x_A}(c)$ using her private key x_A, and the one-way hash value $H_k(m\|T)$ of input $m\|T$ using secret key k, where m is the message, T is the timestamp and $\|$ represents the message concatenation. A sends $\{\sigma, c, H_k(m\|T), m, T\}$ to B. Note that, in our design, the digital signature sign_{x_A} is applied to the message c directly. We will explain this further in the next section.
2) After receiving $\{\sigma, c, H_k(m\|T), m, T\}$, Bob checks whether σ is a legitimate signature of c using Alice's public key y_A. If successful, Bob uses his private key x_B to decrypt the digital envelop and recover the random secret $k = pD_{x_B}(c)$, validates the timestamp T, and authenticates the message m by computing $H_k(m\|T)$ and comparing this value with the received value. If all verifications are successful and the computed value is identical to the received one, then the message is authenticated; otherwise, the authentication fails.

IV. Feature Discussion and Security Analysis

Since public-key computations are more time consuming than symmetric-key and one-way hash computations, our performance evaluation is limited to public-key computations. Alice needs two public-key computations: one for creating the digital envelop and the other for generating the digital signature. Bob also needs two public-key computations: one for validating the digital signature and one for deciphering the digital envelop. The computational complexity for our proposed design is identical to the message confidentiality and message authentication used in PGP and S/MIME.

However, our design can provide two additional security features: *specified authentication* and *deniable authentication*. Specified authentication enables the message sender to specify a message receiver to authenticate the transmitted message. Deniable authentication enables the message sender to deny the generation of this message. In this section, we will discuss these two features.

A. Specified Authentication

In our design, $\{\sigma, c, H_k(m||T), m, T\}$ is transmitted from A to B in a public channel. However, to authenticate the message m with $H_k(m||T)$, a secret key k is needed. From Fig. 1, we know that to derive the secret key k from the cipher text c, Bob's private key x_B is needed. Therefore, only Bob can derive this secret key k and authenticate the message.

B. Deniable Authentication

Deniability is provided if we can show that both the message sender and the intended message receiver can generate a valid transmitted message, $\{\sigma, c, H_k(m||T), m, T\}$. If we can prove this property, the sender can falsely deny generating of the transmitted message by claiming that it was generated by the message receiver. However, if the message is really generated

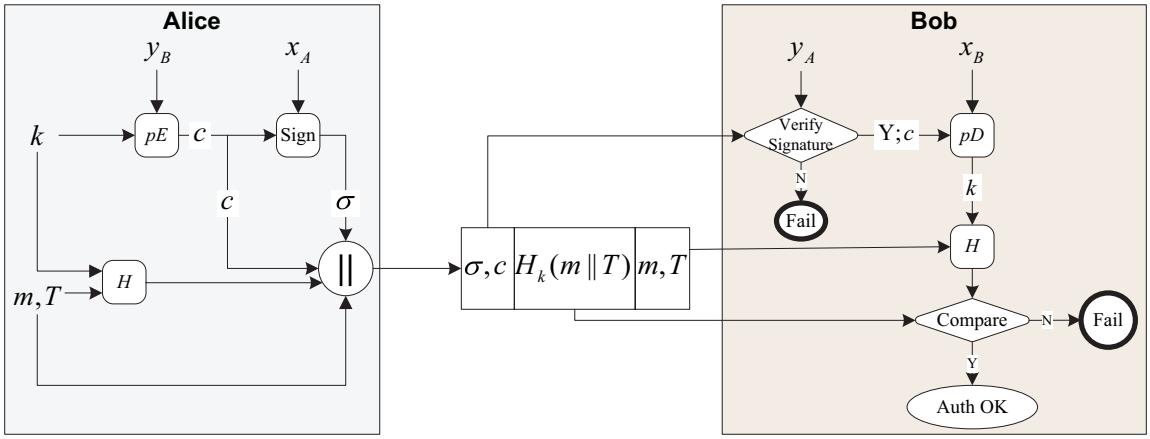

Fig. 1. Deniable message authentication.

by the message sender, then the message receiver knows that this message was generated by the sender. Therefore, the message receiver can authenticate this message.

In the previous section, we have described how Alice uses her private key x_A to generate a valid pair of $\{\sigma, c\}$. Since Alice's private key x_A is needed in this procedure, only Alice can follow the procedure shown in Fig. 1 to generate a valid pair of $\{\sigma, c\}$. Thus, Alice can generate the valid transmitted message $\{\sigma, c, H_k(m||T), m, T\}$.

Next, we will show how the message receiver Bob can also generate $\{\sigma, c, H_k(m||T), m, T\}$. It is well-known that if a digital signature is applied to the message itself directly, then all public-key digital signature algorithms are existentially forgeable. In our design, we use the existential forgeability to provide deniability. Instead of digitally signing a message digest as suggested in all existing digital signature algorithms, *we propose that the digital signature is applied to the message directly*. In the following, we explain the methods of existential forgery of two well-known digital signature algorithms: the RSA scheme [12] and the original ElGamal scheme [13].

Existential forgery of RSA signature: It is easy to see that by first randomly choosing the signature σ and computing the corresponding value c, satisfying $c = \sigma^e \bmod n$, then $\{\sigma, c\}$ is a valid pair of signature and message that can be verified successfully.

Existential forgery of ElGamal signature: Let $e \in [1, p-2]$ and $v \in [1, p-2]$, if we let $r = g^e y^v \bmod p$, and $s = -rv^{-1} \bmod p - 1$, it is easy to see that (r, s) is a valid signature for the message $c = es \bmod (p-1)$.

B (Bob) can first compute a valid pair of $\{\sigma, c\}$ following the existential forgery. Then, Bob uses his private key x_B to decrypt c and obtain the secret value k. Thus, for any given message m, Bob can also compute a valid $\{\sigma, c, H_k(m||T), m, T\})$.

It is computationally infeasible for any adversary to compute a valid $\{\sigma, c, H_k(m||T), m, T\})$. We investigate two possible scenarios. i) the adversary randomly selects a secret key k and computes c using Bob's public key y_B. It is infeasible for the adversary to compute the signature of c without knowing Alice's private key x_A. ii) the adversary forges a valid pair $\{\sigma, c\}$ following the existential forgery, then it is infeasible for the adversary to decrypt c and obtain the secret key k without knowing Bob's private key x_B.

V. Conclusion

In this paper, we propose a new design for email authentication using cryptographic functions supported by PGP and S/MIME. This new design enables only a specified message receiver to authenticate the message. It also allows the message sender to be able to deny generation of the message. This feature can protect the personal privacy.

References

[1] S. Garfinkel, *PGP: Pretty Good Privacy*. OReilly, 1994.
[2] B. Ramsdell, "Secure/multipurpose Internet mail extensions (S/MIME) version 3.1 message specification, RFC 3851," 2004.
[3] B. H. Kim, J. H. Koo, and D. H. Lee, "Robust e-mail protocols with perfect forward secrecy," *IEEE Commun. Lett.*, vol. 10, no. 6, pp. 510512, 2006.
[4] E. J. Yoon and K. Y. Yoo, "Cryptanalysis of robust e-mail protocols with perfect forward secrecy," *IEEE Commun. Lett.*, vol. 11, no. 5, pp. 372374, 2007.
[5] H. Sun, B. Hsieh, and H. Hwang, "Secure e-mail protocols providing perfect forward secrecy," *IEEE Commun. Lett.*, vol. 9, no. 1, pp. 5860, 2005.
[6] A. W. Dent, "Flaws in an e-mail protocol of Sun, Hsieh, and Hwang," IEEE *Commun. Lett.*, vol. 9, no. 8, pp. 718719, 2005.
[7] D. R. L. Brown, "Deniable authentication with RSA and multicasting," Cryptology ePrint Archive, http://eprint.iacr.org/2005/056.pdf, Feb 2005.
[8] C. Kaufman, R. Perlman, and M. Speciner, *Network Security*. Prentice Hall PTR, 2002.
[9] W. Diffle and M. E. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. 22, pp. 644654, 1976.
[10] D. Chaum, "Private signature and proof systems," U.S. patent 5,493,614, 1996.
[11] R. Steinfeld, L. Bull, H. Wang, and J. Pieprzyk, "Universal designated verifier signatures," in *Proc. Asiacrypt03*, vol. LNCS 2894, pp. 523542, 2003.
[12] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120126, 1978.
[13] T. A. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inform. Theory*, vol. 31, no. 4, pp. 469472, 1985.