Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

A digital signature with multiple subliminal channels and its applications

Dai-Rui Lin *, Chih-I Wang, Zhi-Kai Zhang, D.J. Guan

Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, 804, Taiwan, ROC

ARTICLE INFO

Keywords:
Subliminal channels
Digital signatures
One-time signatures
Quadratic residues
Chinese remainder theorem

ABSTRACT

In this paper, we present two schemes for embedding multiple subliminal messages into one-time signature schemes (OTSSs) proposed by Lamport (1971, 1981) [35,36]. Our schemes have the advantage that the subliminal receivers cannot forge a valid signature since they do not share the signer's secret key. Our schemes can also provide more than one independent subliminal message, and the numbers of subliminal messages and receivers are larger than that of the subliminal messages in previous schemes.

1. Introduction

A subliminal channel is a communication channel that allows a sender to transmit an additional secret message to authorized receivers. To do this, subliminal receivers have to share a subliminal key with the signer to protect the subliminal message. Without additional knowledge, the secret message cannot be detected and discovered by any unauthorized receivers.

In 1983, Simmons first constructed a subliminal channel in a digital signature scheme [1]. Since then, many studies on subliminal channels have been published [2–23]. In the Appendix, we show the history of subliminal channel-related publications from 1983 to 2009.

Harn and Gong proposed two digital signature schemes with two subliminal channels in 1997 [3]. The main feature of their schemes was that the subliminal receivers had to share a part of the signer's secret key as the subliminal key. Therefore, their schemes could be vulnerable to conspiracy attack. That is, if a sufficient number of subliminal receivers conspire against the message signer, they can derive the secret key and forge a valid signature, thereby reducing the security of the digital signature scheme. In 1999, Jan and Tseng proposed two digital signature schemes with subliminal channels on the basis of the discrete logarithm problem [4]. Their first scheme has the same problem of conspiracy attack as Harn and Gong's scheme [3]. The security of their second scheme could also be vulnerable to conspiracy attack. The subliminal receivers can obtain information about the signer's secret key through cooperation of the receivers. Lee and Lin [24] pointed out that Jan and Tseng's schemes could be vulnerable to a dishonest receiver attack, wherein a malicious designated receiver can forge the signature and hide a new subliminal message in the signature. The new subliminal message will be accepted by other receivers. Lee and Lin also showed an improvement for avoiding the above security flaw [24].

One-time signature schemes (OTSSs) are secure, fast, and have many applications [25–27]. They are also useful in on-line, off-line, and forward-secure signatures [28]. Recently, several OTSSs have been proposed [25,29,30,27,31–33], and some stream signature schemes using one-time signatures have been presented [28,34,29,30,26]. Stream signature schemes are used for streamed media authentication and signing. Streamed media, such as streamed radio and video, broadcast or multicast via the Internet. In order to enable a widespread and trusted streamed media dissemination, the user needs

* Corresponding author.
E-mail address: javacpc@gmail.com (D.-R. Lin).

0898-1221/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2010.01.001

assurance that the data stream originated from the purported sender. Therefore, using a one-time signature is a good method for signing the digital streams. The cited researchers pointed out that OTSSs have many applications.

We propose two schemes for embedding multiple subliminal messages based on an important concept: the subliminal secret keys of each of the subliminal receivers are not only independent of each other but also independent from the secret signing key. This can ensure the security of both the signature and the subliminal messages. In our two subliminal channel schemes, the subliminal keys are independent from the signer's secret key, and this avoids vulnerability to conspiracy attacks [3]. Moreover, an attack using a malicious subliminal receiver [24] will not work in either of our schemes. That is, a malicious subliminal receiver cannot forge subliminal messages that will be accepted by other subliminal receivers that belong to the malicious subliminal receiver's channel.

In a digital signature scheme such as RSA, ElGamal and DSA, a hash function is applied before signing to shorten the signature. This limits the size of subliminal messages that can be embedded in these signature schemes. For example, if the computation is in Z_p and the hash of the message is k bits long, the subliminal message in these schemes can be no more than $\log p$ bits. On the other hand, the size of an OTSS is usually large. This feature is useful in embedding subliminal messages because it sufficiently increases the size of the subliminal message that a meaningful long message can be sent. For example, in our second scheme, the length of the subliminal message depends on the amount of 1 bits in the hash of the message. In an average case, the subliminal message can be as large as $\frac{1}{2}k \log p$ bits.

The rest of this paper is organized as follows. In the next section, we briefly review Lamport's OTSSs. The proposed schemes are presented in Section 3. Finally, a security analysis and concluding remarks are given in Sections 4 and 5, respectively. The Appendix shows the history of subliminal channel-related publications from 1983 to 2009.

2. Preliminaries

2.1. Review of Lamport's one-time signature schemes

The first of Lamport's OTSSs was presented in [35]. Suppose that the signer S wants to sign a message MSG which may be quite long. In the signature scheme, the signer signs the hashed value of MSG instead of MSG itself. Let M, whose length is n, be the hash of MSG ($M = H(MSG) \in \{0, 1\}^n$). An extra $\log_2(n + 1)$ bits are appended at the end of the message. The value represented in the appended bits can be the length or the checksum of the message M.

Lamport's scheme is divided into three phases: (1) *key generation*, (2) *signature generation*, and (3) *signature verification*.

(1) *Key generation phase*: The signer S chooses $n + \log n$ positive integers $x_1, x_2, \ldots, x_{n+\log n}$ that are large enough as the secret key SK. Then the public key is $PK = H(H(x_1), H(x_2), \ldots, H(x_{n+\log n})) = H(y_1, \ldots, y_{n+\log n})$, where $y_i = H(x_i)$, $1 \le i \le n + \log n$.
(2) *Signature generation phase*: The signature of M is $(s_1, s_2, \ldots, s_{n+\log n})$. Each s_i is computed as follows: If the i-th bit of M, m_i, is equal to 1, then $s_i = x_i$; otherwise, $s_i = H(x_i)$. The signature of M is $\sigma = (s_1, s_2, \ldots, s_{n+\log n})$.
(3) *Signature verification phase*: To verify the signature, let $y'_i = H(s_i)$ when the i-th bit of M is 1 and $y'_i = s_i$ when the i-th bit of M is 0. The signature is valid if $H(y'_1, y'_2, \ldots, y'_{n+\log n}) = PK = H(y_1, y_2, \ldots, y_{n+\log n})$.

The second of Lamport's OTSSs was presented in [36]. The length of the public key is two times that of the key used in the first of Lamport's OTSSs. As in the previous scheme, the signer signs the hash value of MSG instead of MSG itself. Let M, whose length is n, be the hash value of MSG. Note that there are no extra $\log_2(n+1)$ bits appended to the end of the message in this scheme. We will not describe it in detail here.

2.2. Definitions

In this subsection, we will describe a generic multiple-subliminal-channel scheme by making use of the generic algorithm representation, which will be adopted to construct our two schemes. All components and several definitions are given below.

Definition 1. A generic multiple-subliminal-channel scheme based on Lamport's OTSSs consists of five algorithms: Setup, KeyGen, Embed&Signing, Verifying and Extract, which are described as follows:

- **Setup**: Let l be a security parameter and H be a collision-resistant hash function, where $H : \{0, 1\}^* \rightarrow \{0, 1\}^l$. There are one signer S and a receiver set $R = \{R_1, R_2, \ldots, R_{(n+\log n)/2}\}$ in the proposed scheme. The corresponding subliminal key K of R is $K = \{k_1, k_2, \ldots, k_{(n+\log n)/2}\}$, where $k_i \in \{0, 1\}^l$ is a prime number. Each subliminal receiver in R has to pre-share the corresponding subliminal key in K with S in advance. Assume that each receiver R_j knows j, the position of his/her subliminal message in the signature stream.
- **KeyGen**: This is the key generation algorithm. Given secure parameters l and n, the algorithm will return the candidate secret key $CSK = \{sk_1, sk_2, \ldots, sk_{n+\log n}\}$, where $sk_i \in \{0, 1\}^l$.
- ***Embed&Signing***: This is the subliminal message embedding and signing algorithm. Given CSK, the hashed message $M = \{m_1, m_2, \ldots, m_{n+\log n}\}$ to be signed and the subliminal message $SM = \{sm_1, sm_2, \ldots, sm_{(n+\log n)/2}\}$, the algorithm will return the signature σ and the public key $PK = \{pk_1, pk_2, \ldots, pk_{n+\log n}\}$ of σ, where $pk_i = H(sk_i)$.