Theoretical Computer Science 560 (2014) 172–189

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Timed encryption with application to deniable key exchange [☆]

Shaoquan Jiang [a,b,*]

[a] Institute of Information Security, Mianyang Normal University, Mianyang, 621000, China
[b] School of Computer Science and Engineering, University of Electronic Science and Technology of China, 2006 Xiyuan Rd, High Tech District West, Chengdu, 611731, China

ARTICLE INFO

Available online 15 February 2014

Keywords:
Public-key encryption
Key exchange
Deniability

ABSTRACT

In this paper, we propose a new notion of timed encryption, in which the encryption is secure within time t while it is completely insecure after some time $T > t$. We consider the setting where t and T are both polynomial (in the security parameter). This primitive seems useful in applications where some intermediate data needs to be private temporarily while later it is desired to be public. We propose two schemes for this. One is reasonably efficient in the random oracle model; the other is generic without a random oracle. To demonstrate its usefulness, we use it as a building block to construct a new deniable key exchange (KE) protocol. A deniable KE protocol is a protocol that allows two parties to securely agree on a secret while neither of them can prove to a third party the fact of communication. So an honest party can deny his participation in the communication. Our protocol is adaptively deniable and secret in the concurrent and non-eraser model that admits session state reveal attacks and eavesdropping attacks. Here a session state reveal attack in a non-eraser model means that a user does not erase his intermediate data (e.g., due to a system backup) and, when compromised, will hand it out faithfully to an adversary. An eavesdropping attack allows an adversary to eavesdrop transcripts between honest users, in which he is unaware of the randomness. As emphasized by Di Raimondo et al. [14] and Yao and Zhao [30], an eavesdropping attack is very serious toward breaking the deniability. Our protocol is the first to simultaneously achieve all of the above properties without random oracles. The only price we pay is a timing restriction on the protocol execution. However, this restriction is rather weak and is essentially to require a user to answer an incoming message as soon as possible, which can be satisfied by almost all protocols that are executed online.

1. Introduction

In this paper, we propose a new notion of *timed encryption*. This is a public key encryption primitive, except that the secrecy of the plaintext is only required to hold in a short time t and the encrypted content will be completely insecure after a longer time T. Here t and T are pre-determined at the system setup stage. Any regular public key encryption scheme can be regarded as a timed encryption scheme with an exponentially (in the security parameter) large T. However, we are

[☆] Abstract appeared in the 9th Annual Conference on Theory and Application of Models of Computation (TAMC'12).

[*] Correspondence to: School of Computer Science and Engineering, University of Electronic Science and Technology of China, 2006 Xiyuan Rd, High Tech District West, Chengdu, 611731, China. Tel.: +86 2861830360; fax: +86 2861830360.

E-mail address: shaoquan.jiang@gmail.com.

http://dx.doi.org/10.1016/j.tcs.2014.02.005
0304-3975/© 2014 Elsevier B.V. All rights reserved.

interested in the case where both t and T are polynomial. Practically, such a setup is possible if a timed encryption scheme is used in an interactive protocol and the secrecy is required to hold only during the protocol execution. In this case, t can be set as short as a few seconds and T can be set as a few hours. For a concrete application of this primitive, we consider an auction scheme which consists of two phases: a bidding phase and an opening phase. In the bidding phase, every bidder casts his bid and no one else can read it; in the opening phase, the bidding result is made public and it is desired that the result is publicly verifiable. Assume we are only interested in the fairness of the scheme. Then, a bidder can cast his bid using a timed encryption scheme. Under this, t can be set such that the bid remains private before the opening phase. Later, after time T, one can verify the result by forcefully decrypting all the encrypted bids. For another application, consider a deniable authentication protocol, where Alice wishes to authenticate a message to Bob such that Bob cannot prove to a third party the fact of communication. To do this, Bob can first send a secure key encrypted using Alice's timed encryption scheme. Alice then decrypts this key to generate and send an authentication tag within time t to Bob. Since no one except Alice can reply to Bob's request within time t, the authentication is guaranteed. Further, after time $T > t$, anybody can decrypt the authentication key and create the authentication tag. The deniability is guaranteed.

1.1. Related works

A *timed-release encryption* (TRE), initiated by May [26], captures the intuition of "send a message into the future". There are two types of TREs in the literature; see below.

Time-lock based TRE. In TRE [29], a sender generates an RSA scheme and uses the factorization trapdoor to compute 2^t squarings efficiently in order to encrypt a secret. No one else has this trapdoor and hence has to sequentially repeat 2^t squarings in order to decrypt it, due to which the decryption delay is achieved. This approach was adopted by Mao [25] to build a timed-release of RSA encryption and RSA signature. A time-lock based TRE is different from a timed encryption as the latter has a legal receiver who has a decryption key and can decrypt the ciphertext at any time without any delay.

Trusted server based TRE. In this approach, the decryption needs a secret (we call it *time secret*) from a trusted server who will release it only after a period of time, due to which a decryption delay is achieved. This type of TRE has several variants. Identity-based encryption (IBE) [4] can work as this primitive through a key control, where the release time together with the receiver name serves as an identity. The key for this identity will be released by a server only after a desired length of time. This approach has a drawback that the time secret changes with a different receiver, which is inconvenient for a server. In Di Crescenzo et al. [11], the time secret is released through an interaction between a receiver and a server. In Blake and Chan [3], the server releases the time secret without involving a sender or a receiver. The server's time secret works for all users. Their scheme is scalable, compared with IBE or [11]. This approach was further discussed in [7–9]. Following the model [3], a timed-release of a signature was considered by Dodis and Yum [16]. Paterson and Quaglia [28] considered time-specific encryption (TSE), where the server's release time lies in a specific time interval. Hwang, Yum and Lee [21] extended the model [3] with pre-open capability (TRE-PC), where the sender can publish a release key to allow the receiver to decrypt before the server releases the time secret.

The server-based TREs except TRE-PC do not allow a receiver to decrypt before releasing the time secret. TRE-PC allows a receiver to do this by requiring a sender to publish a release key and hence has controlled security: a receiver can decrypt at any time if a sender provides a release key; otherwise, the ciphertext will be secure within time T and completely insecure after that (when the time secret will be provided by the server). This controlled security is for a legal receiver. This is different from a timed encryption. For the latter, the receiver has a decryption key and can decrypt the ciphertext at any time. So its controlled security is only for an outsider: within time t, no outsider can decrypt while after time T anyone can do this. It should be noted that an outsider in TRE does not have the decryption capability at any time. Decryption delay in a timed encryption scheme is achieved by forcing the decryptor to finish a reasonably large amount of sequential computation (similar to a time-lock based TRE). The decryption delay in TREs is achieved through the server's control on the release of a time secret.

Due to the differences between a timed encryption and TRE, they are favored by different applications (even if both are applicable). Consider a key escrow system for instance, where a user encrypts his key so that the government can get it only one year later. In this case, a server based TRE might be better since it only requires a time server to release a time secret one year later while if we use a timed encryption, the government has to keep the forceful decryption running for a whole year. For a deniable key exchange, we will see that a timed encryption is more suitable. Partially, this is because it is hard to keep an online server as in TRE. In addition, if a timed encryption is adopted, the protocol never requires a user to run a forceful decryption. Indeed, this algorithm only works as a proof of the *existence* of a way that an adversary can decrypt a ciphertext without a key and hence he cannot claim that he does not know the plaintext while the receiver does. For a time-lock based TRE, it is not clear how it can be applied to a deniable key exchange. It is certainly interesting if one can find a way.

A more related work is a timed-commitment by Boneh and Naor [5], where one can commit to a message m and within time t the message m remains confidential while after time T, m can be obtained forcefully. They used it to build a timed signature.

We will apply a timed encryption to build a deniably authenticated key exchange protocol. We consider the deniability advocated by Di Raimondo et al. [14] and Yao and Zhao [30], where the deniability remains valid even if an adversary can eavesdrop some communication records between honest users (note: this threat is captured in [14,30] by giving the