Information Sciences 423 (2018) 313–325

Contents lists available at ScienceDirect

Information Sciences

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

Black-box constructions of signature schemes in the bounded leakage setting

Jianye Huang[a], Qiong Huang[a,b,*]

[a] College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China
[b] Nanjing University of Information Science & Technology, Nanjing, China

ARTICLE INFO

Article history:
Received 16 November 2016
Revised 21 August 2017
Accepted 18 September 2017
Available online 21 September 2017

Keywords:
Identity-based signature
Certificateless signature
Black-box construction
Bounded leakage model
Leakage-resilient cryptography

ABSTRACT

To simplify the certificate management procedures in public key infrastructure, Shamir introduced the concept of *identity-based* cryptography. However, it suffers from the key escrow problem. To solve the problem, Al-Riyami and Paterson introduced the notion of *certificateless* cryptography (CLC). However, if a cryptosystem is not perfectly implemented, adversaries would be able to obtain part of the system's secret state via *side-channel attacks*, and thus break the system. This is not considered in the security model of traditional cryptographic primitives. *Leakage-resilient cryptography* was proposed to prevent adversaries from doing so. There are fruitful works on leakage-resilient encryption schemes, while there are not many on signature schemes in the leakage setting.

In this work, we review the folklore generic constructions of identity-based signature and certificateless signature schemes, and show that if the underlying primitives are leakage-resilient, so are the resulting identity-based signature scheme and certificateless signature scheme. The leakage rate follows the minimum one of the underlying primitives. To demonstrate, we show some instantiations of these generic constructions.

1. Introduction

Digital signature is the analogy of message authentication code (MAC) in the public key setting that ensures the integrity of transmitted messages over public channels. In traditional public key infrastructure (PKI), to bind a user's identity with its public key, a digital certificate issued by a *certificate authority* (CA) is needed. Although PKI ensures the binding of a user's identity and its public key and is well used in practice, it suffers from the complex certificate management problem. Certificate creation, storage, revocation and etc, require high computation and storage costs.

To simplify the certificate management procedure, Shamir [38] introduced the notion of *identity-based cryptography* (IBC), in which a user can use any personal information as its public key, such as its name, telephone number, email address, IP address and etc. A third party, called the Key Generation Center (KGC)[1], is responsible for generating secret keys for all users in the system. IBC reduces the computational and storage effort by simplifying the key distribution, which makes it advantageous over the traditional PKI. On the other hand, however, IBC inherently suffers from the key escrow problem, i.e.

* Corresponding author at: College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China.
E-mail addresses: hjnubys@stu.scau.edu.cn (J. Huang), qhuang@scau.edu.cn (Q. Huang).

[1] KGC is also known as the *Public Key Generator* (PKG) in many other works.

https://doi.org/10.1016/j.ins.2017.09.055
0020-0255/

the KGC knows all users' secret keys, and therefore a trusted KGC is necessary. Moreover, a secure channel is required for key distribution between the KGC and a user.

To get rid of the key escrow problem, Al-Riyami and Paterson introduced the notion of *certificateless cryptography* [2]. In a certificateless signature (CLS) scheme, the KGC is responsible for issuing partial secret keys *psk* to users according to their identities *ID*. Besides, each user chooses a public key *upk* along with some secret information *usk* (*a.k.a.* user secret key). Only if the user knows both *psk* and *usk* can it sign a message. Verification of a signature requires the knowledge of both the signer's identity *ID* and its public key *upk*. Therefore, the KGC in a CLS scheme could not sign a message on behalf of a user, and the key escrow problem does not exist any more.

(*Leakage-Resilient Cryptography*). Traditional cryptography focuses on that the input/output of a cryptosystem does not affect the system's security. However, recent works have shown that if a cryptosystem is not implemented well, an adversary is capable of obtaining (part of) the system secret state (and even the secret key) via *side channel attacks*, which is not captured in the security model. For example, Halderman et al. showed that the *cold boot attacks* can be used to recover encryption keys that are stored in the memory even when the computer loses power [21]. Other side-channel attacks include running-time attack [29], electromagnetic radiation analysis [36], power consumption analysis [15], fault detection [5] and etc. To address side-channel attacks, Dziembowski et al. [17] introduced the notion of *leakage-resilient cryptography*, which aims to constructing cryptographic schemes that remain secure even if the adversary is able to obtain part of the secret state of the scheme. They constructed stream ciphers under the assumption *only computation leaks information* which was introduced in the influential work [34] of Micali and Reyzin.

How to formalize the *leakage* is one of the major research topics of leakage-resilient cryptography. Inspired by [1,34], the community now models the information leakage as a function. That is, the adversary is given access to a leakage oracle $\mathcal{O}_L(\cdot)$, which takes as input a polynomial-time computable *leakage function* $f(\cdot)$ chosen by the adversary and outputs the value of the function on input the system's secret key, e.g. $f(sk)$. It is required that the leakage functions should be efficiently computable since we only consider probabilistic polynomial-time (PPT) adversaries. If we further consider the leakage on the randomness used in the computation, e.g. the random number used in signature generation, we call this stronger notion *fully leakage resilience* (FLR) [28]. To prevent the adversary from winning the security game trivially, we usually restrict the amount of leakage to the adversary. In the *bounded leakage-resilient* (BLR) model, the amount of leakage should satisfy $\sum_{i=1}^{q} \|f_i(X)\| \leq \lambda$ for some bound λ after q leakage queries, in order to ensure certain entropy of the secret state. Otherwise, secure cryptographic schemes are never achievable if the system internal state keeps unchanged over time. If we do not set a bound on the total length of leakage, we should resort to the *continuous leakage-resilient* (CLR) model [33], in which the system's secret state is updated periodically while the public information keeps unchanged. The adversary is allowed to get a limited amount of leakage in each period.

There have been many works on leakage-resilient encryption schemes, but not many on the construction of signature schemes in the leakage setting. In this paper we focus on the black-box construction of leakage-resilient signature schemes in the bounded leakage model.

1.1. Related work

(*Leakage-resilient signature*). Katz and Vaikuntanathan proposed signature schemes which are existentially unforgeable under chosen-message attacks in the bounded leakage model based on standard assumptions [28]. There are also leakage-resilient signature schemes secure in other leakage models, for example, continuous leakage model [33], hard-to-invert leakage model [18], auxiliary input model [48] and etc. Furthermore, *full* leakage model was considered for digital signature schemes [9,48], in which the adversary is capable of obtaining leakage of both the secret key and the randomness used in the signing procedure. Wang et al. [40,41] proposed a leakage-resilient and *strongly* unforgeable signature scheme following the framework of [8], which however, requires to add additional elements to the key pair. Recently, [24,42] further showed that Huang et al.'s strongly unforgeable signature scheme [25] can be extended to the leakage setting, which removes the need of changing the key pair of the underlying signature scheme.

(*Identity-based signature*). Shamir proposed the first identity-based signature (IBS) scheme based on integer factorization problem in [38]. Since then, a lot of IBS schemes have been proposed, e.g. [11,13,14,22,35]. To name a few, a provably secure IBS scheme was proposed by Hess in [22], which is existentially unforgeable under adaptive chosen message and fixed identity attacks. In 2003, Choon et al. [14] proposed an IBS scheme based Gap Diffie–Hellman groups. They formalized the security definition of IBS, called existential unforgeability under adaptive chosen-message and identity attacks. An IBS scheme supporting batch verification was later proposed by Cheon et al. in [13], which is a variant of the scheme given in [14]. Chen et al. [11] proposed an IBS scheme without the need of a trusted KGC, eliminating the inherent key escrow problem. In the leakage setting, Li et al. [31] proposed a traceable IBS scheme and proved its security under the hardness of DDH problem. Wu et al. [43] proposed another leakage-resilient IBS scheme based on Galindo and Vivek's leakage-resilient signature scheme [19] in the continuous leakage model, and proved that their IBS scheme is secure against leakage attacks in the generic group model.

(*Certificateless signature*). Al-Riyami et al. [2] proposed the first CLS scheme in 2003 to solve the key escrow problem of IBS. Later, many different security models and schemes [27,39,45,49] as well as applications [10,12,16,37,46] of CLS have been proposed. However, lots of the existing certificateless schemes are proven secure in the random oracle model. Liu et al. [32] proposed the first CLS scheme without random oracles. However, [26,44] demonstrated that Liu et al.'s CLS scheme