Theoretical Computer Science 660 (2017) 23–56

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Fully leakage-resilient signatures revisited: Graceful degradation, noisy leakage, and construction in the bounded-retrieval model

Antonio Faonio [a], Jesper Buus Nielsen [a], Daniele Venturi [b,*]

[a] *Aarhus University, IT-Parken, Aabogade 34, DK-8200 Aarhus N, Denmark*
[b] *University of Trento, Via Sommarive 9, 38123 Trento, Italy*

ARTICLE INFO

Article history:
Received 14 March 2016
Received in revised form 26 October 2016
Accepted 15 November 2016
Available online 24 November 2016
Communicated by G. Persiano

Keywords:
Signature scheme
Leakage resilience
Noisy leakage
Bounded retrieval model

ABSTRACT

We construct new leakage-resilient signature schemes. Our schemes remain unforgeable against an adversary leaking arbitrary (yet bounded) information on the entire state of the signer (sometimes known as *fully* leakage resilience), including the random coin tosses of the signing algorithm.

The main feature of our constructions is that they offer a graceful degradation of security in situations where standard existential unforgeability is impossible. This property was recently put forward by Nielsen, Venturi, and Zottarel (PKC 2014) [19] to deal with settings in which the secret key is much larger than the size of a signature. One remarkable such case is the so-called Bounded-Retrieval Model (BRM), where one intentionally inflates the size of the secret key while keeping constant the signature size and the computational complexity of the scheme.

Our main constructions have leakage rate $1 - o(1)$, and are proven secure in the standard model. We additionally give a construction in the BRM, relying on a random oracle. All of our schemes are described in terms of generic building blocks, but also admit efficient instantiations under fairly standard number-theoretic assumptions. Finally, we explain how to extend some of our schemes to the setting of noisy leakage, where the only restriction on the leakage functions is that the output does not decrease the min-entropy of the secret key by too much.

1. Introduction

Cryptography relies on secret information and random sources to accomplish its tasks. In order for a given cryptographic primitive to be secure, it is typically required that its secrets and randomness are well-protected, and cannot be influenced by an attacker. In practice, however, it is not always possible to fulfill this requirement, and partial information about the secret state of a cryptosystem can leak to an external adversary, e.g., via so-called side-channel attacks exploiting physical characteristics of a crypto-device, such as power consumption [1], electromagnetic radiation [2], and running times [3].

Recently a lot of effort has been put into constructing cryptographic primitives that come along with some form of leakage resilience, meaning that the scheme should remain secure even in case the adversary obtains some type of leakage

[*] Corresponding author.
E-mail address: daniele.venturi@unitn.it (D. Venturi).

http://dx.doi.org/10.1016/j.tcs.2016.11.016
0304-3975/© 2016 Elsevier B.V. All rights reserved.

on the secrets used within the system. A common way to model leakage attacks, is to empower the adversary with access to a leakage oracle, taking as input (adaptively chosen) functions f_i and returning $f_i(st)$ where *st* is the current secret state of the cryptosystem under attack. Clearly some restriction on the functions f_i has to be put, as otherwise there is no hope for security. By now, a plethora of leakage models (corresponding to different ways how to restrict the functions f_i) have been proposed. We review the ones that are more relevant to our work below (and refer the reader to Section 1.3 for a bigger picture).

- **Bounded leakage:** One natural restriction is to just assume that the total bit-length of the leakage obtained via the functions f_i is smaller than some a priori determined leakage bound ℓ. Usually the leakage bound ℓ is also related to the secret key size, so that a relatively large fraction of the secret key can be leaked. Leakage-resilient schemes in this model include storage schemes [4,5], public-key and identity-based encryption [6–14], signature schemes [15,7,10,11, 16–21], and more (see, e.g., [22–25]).
- **Noisy leakage:** A drawback of the bounded leakage model is that physical leakage rarely obeys to the restriction that the length of the leakage is a priori bounded (e.g., a power trace could be much longer than the secret key). A milder restriction (which weakens the above) is to just assume that the total amount of leakage does not reduce the entropy of the secret key by too much. Leakage-resilient primitives in this model include one-way relations, public-key encryption, and signature schemes [6,11,22].

The focus of this paper is on constructing leakage-resilient signatures in the bounded leakage and noisy leakage model, where one demands that a signature scheme remains unforgeable even against an adversary leaking arbitrary (yet restricted as above) information on the signing key and the overall randomness used within the life-time of the system (this flavor is sometimes known as *fully* leakage resilience).

Graceful degradation. Note that in order for a signature scheme to remain existentially unforgeable in the bounded leakage model, it must necessarily be the case that the length of a signature is larger than the length of the secret key (as otherwise an adversary could simply leak a forgery). A first consequence of this is that signatures are very long, as the goal is to enlarge the secret key to tolerate more and more leakage, which is impractical. A second consequence is that we cannot make any meaningful security statement (w.r.t. security in the bounded leakage model) for schemes where the size of the secret key is much larger than the size of a single signature. One remarkable such case is the setting of the Bounded-Retrieval Model [26–28] (BRM), where one intentionally inflates the size of the secret key while keeping constant the size of a signature and the verification key, as well as the computational complexity of the scheme (w.r.t. signature computation/verification).

A similar concern applies to the noisy leakage model, for those schemes where signatures (statistically) reveal little information about the secret key. In such cases leaking a forgery is, again, a valid leakage query, as a signature does not decrease the uncertainty of the secret key by too much. Still, we would like to not consider a scheme completely insecure if the adversary cannot do better than that.

A first step towards addressing the above issues, has recently been taken by Nielsen, Venturi and Zottarel [20] (for the bounded leakage model) who introduced a “graceful degradation” property, essentially requiring that an adversary should not be able to produce more forgeries than what he could have leaked via leakage queries. More precisely, in order to break unforgeability, an adversary has to produce n forgeries where $n \approx \lambda/(\gamma \cdot s) + 1$ for signatures of size s, a total of λ bits of leakage, and a “slack parameter” $\gamma \in (0, 1]$ measuring how close to optimal security a scheme is. The main advantage is that one can design schemes where the size of the secret key is independent of the signature size, leading to shorter signatures. Moreover, this flavor of leakage resilience still allows for interesting applications (e.g., to leaky identification [20]).

1.1. Our contribution

New definitions. We start by generalizing the above graceful degradation property to the setting of fully-leakage resilience (both in the bounded and noisy leakage model). Our main notion, dubbed fully-leakage one-more unforgeability, is essentially the same as the one of [20], with the twist that leakage functions can be applied to the entire state of the signer (both for noisy and length-bounded leakage).

We also establish a “middle-ground” notion, which models a setting where secure erasures of the state are available. In particular, we imagine a situation in which the random coins sampled by the signer are completely erased after each invocation of the signing algorithm. Note that in this setting the leakage can essentially depend only on the secret key and the random coins used to compute a single signature. While requiring perfect erasure is a strong assumption (see, e.g., [29]) and cannot be applied to some scenarios (e.g., to the case of stateless signers that are not equipped with a private source of randomness), we believe our notion might still make sense for some applications, as it in particular allows to design simpler and more efficient schemes.

New generic constructions. Next, we present new constructions of fully leakage-resilient signature schemes based on generic cryptographic building blocks, improving over previous work in several directions. All of our schemes tolerate leakage on the entire state of the signer, up to a $1 - o(1)$ fraction of the secret key length in the bounded leakage model. They also offer graceful degradation, allowing to have short signatures of size independent of the size of the secret key.