A CDH-Based Strongly Unforgeable Signature Without Collision Resistant Hash Function

Takahiro Matsuda[1], Nuttapong Attrapadung[2], Goichiro Hanaoka[2], Kanta Matsuura[1], and Hideki Imai[2,3]

[1] Institute of Industrial Science, The University of Tokyo, Tokyo, Japan
{tmatsuda,kanta}@iis.u-tokyo.ac.jp
[2] Research Center for Information Security, National Institute of Advanced Industrial Science and Technology, Tokyo, Japan
{n.attrapadung,hanaoka-goichiro,h-imai@}@aist.go.jp
[3] Faculty of Science and Engineering, Chuo University, Tokyo, Japan

Abstract. Unforgeability of digital signatures is closely related to the security of hash functions since hashing messages, such as *hash-and-sign* paradigm, is necessary in order to sign (arbitrarily) long messages. Recent successful collision finding attacks against practical hash functions would indicate that constructing practical collision resistant hash functions is difficult to achieve. Thus, it is worth considering to relax the requirement of collision resistance for hash functions that is used to hash messages in signature schemes. Currently, the most efficient strongly unforgeable signature scheme in the standard model which is based on the CDH assumption (in bilinear groups) is the Boneh-Shen-Waters (BSW) signature proposed in 2006. In their scheme, however, a collision resistant hash function is necessary to prove its security. In this paper, we construct a signature scheme which has the same properties as the BSW scheme but does not rely on collision resistant hash functions. Instead, we use a target collision resistant hash function, which is a strictly weaker primitive than a collision resistant hash function. Our scheme is, in terms of the signature size and the computational cost, as efficient as the BSW scheme.

Keywords: digital signature, strong unforgeability, target collision resistant hash function, standard model.

1 Introduction

Unforgeability of digital signatures is closely related to the security of hash functions. In particular, signature schemes that utilize the *hash-and-sign* paradigm [14], where a message of arbitrary length is hashed to fixed length and then the hashed value is signed, are no more secure if collision resistance of the hash function is broken. In theory, it is possible to construct an arbitrary-input-length collision resistant hash function from some concrete assumptions such as the discrete logarithm assumption. In practice, if length of the messages to be signed is always fixed and short, then a hash function from such theoretical construction would be possible. However, if the length of messages varies arbitrarily or

W. Susilo, J.K. Liu, and Y. Mu. (Eds.): ProvSec 2007, LNCS 4784, pp. 68–84, 2007.

becomes long, hash functions for hashing messages would be, in consideration of computational efficiency, replaced with practical cryptographic hash functions such as MD5 or SHA-1 and assumed that such hash functions have collision resistance. Recent successful results of collision finding attacks against practical cryptographic hash functions (e.g., an attack against SHA-1 by Wang *et al.* [26]) show that it would be no longer easy to construct practical collision resistant hash functions. Thus, if we follow the common practice of using practical hash functions to achieve unforgeability, it is worth considering to relax the requirement of collision resistance for hash functions in signature schemes.

A good substitute for collision resistant hash functions in the hash-and-sign schemes would be *target collision resistant* hash functions (TCRHFs). A TCRHF, firstly introduced by Naor and Yung [21] as a *universal one-way hash function* and later renamed by Bellare and Rogaway [4], is a class of *keyed* hash functions. The important fact is that a TCRHF is proven to be a strictly weaker primitive (hence, easier to construct) than a collision resistant hash function by Simon [23].

Even when we allow arbitrary-length messages in signature schemes, if we use TCRHFs, we also have a general and theoretically secure construction of signature schemes [21]. Its signing process is: choose a random hash-key, hash a message with the key, and sign the concatenation of the hash-key and the hashed value. The signature output from the general construction consists of a signature output from the underlying signing algorithm and the hash-key (we call this general construction the *TCRHF-based hash-and-sign construction*). A obvious but crucial problem of this general construction, however, is that the signature size increases by the hash-key size. Solutions for several concrete schemes were proposed by Mironov [19]. In [19], Mironov constructed modified versions of DSA, PSS-RSS and Cramer-Shoup [13] signatures. Unforgeability of the modified DSA and PSS-RSA are proven in the random oracle model, and the modified Cramer-Shoup signature is proven in the standard model. The sizes of signatures obtained from the modified schemes are the same as the original ones for DSA and PSS-RSA, and shorter by a hash-key than the original one for the Cramer-Shoup scheme. The main idea to obtain the modified schemes is to *reuse* randomness generated in a signing algorithm as a hash-key of a TCRHF. In other words, the randomness plays a double role: used to create a signature itself and also used as a hash-key of the TCRHF that hashes the message to be signed. However, this approach is not generically applicable but scheme-dependent, which means that the ways of reusing randomness and the ways of proving security of the schemes are different from each other.

Such efforts of improving the efficiency of TCRHF-based signature schemes could play an important role when we observe another application of digital signatures. Since we need this observation to show the motivation of our work, we review existing works related to that application, in the following. Typically, "unforgeability" for signature schemes means existential unforgeability against adaptive chosen message attacks [15]. And if we introduce a stronger definition called *strong unforgeability* [1], we can have interesting applications of digital signature schemes: constructions of other cryptographic schemes such as