Theoretical Computer Science 412 (2011) 5498–5512

Contents lists available at ScienceDirect

Theoretical Computer Science

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

Identity-based trapdoor mercurial commitments and applications

Xiaofeng Chen [a], Willy Susilo [b,*], Fangguo Zhang [c], Haibo Tian [c], Jin Li [d]

[a] Key Laboratory of Computer Networks and Information Security, Ministry of Education, Xidian University, Xi'an 710071, PR China

[b] Centre for Computer and Information Security Research, School of Computer Science and Software Engineering, University of Wollongong, New South Wales 2522, Australia

[c] School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510275, PR China

[d] School of Computer Science and Educational Software, Guangzhou University, Guangzhou 510006, PR China

ARTICLE INFO

Article history:
Received 9 April 2010
Received in revised form 29 March 2011
Accepted 21 May 2011
Communicated by X. Deng

Keywords:
Mercurial commitments
Trapdoor commitments
Zero-knowledge sets
Identity-based zero-knowledge

ABSTRACT

In this paper, we first introduce the notion of identity-based trapdoor mercurial commitment which enjoys the advantages of both the identity-based trapdoor commitment and trapdoor mercurial commitment, while using the idea of "Customized Identity". Inherently, an identity-based trapdoor mercurial commitment is an underlying building block for constructing identity-based (non-interactive) zero-knowledge sets. That is, a prover can commit to a set S in a way that reveals nothing about S and prove to a verifier, in zero-knowledge, statements of the form $x \in S$ and $x \notin S$. Besides, although the (non-interactive) proof is publicly verifiable, it is also bound to the identity of the prover in a way which is recognizable to any verifier.

Contents

* Corresponding author. Tel.: +61 242215535.
E-mail address: wsusilo@uow.edu.au (W. Susilo).

0304-3975/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2011.05.031

1. Introduction

The notion of a commitment is a fundamental primitive and plays an important role in almost all cryptographic protocols such as auction, voting, identification, zero-knowledge proof. Intuitively, a commitment scheme can be viewed as the digital equivalent of a sealed envelope. The sender places a message in the sealed envelope and gives it to the receiver. On one hand, no one except the sender could open the envelope to learn the message from the commitment (this is called *hiding*). On the other hand, the sender could not change the message any more (this is called *binding*). However, in many applications one needs commitment schemes with additional properties besides hiding and binding, such as trapdoor commitments, non-malleable commitments and mercurial commitments.

Trapdoor commitments (also called chameleon commitments) [3] are a commitment with the so-called "equivocation" property. Roughly speaking, a trapdoor commitment scheme allows anyone with the knowledge of trapdoor to open the commitment in any desired ways (and thus "equivocate"). Naturally, without the trapdoor, equivocation would remain computationally infeasible.

In Eurocrypt 2005, Chase et al. [11] introduced a variant of commitments called mercurial commitments. Compared with the traditional commitments, the opening of mercurial commitments is two-tiered. In the soft opening (also called *teasing*), it is possible for the sender to come up with a commitment that can be teased to any value of the sender's choice.[1] On the other hand, it is computationally binding in the hard opening (also called *true opening*).

Mercurial commitments are somewhat different from trapdoor commitments. Note that trapdoor commitments are equivocal whenever the sender knows the trapdoor information. However, in mercurial commitments the sender must decide whether to make the commitment equivocal or binding before forming the commitment. In other words, the sender must beforehand choose whether to "soft commit" so as to be able to tease to any value but not open at all, or to "hard commit" so as to be able to tease and to open to only one particular value.

The notion introduced in [11] is actually a trapdoor mercurial commitment which satisfies a strong equivocation (namely, simulatability) property. However, a mercurial commitment scheme without such equivocation property also has some applications. Therefore, Catalano et al. [12] gave a noticeably simpler definition for a plain mercurial commitment and a trapdoor mercurial commitment, respectively. Besides, Catalano et al. [13] introduced the notion of trapdoor q-mercurial commitments, which allows the sender to commit to a sequence of exactly q messages $(m_1, \ldots, m_q)$, rather than to a single one, as with standard mercurial commitments. Moreover, an interesting problem left is whether there is an efficient construction for a trapdoor q-mercurial commitment that allows for openings whose length is independent of q. Very recently, Libert and Yung [24] introduced a new efficient instantiation of q-mercurial commitments to solve this problem.

A trapdoor mercurial commitment scheme is an important building block for constructing zero-knowledge sets (ZKS). ZKS, firstly introduced by Micali et al. [25], allow a prover to commit to an arbitrary finite set S in such a way that for any string x he can non-interactively provide an efficient sound proof of whether $x \in S$ or $x \notin S$, without revealing any other knowledge about S (not even for its size). All of the constructions [11,13,25] for ZKS used the Merkle-tree-like based approach. Informally, to generate a commitment *com* to the database D, the prover views each $x \in D$ as an integer numbering a leaf of a height-l binary tree, and places a commitment to the information $v = D(x)$ into leaf number x. Each internal node of the tree is generated to contain the commitment to the contents of its two children. Then, *com* is the value of the root in the tree.

Related work. Ostrovsky et al. [28] provided constructions for consistent database queries, which allow the prover to commit to a database, and then provide query answers that are provably consistent with the commitment. Their constructions can handle queries much more general than just membership queries. Recently, Prabhakaran and Xue [29] introduced a related notion of statistically hiding sets that requires the hiding property of zero-knowledge sets to be preserved against unbounded verifiers.

Non-interactive zero-knowledge (NIZK) proof systems, introduced by Blum et al. [5], play a significant role in the theory of cryptography. NIZK proofs satisfy the property of transferability. That is, if the prover P gives a NIZK proof to the verifier V, the proof is still convincing when V gives it to the third party V'. Such a feature has some advantages if one would like to disseminate proofs as widely as possible. On the other hand, NIZK proofs could not offer any guarantees against plagiarism since there is no evidence for V' to recognize the original prover who actually composed the proof. Only when the dispute occurs, the original prover provides an evidence (e.g. an interactive proof) to convince the judge. Jakobsson et al. [22] firstly introduced the notion of designated verifier knowledge proof which has the property of non-transferability. That is, only the designated verifier can verify the proof and cannot convince any third party. Therefore, the problem of plagiarism can be easily solved. However, it limits the widespread dissemination of the proof. A seemingly trivial idea to solve the conflicts between dissemination and plagiarism of NIZK proofs is that, the prover could sign on the proof with his signing key. However, it must rely on the setting of public-key infrastructure. Obviously, the cost of NIZK proofs is increased due to the key management problem in the public-key infrastructure. Besides, such a proof could not be constructed in the common reference string model. On the other hand, when a dishonest verifier received the proof σ and the corresponding signature $Sig(sk_P, \sigma)$ of the prover, he could generate his signature $Sig(sk_V, \sigma)$ and then convince others. Thus, this idea cannot provide a full solution to this problem.

[1] A tease of a commitment to a value m is actually a guarantee that the commitment cannot be opened to any value other than m.