Theoretical Computer Science 562 (2015) 112–121

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Efficient algorithms for secure outsourcing of bilinear pairings

Xiaofeng Chen [a,*], Willy Susilo [b], Jin Li [c], Duncan S. Wong [d], Jianfeng Ma [a], Shaohua Tang [e], Qiang Tang [f]

[a] *State Key Laboratory of Integrated Service Networks (ISN), Xidian University, Xi'an, PR China*
[b] *Centre for Computer and Information Security Research (CCISR), School of Computer Science and Software Engineering, University of Wollongong, Australia*
[c] *School of Computer Science and Educational Software, Guangzhou University, Guangzhou, PR China*
[d] *Department of Computer Science, City University of Hong Kong, Hong Kong, PR China*
[e] *School of Computer Science and Engineering, South China University of Technology, Guangzhou, PR China*
[f] *APSIA group, SnT, University of Luxembourg, 6, rue Richard Coudenhove-Kalergi, L-1359 Luxembourg, Luxembourg*

ARTICLE INFO

Article history:
Received 14 March 2013
Received in revised form 18 April 2014
Accepted 18 September 2014
Available online 28 September 2014
Communicated by G. Ausiello

Keywords:
Cloud computing
Outsource-secure algorithms
Bilinear pairings
Untrusted program model

ABSTRACT

The computation of bilinear pairings has been considered the most expensive operation in pairing-based cryptographic protocols. In this paper, we first propose an efficient and secure outsourcing algorithm for bilinear pairings in the two untrusted program model. Compared with the state-of-the-art algorithm, a distinguishing property of our proposed algorithm is that the (resource-constrained) outsourcer is not required to perform any expensive operations, such as point multiplications or exponentiations. Furthermore, we utilize this algorithm as a subroutine to achieve outsource-secure identity-based encryptions and signatures.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

With the rapid development in availability of cloud services, the techniques for securely outsourcing the prohibitively expensive computations to untrusted servers are getting more and more attention in the scientific community. In the outsourcing computation paradigm, the resource-constrained devices can enjoy the unlimited computation resources in a pay-per-use manner, which avoids large capital outlays in hardware/software deployment and maintenance.

Despite the tremendous benefits, outsourcing computation also inevitably introduces some new security concerns and challenges. Firstly, the computation tasks often contain some sensitive information that should not be exposed to the untrusted cloud servers. Therefore, the first security challenge is the *secrecy* of the outsourcing computation: the cloud servers should not learn anything about the data (including the *secret* inputs and the outputs). We argue that the encryption can only provide a partial solution to this problem since it is very difficult to perform meaningful computations over the encrypted data. Note that fully homomorphic encryption could be a potential solution, but the existing schemes are impractical. Secondly, the semi-trusted cloud servers may return an invalid result. For example, the servers might contain a software bug that will fail on a constant number of invocations. Moreover, the servers might decrease the amount of the computation due to financial incentives and then return a computationally indistinguishable (invalid) result. Therefore, the second security challenge is the *checkability* of the outsourcing computation: the outsourcer should have the ability to detect

[*] Corresponding author.
E-mail address: xfchen@xidian.edu.cn (X. Chen).

http://dx.doi.org/10.1016/j.tcs.2014.09.038
0304-3975/© 2014 Elsevier B.V. All rights reserved.

any failures if the cloud servers misbehave. Trivially, the test procedure should never need to perform other complicated computations since the computationally limited devices such as RFID tags or smartcard may be incapable to accomplish the test. At the very least, it must be *far more* efficient than accomplishing the computation task itself (recall the motivation for outsourcing computations).

In the last decade, the bilinear pairings, especially the Weil pairing and Tate pairing of algebraic curves, have initiated some completely new fields in cryptography, making it possible to realize cryptographic primitives that were previously unknown or impractical [14,15,34]. Trivially, implementing the pairing-based cryptographic protocols is dependent on the fast computation of pairings, and thus plenty of research work has been done to implement this workload efficiently [8,5,15,32,37,42].

The computation of bilinear pairings has been considered the prohibitive expensive operation in embedded devices such as the RFID tag or smartcard (note that we even assume that the modular exponentiation is too expensive to be carried out on such devices). Chevallier-Mames et al. [23] presented the first algorithm for secure delegation of elliptic-curve pairings based on an untrusted server model. Besides, the outsourcer could detect any failures with probability 1 if the server misbehaves. However, an obvious disadvantage of the algorithm is that the outsourcer should carry out some other expensive operations such as point multiplications and exponentiations. More precisely, on the one hand, we argue that these expensive operations might be too resource consuming to be carried out on a computationally limited device. On the other hand, the computation of point multiplications is even comparable to that of bilinear pairings in some scenarios [25,42].[1] Therefore, it is meaningless if the client must perform point multiplications in order to outsource pairings since this contradicts with the aim of outsourcing computation. Therefore, the algorithm is meaningless for real-world applications in this sense. To the best of our knowledge, it seems that all of the following works on delegation of bilinear pairings [24,35,44] also suffer from the same problems.

Our contribution. In this paper, we propose the first efficient and secure outsourcing algorithm of bilinear pairings in the one-malicious version of two untrusted program model [33]. Compared with the state-of-the-art algorithm in [23], a distinguishing property of our proposed algorithm is that the (resource-constrained) outsourcer never needs to perform any expensive operations such as point multiplications and exponentiations. Hence, our proposed algorithm is very practical. Furthermore, we also utilize this algorithm as a subroutine to achieve outsource-secure Boneh–Franklin identity-based encryptions and Cha–Cheon identity-based signatures.

1.1. Related work

Abadi et al. [1] proved the impossibility of secure outsourcing an exponential computation while locally doing only polynomial time work. Therefore, it is meaningful only to consider outsourcing expensive polynomial time computations. The theoretical computer science community has devoted considerable attention to the problem of how to securely outsource different kinds of expensive computations. Atallah et al. [4] presented a framework for secure outsourcing of scientific computations such as matrix multiplications and quadrature. However, the solution used the disguise technique and thus allowed leakage of private information. Atallah and Li [3] investigated the problem of computing the edit distance between two sequences and presented an efficient protocol to securely outsource sequence comparisons to two servers. Recently, Blanton et al. proposed a more efficient scheme for secure outsourcing sequence comparisons [12]. Blanton and Aliasgari [10,11] proposed an efficient scheme for secure outsourcing DNA computations and biometric comparisons. Benjamin and Atallah [7] addressed the problem of secure outsourcing for widely applicable linear algebra computations. However, the proposed protocols required the expensive operations of homomorphic encryptions. Atallah and Frikken [2] further studied this problem and gave improved protocols based on the so-called weak secret hiding assumption. Recently, Wang et al. [45] presented efficient mechanisms for secure outsourcing of linear programming computations.

The problem of securely outsourcing expensive computations has been well studied in the cryptography community. In 1992, Chaum and Pedersen [21] firstly introduced the notion of wallets with observers, a piece of secure hardware installed on the client's computer to perform some expensive computations. Hohenberger and Lysyanskaya [33] proposed the first outsource-secure algorithm for modular exponentiations based on the two previous approaches of precomputation [16,41] and server-aided computation [27,38]. Very recently, Chen et al. [22] proposed more efficient outsource-secure algorithms for (simultaneously) modular exponentiation in the two untrusted program model.

Since the servers (or workers) are not trusted by the outsourcers, Golle and Mironov [30] first introduced the concept of ringers to solve the trust problem of verifying computation completion. The following works focused on the other trust problem of retrieving payments [9,18,19,43]. Besides, Gennaro et al. [26] first formalized the notion of verifiable computation to solve the problem of verifiably outsourcing the computation of an arbitrary functions, which has attracted the attention of plenty of researchers [13,29,28,36,39]. Gennaro et al. [26] also proposed a protocol that allowed the outsourcer to efficiently verify the outputs of the computations with a computationally sound, *non-interactive* proof (instead of interactive ones). Benabbas et al. [6] presented the first practical verifiable computation scheme for high degree polynomial functions. In 2011,

[1] As pointed out in [25,42], when the supersingular elliptic curve is defined over a 512-bit finite field with embedding degree 2, the computational overhead of a point multiplication is almost the same as that of a standard Tate pairing.

Download English Version:

https://daneshyari.com/en/article/436131

Download Persian Version:

https://daneshyari.com/article/436131

Daneshyari.com