SCIENCE CHINA
Information Sciences

• LETTER •

March 2018, Vol. 61 039104:1–039104:3
doi: 10.1007/s11432-017-9161-2

Efficient and secure outsourcing of bilinear pairings with single server

Min DONG & Yanli REN*

School of Communication and Information Engineering, Shanghai University, Shanghai 200444*, China*

Received 13 April 2017/Accepted 21 June 2017/Published online 13 September 2017

Citation Dong M, Ren Y L. Efficient and secure outsourcing of bilinear pairings with single server. Sci China Inf Sci, 2018, 61(3): 039104, doi: 10.1007/s11432-017-9161-2

Dear editor,

Cloud computing is playing important role at the era of big data, and its availability makes outsourcing computations possible [1]. By outsourcing computations, mobile devices with weak computational ability can outsource complex operations to powerful cloud servers so that the time cost can be greatly reduced [2]. Despite of unique advantages, there are also some security challenges we need pay attention to.

• Secrecy. The data that resource-limited client outsources may contain some sensitive information, so it should be encrypted before outsourcing.

• Verifiability. The cloud servers are not fully trusted and they may work maliciously, and therefore the outsourcer should have the ability of verifying the results returned from the servers.

• Efficiency. When we ensure the secrecy and verifiability of outsourcing computation, we should also make sure that the process of encrypting and verifying is not involved in any expensive computation so that the outsourcing scheme is practical.

In the cryptographic community, there were many researchers tried to outsource expensive computation to untrusted cloud server [3–5]. Moreover, plenty of researches have been done to enhance the efficiency of carrying out bilinear pairing. Chevallier et al. [6] first proposed an algorithm for outsourcing bilinear pair with one untrusted server, but the outsourcer was involved in some other complex computations in their algorithm, i.e., this algorithm was not practical. Different from [6], other researchers tried to achieve efficient outsourcing algorithm based on two servers. Chen et al. [7] first presented a practical algorithm for outsourcing bilinear pairing with two servers. One the client could detect the errors with a probability of 1/2. Then Tian et al. [8] presented two outsourcing algorithms for pairing based on two servers. One of them improved the efficiency and kept a checkability of 1/2, and the other one obtained a high verifiability at the expense of efficiency. Recently, Ren et al. [9] proposed a fully verifiable algorithm for outsourcing bilinear pairing with two servers, but the outsourcer needed to communicate with servers for two times to carry out single bilinear pairing.

Our contributions. In this letter, we propose two new outsourcing algorithms for bilinear pairings based on single untrusted server. The main contributions can be shown as follows.

• Different from previous algorithms, the proposed algorithms are both based on single untrusted server, which is more practical in real cloud environment.

• We propose an efficient outsourcing algorithm for t-simultaneous bilinear pairings.

• The proposed algorithms are efficient and their checkability are both close to 1. Moreover, we keep both inputs and outputs private.

* Corresponding author (email: renyanli@shu.edu.cn)
The authors declare that they have no conflict of interest.

• The outsourcer can set different values of some parameters according to security requirement, efficiency and verifiability.

The proposed BPS algorithm. As shown in [3], we also need a subroutine called Rand to speed the precomputation of the outsourcer. When it is invoked, the client will get a random vector in the form of follows: $(a_1P, a_2P, \ldots, a_{i+3}P, b_1Q, b_2Q, \ldots, b_{j+3}Q, \rho = (\rho_2, \ldots, \rho_i) \in \{+1, -1\}^{i-1}, e(a_1P, b_1Q), \sigma = (\sigma_2, \ldots, \sigma_j) \in \{+1, -1\}^{j-1}, t_u, t'_u)$, where $t_u, t'_u \in \{1, \ldots, s\}$ and $i = j$, s is a small integer and $u \in \{1, \ldots, 7\}$.

Moreover, the relationship of the elements in Rand can be described as follows:
$b_{j+1}Q + t_1 \sum_{E_k \in B_1} \sigma_k E_k + t_2 \sum_{E_k \in B_{12}} \sigma_k E_k + t_3 \sum_{E_k \in B_{13}} \sigma_k E_k = -b_1 Q$,
$b_{j+2}Q + t_4 \sum_{E_k \in B_2} \sigma_k E_k + t_5 \sum_{E_k \in B_{12}} \sigma_k E_k + t_6 \sum_{E_k \in B_{23}} \sigma_k E_k = -b_1 Q$,
$b_{j+3}Q + t_7 \sum_{E_k \in B_3} \sigma_k E_k + t_3 \sum_{E_k \in B_{13}} \sigma_k E_k + t_6 \sum_{E_k \in B_{23}} \sigma_k E_k = b_1 Q$,
$a_{i+1}P + t'_1 \sum_{E_l \in A_1} \rho_l E_l + t'_2 \sum_{E_l \in A_{12}} \rho_l E_l + t'_3 \sum_{E_l \in A_{13}} \rho_l E_l = -a_1 P$,
$a_{i+2}P + t'_4 \sum_{E_l \in A_2} \rho_l E_l + t'_5 \sum_{E_l \in A_{12}} \rho_l E_l + t'_6 \sum_{E_l \in A_{23}} \rho_l E_l = -a_1 P$,
$a_{i+3}P + t'_7 \sum_{E_l \in A_3} \rho_l E_l + t'_3 \sum_{E_l \in A_{13}} \rho_l E_l + t'_6 \sum_{E_l \in A_{23}} \rho_l E_l = a_1 P$.

In above equations, $k \in \{2, \ldots, j\}, l \in \{2, \ldots, i\}$, E_k and E_l denote b_kQ and a_lP, respectively. Meanwhile, E_k and E_l are distributed randomly in different subsets of set B or A where $B = B_1 \cup B_2 \cup B_3$ and $B_{12} = B_1 \cap B_2, B_{23} = B_2 \cap B_3, B_{13} = B_1 \cap B_3$, $A = A_1 \cup A_2 \cup A_3$ and $A_{12} = A_1 \cap A_2, A_{23} = A_2 \cap A_3, A_{13} = A_1 \cap A_3$.

Let q be a large prime, the inputs of algorithm are two random points $A \in G_1, B \in G_2$, and the output is $e(A, B)$. Moreover, the inputs A, B and the output $e(A, B)$ should be private to U. In order to outsource single bilinear pairing, the proposed algorithm includes the following four steps.

(1) T firstly invokes Rand once to get a random vector:

$$e(a_1P, b_1Q), t_u, t'_u,$$
$$\rho = (\rho_2, \ldots, \rho_i), a_1P, a_2P, \ldots, a_{i+3}P,$$
$$\sigma = (\sigma_2, \ldots, \sigma_j), b_1Q, b_2Q, \ldots, b_{j+3}Q.$$

(2) Then T queries server U in random order:

$$e(A + a_1P, B + b_{j+3}Q) = \theta_1,$$
$$e(A + a_1P, b_mQ) = \alpha_m, m = \{2, \ldots, j+2\},$$
$$e(A + a_{i+3}P, B + b_1Q) = \theta_2,$$
$$e(a_nP, B + b_1Q) = \beta_n, n = \{2, \ldots, i+2\}.$$

(3) After receiving the results returned from the server U, T checks whether the equations $\{Q_1, Q_2, Q_3\}$ hold based on the relationship described in Rand. Details about the equations $\{Q_1, Q_2, Q_3\}$ are given in Appendix A.

(4) If all of the above equations hold, the outsourcer T obtains the final results:

$$Q_1 = e(A + a_1P, -b_1Q), \quad (1)$$
$$Q_2 = e(A + a_1P, B + b_1Q), \quad (2)$$
$$Q_3 = e(-a_1P, B + b_1Q). \quad (3)$$

Then the outsourcer T computes: $e(A, B) = Q_1Q_2Q_3 \cdot e(a_1P, b_1Q)$.

Comparison. In Table 1, we compare the first proposed algorithm BPS with the previous ones, where PA, SM separately denote point addition and scalar multiplication in G_1 or G_2, MM denotes modular multiplication in G_T, and "Pair" denotes bilinear pairing. In BJN algorithm [6], the client needs to execute other expensive operations including ten modular exponentiations and six scalar multiplications, so it is not practical and we do not compare it with the proposed algorithm BPS in Table 1.

According to the performance analysis (given in Appendix B), if we set $s = 2, i = j = 10$, the probability of exposing of the sensitive information is about 10^{-22}, which is negligible.

Assume that we set $s = 2, i = j = 10$, there are approximately 46 MMs in BPS and the checkability is about 0.969. Compared with algorithm in Pair [7] and TZR1 [8], the proposed BPS algorithm improves checkability to almost 1 though a little computation cost is appended. In algorithms TZR2 [8], VBP [9] and BPS, the outsourcer can check the error with a high probability. Moreover, the proposed algorithm BPS is more efficient than TZR2 [8], and its efficiency is similar to VBP [9].

Note that the most important difference among those algorithms is that previous outsourcing algorithms for bilinear pairing are all based on two servers, but the proposed BPS algorithm outsources bilinear pairing based on single server. It is well known that it is difficult to find two non-colluding servers in the real cloud environment, so the improvement of proposed one is obvious.

The proposed NBPS algorithm. We extend the outsourcing algorithm for single bilinear pairing to outsource t-simultaneous bilinear pairings with an untrusted server. The inputs are $\{A_1, \ldots, A_t\} \in G_1, \{B_1, \ldots B_t\} \in G_2$, and the output is $\prod_{y=1}^{t} e(A_y, B_y)$.

Similar to Rand, we use Rand$'$ to achieve the second proposed algorithm NBPS. The relationship of the elements in Rand$'$ is same to that in Rand.

Table 1 Comparison among different algorithms

Algorithm	PA(T)	MM(T)	Pair(U)	Servers	Checkability	Communication-rounds
Pair [7]	5	4	8	2	1/2	2
TZR1 [8]	4	3	6	2	1/2	2
TZR2 [8]	12	10	6	2	0.918	2
VBP [9]	8	14	6	2	1	4
BPS	4	46	24	1	0.969	2

(1) T firstly invokes Rand$'$ once to obtain a random vector:

$$e(a_1P, tb_1Q), t_u, t'_u,$$
$$\rho = (\rho_2, \ldots, \rho_i), a_1P, a_2P, \ldots, a_{i+3}P,$$
$$\sigma = (\sigma_2, \ldots, \sigma_j), b_1Q, b_2Q, \cdots, b_{j+3}Q.$$

(2) Then T invokes the server U in random order:

$$e(A_y + a_1P, B_y + b_{j+3}Q) = \theta_{y1},$$
$$e(A_y + a_1P, b_mQ) = \alpha_{ym},$$
$$e(A_y + a_{i+3}P, B_y + b_1Q) = \theta_{y2},$$
$$e(a_nP, B_y + b_1Q) = \beta_{yn},$$
$$y = \{1, \ldots, t\},$$
$$m = \{2, \ldots, j+2\}, n = \{2, \ldots, i+2\}.$$

(3) After receiving the results returned from the server U, T checks whether the equations $\{Q_4, Q_5, Q_6\}$ hold based on the relationship described in Rand. Details about the equations $\{Q_4, Q_5, Q_6\}$ are given in Appendix C.

(4) If all of the above equations hold, then the client T can get the final results:

$$Q_4 = \prod_{y=1}^{t} e(A_y + a_1P, -b_1Q),$$
$$Q_5 = \prod_{y=1}^{t} e(A_y + a_1P, B_y + b_1Q),$$
$$Q_6 = \prod_{y=1}^{t} e(-a_1P, B_y + b_1Q).$$

Finally, T computes: $\prod_{y=1}^{t} e(A_y, B_y) = Q_4Q_5Q_6 \cdot e(a_1P, tb_1Q)$.

Conclusion. In this letter, we propose two efficient outsourcing algorithms for single bilinear pairing and t-simultaneous bilinear pairings based on a cloud server. In the proposed algorithms, the client can detect the errors those the server may make with a probability of almost 1. Moreover, all of the inputs and outputs are private for the server. In the future, we will do some work to reduce the computational cost of the server or improve the efficiency of outsourcing bilinear pairings based on single untrusted cloud server with high checkability.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 61572309).

Supporting information Appendixes A–C. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Subashini S, Kavitha V. A survey on security issues in service delivery models of cloud computing. J Netw Comput Appl, 2011, 34: 1–11
2 Gennaro R, Gentry C, Parno B. Non-interactive verifiable computing: outsourcing computation to untrusted workers. In: Proceedings of the 30th Annual International Cryptology Conference, Santa Barbara, 2010. 465–482
3 Hohenberger S, Lysyanskaya A. How to securely outsource cryptographic computations. In: Proceedings of the 2nd Theory of Cryptography Conference, Cambridge, 2005. 264–282
4 Chen X F, Li J, Ma J F, et al. New algorithms for secure outsourcing of modular exponentiations. IEEE Trans Paral Distr Syst, 2014, 25: 2386–2396
5 Li J, Li J W, Chen X F, et al. Identity-based encryption with outsourced revocation in cloud computing. IEEE Trans Comput, 2015, 64: 425–437
6 Chevallier B, Coron J, McCullagh N, et al. Secure delegation of elliptic-curve pairing. In: Proceedings of the 9th IFIP WG 8.8/11.2 International Conference on Smart Card Research and Advanced Application, Passau, 2010. 24–35
7 Chen X F, Susilo W, Li J, et al. Efficient algorithms for secure outsourcing of bilinear pairings. Theor Comput Sci, 2015, 562: 112–121
8 Tian H B, Zhang F G, Ren K. Secure bilinear pairing outsourcing made more efficient and flexible. In: Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, Security, Singapore, 2015. 417–426
9 Ren Y L, Ding N, Wang T Y, et al. New algorithms for verifiable outsourcing of bilinear pairings. Sci China Inf Sci, 2016, 59: 099103