Future Generation Computer Systems 62 (2016) 196–204

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Batch Public Key Cryptosystem with batch multi-exponentiation

Qianhong Wu [a,e,g], Yang Sun [a], Bo Qin [b], Jiankun Hu [c], Weiran Liu [a], Jianwei Liu [a], Yong Ding [d,f,*]

[a] *School of Electronics and Information Engineering, Beihang University, China*

[b] *Key Laboratory of Data Engineering and Knowledge Engineering, Ministry of Education, School of Information, Renmin University of China, China*

[c] *School of Engineering and IT, University of New South Wales, Australia*

[d] *School of Mathematics and Computing Science, Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi, China*

[e] *State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, China*

[f] *Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan, China*

[g] *State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100093, China*

HIGHLIGHTS

- We propose a Basic Batch Multi-exponentiation algorithm and a Simultaneous Batch Multi-exponentiation algorithm.
- We propose batch public-key encryption and decryption.
- We conduct extensive experiments on the proposal.

ARTICLE INFO

Article history:
Received 31 May 2015
Received in revised form
9 November 2015
Accepted 12 December 2015
Available online 19 December 2015

Keywords:
Modular exponentiation
Batch multi-exponentiation
Batch encryption
Batch decryption
Cramer–Shoup cryptosystem
Cloud security

ABSTRACT

A Public Key Cryptosystem (PKC) is a fundamental tool to protect data security. Most PKC schemes involve complicated operations, e.g., modular exponentiations, which are expensive for cloud environment where enormous data are collected from capability-limited devices, e.g., wireless sensors, mobile phones and tablets. To address this problem, this paper investigate how to reduce the laborious computations of a large number of exponentiations in public key encryption and decryption systems. Firstly, we propose algorithms to speed up *batch multi-exponentiation* in different configurations. Our algorithms improve the existing multi-exponentiation and batch single-base exponentiations by allowing a large number of multi-base exponentiations to be processed in batch. Secondly, we build a batch PKC scheme from the famous Cramer–Shoup cryptosystem by allowing batch encryption and batch decryption. For batch encryption, we exploit our proposed batch multi-exponentiation approach so that multiple messages can be encrypted in batch to reduce the computation overhead; and for batch decryption, we further incorporate techniques derived from batch signature verification so that the received ciphertexts can be decrypted in batch. We conduct thorough theoretical and experimental performance analysis of the proposed batch cryptosystem. The analyses show that the batch multi-exponentiation algorithms greatly accelerate calculation speed of the Cramer–Shoup system, compared with the naive implementations with existing multi-exponentiation approaches, by more than 40% in encryption and 80% in decryption. We also provide optimal batch size configurations in the case that some ciphertexts are erroneous. This work will help make PKC towards practical applications in the cloud environment.

1. Introduction

A Public Key Cryptosystem (PKC), due to its freeness of secret key exchange, is a popular tool to protect data security. A typical operational PKC paradigm is to use the receiver's public key to encrypt a session key and then use the session key to encrypt the digital contents. For purpose of fine-grained access control, it

[*] Corresponding author at: School of Mathematics and Computing Science, Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi, China.

E-mail address: dingy345@126.com (Y. Ding).

http://dx.doi.org/10.1016/j.future.2015.12.009
0167-739X/© 2015 Elsevier B.V. All rights reserved.

is necessary to encrypt different session keys for different digital items, which yields a large number of public-key operations if there is a large amount of data to be protected. The emerging cloud infrastructure is a typical case of big data environment and it is very challenging to apply PKCs in this infrastructure.

There are a large body of references on PKC systems. Among them, the Cramer–Shoup cryptosystem [1] derived from the ElGamal cryptosystem [2] is a practical public key cryptosystem with provable security against adaptively chosen ciphertext attacks (CCA2) [1]. In their work, they considered an adversary which cannot only compromise users by occasionally accessing their decryption devices, but also tamper the ciphertexts in order to obtain some useful information about the encrypted messages, even such an attacker cannot distinguish the ciphertexts of different messages. The strong CCA2-security is proven under a standard complexity assumption in the standard model. Due to its high-level security and practicality, the Cramer–Shoup cryptosystem has been adopted in international encryption standards [3] and widely deployed in industries.

In the Cramer–Shoup public key cryptosystem, also in many other discrete logarithm problem based cryptosystems, the most time-consuming operation is modular exponentiation. This operation significantly affects the performance of both encryption and decryption procedures. They are not applicable to secure big data where enormous of data are required to be encrypted.

There have been several approaches proposed to accelerate modular exponentiation. These approaches mainly fall into two categories. One is the batch exponentiation approach [4,5] that can reduce the time to compute multiple exponentiations with the same base, i.e., $g^{r_1}, g^{r_2}, \ldots, g^{r_n}$. It can only speed up the single-base exponentiations. The other is the fast multi-exponentiation approach [6–9] that can reduce the time to compute a multi-base exponentiation $g_1^{r_1} g_2^{r_2} \cdots g_m^{r_m}$. For multiple messages to be encrypted in Cramer–Shoup system, each base g_i will be randomized by distinct exponents. These methods would help to improve the performance of multiple encryptions, but are still expensive.

Moreover, in the Cramer–Shoup scheme, a valid ciphertext verification mechanism is employed to resist CCA2 attacks. In the case of multiple ciphertexts to be decrypted, the decryptor needs to perform a multi-exponentiation for each ciphertext. When the number of ciphertexts required to be verified is large, the consumed time and battery will be significant and may be unaffordable for mobile devices. Therefore, it is essential to allow many multi-exponentiations of the ciphertexts to be verified in batch to reduce the power and time consumption in decryption.

1.1. Our work

We investigate how to speed up discrete logarithm problem based public-key cryptosystems, with an emphasis on the well-recognized Cramer–Shoup cryptosystem, when there are a large number of messages and ciphertexts to be processed. Our contribution is twofold.

First, we propose batch multi-exponentiation to compute $\{g_1^{r_{i,1}} g_2^{r_{i,2}} \cdots g_m^{r_{i,m}}\}_{i>1}$ in batch. This is the most general case of batch exponentiations. We present two batch multi-exponentiation algorithms, i.e., basic batch multi-exponentiation (BBM) and simultaneous batch multi-exponentiation (SBM) for different configurations. The BBM algorithm requires the least memory, and thus applicable to storage-limited devices such as wireless sensors and mobile phones. The SBM algorithm improves the BBM algorithm with the ideas of simultaneous multi-exponentiation [10] and can reduce the computations of BBM by 25% when the number of different bases is two.

Second, we propose a batch PKC scheme, i.e., batch Cramer–Shoup cryptosystem, which allows multiple messages to be encrypted in batch and similarly, a number of ciphertexts to be verified also in batch. In practice, PKC is usually used to encapsulate session keys and then a symmetric encryption scheme is used to encrypt the digital content with that key. However, if one would like to enforce fine-grained access control over a large amount of digital contents, then one needs a large number of secret session keys to encrypt different digital items. In such scenarios, one has to employ the PKC scheme to process a large number of plaintexts (i.e., secret session keys) and ciphertexts (i.e., the encapsulations of the session keys). To address this problem, we employ our batch multi-exponentiation approach for batch encryption to reduce the computation overhead; and for batch decryption, we further incorporate techniques derived from available batch signature verification so that the received ciphertexts can be validated and decrypted in batch. We provide thorough theoretical analysis and conduct extensive experiments on the proposed batch Cramer–Shoup cryptosystem. Both the theoretical and the experimental results indicate that our batch multi-exponentiation algorithms greatly accelerate the Cramer–Shoup system, compared with repetitively invoking multi-exponentiation, by more than 40% in encryption and 80% in decryption. We also provide optimal batch size configuration in the case that some ciphertexts are erroneous.

We stress that although we only show the application of our batch multi-base exponentiation approach to the Cramer–Shoup cryptosystem. Our approach can also be applied to other public key encryption and signature schemes which need to process a large number of messages or ciphertexts. Our approach is also applicable to cryptographic protocols if the participants need to execute many multi-base exponentiations. Hence, our batch multi-base exponentiation algorithms are of independent interest and useful in practice.

1.2. Paper organization

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 proposes two batch multi-exponentiation algorithms for different configurations and provides detailed complexity analyses and comparisons. In Section 4 we propose batch Cramer–Shoup cryptosystem by applying our batch multi-exponentiation technique and batch ciphertext verification technique derived from batch signature verification, with thorough theoretical performance analysis and extensive experimental results. Section 5 concludes the paper.

2. Related work

Modular exponentiation has received considerable research attentions as it is the basic and also the most expensive operation in many widely deployed cryptosystems. The most well-known algorithm to compute modular exponentiation may be the binary method [11], also called "square and multiply" method. The average computation complexity is $1.5l$ multiplications (without discriminating a multiplication from a square which is slightly more efficient) where l is the bit length of the exponent. A good survey of early efforts to speed up modular exponentiation can be found in [12] and there are many other approaches [13–18]. Shamir's trick described in [2] improves the acceleration of multi-exponentiation by handling the identical bit of the exponents simultaneously, which reduces the number of multiplications and thus reduces the average computation complexity to $1.75l$ for two bases. The simultaneous 2^w-ary exponentiation method [7] handles w bits of an exponent once, which further reduces the number of multiplications. The simultaneous sliding window