Vladyslav Kuchin, Alireza Mirataei, Olexander Markovskyi

Method of Secure Modular Exponentiation on Remote Computing Platforms

The paper deals with a method of secure calculation of the modular exponent, which speeds up the operation by using remote capacities. In the proposed method, protection against disclosure of the base and exponent is implemented. The computation speedup provided by the developed method is estimated.

Key words: modular exponentiation, secure computation, remote resources, homomorphic encryption.

Target setting. Due to the widespread distribution of portable low-power devices with support for cryptographic information protection protocols, which use the modular exponentiation operation in their implementations, the question of attracting powerful cloud resources to accelerate the execution of this operation is relevant.

Actual scientific researches and issues analysis. In connection with the rapid spread of cloud technologies in recent years, the topic of remote secure information processing is increasingly common in scientific research.

Uninvestigated parts of general matters defining. Despite the existence of secure remote modular exponentiation methods, the computational speedup they provide remains low. This paper is aimed at increasing the level of computational acceleration that can be achieved through the use of remote capacities.

The research objective. The purpose of this paper is to increase the efficiency of the use of remote computing resources in the implementation of cryptographic algorithms for information protection, which are based on the operation of modular exponentiation.

The statement of basic materials. The task of the secure computation of the modular exponent is to create such an organization of computations that would at the same time ensure the protection of the base and the exponent and allow to speed up the calculations, thanks to the involvement of remote resources.

Опис запропонованого методу. The main idea of the method is to decompose the exponent in the form

$$d = a \cdot x + b, \quad (1)$$

where a and b are secret decomposition coefficients that ensure the hiding of the exponent, x is exponent that is passed to the remote platform for computation. The described decomposition occurs once at the start of the program. The digit capacity of the coefficients a and b is chosen according to the required level of security of the exponent d. Another important component of the developed method is the base m hiding mechanism. The basis of this mechanism is the RSA asymmetric encryption algorithm. The user needs to select the public and private keys (E, n) and (D, n) that match the condition $E \cdot D \bmod \varphi(n) = 1$, where φ is the Euler function, n is modulus. The user also calculates the product $y = D \cdot x$ where x is component of decomposition of exponent d. This product is sent to the remote platform as an exponent for calculations. Since the values of D and x are independent of the base m, y can be calculated once at the beginning of the program.

Before sending data, the user raises m to the power of E modulo n, obtaining $c = m^E \bmod n$. Taking into account that this operation is performed by user resources, E should have a small bit size to reduce the computation time.

The value of c is sent to the remote platform as the base, and the number y as the exponent. The remote platform must calculate and return to the user the result $r = c^y \bmod n$, which, taking into account the transformations described above, is $r \equiv c^y \equiv (m^E)^{D \cdot x} \equiv (m^{E \cdot D})^x \equiv m^x \pmod{n}$.

After the platform returns the calculated result r, the user is left to decipher the answer by performing such calculations: $z = (r^a \bmod n \cdot m^b \bmod n) \bmod n$, where r is result returned from the remote platform, m is base; a, b are coefficients of exponent decomposition.

Assessment of the level of security of secret data. The reliability of m-base encryption depends on the reliability of the RSA algorithm. As mentioned

above, in order to reveal the encrypted value $c = m^E \bmod n$, a potential attacker needs to solve the problem of factorization of the modulus n in an acceptable time, which today is considered an unsolvable problem.

The problem of revealing the encrypted exponent d is reduced to the problem of selecting such coefficients a and b that satisfy equality (1). Taking into account the capacity of the coefficients a and b, the total number of pairs (a, b) will be determined by the formula

$$N = 2^{l_a} \cdot 2^{l_b} = 2^{l_a + l_b},$$

(2)

where l_a, l_b are the bit lengths of the coefficients a and b, respectively. The above formula allows us to estimate the number of pairs of coefficients among which the attacker will brute force, trying to restore the secret exponent.

Assessment of efficiency. One of the ways to evaluate the efficiency of the method of secure calculation of the modular exponent is to compare the number of elementary operations performed in this method on the user side with the number of such operations that need to be performed when calculating the modular exponent without involving remote resources. The ratio of the size of calculations performed by the user and the remote platform is called the acceleration factor k. The modular exponentiation operation requires l_d to $2l_d$ modular multiplication operations, where l_d is the bit capacity of exponent d. Based on this, we can determine the average estimate of the number of multiplications performed as $N_{\text{зат.}} = 1{,}5l_d$.

The number of multiplications performed by the user will be determined by the capacity of the numbers a, b and E. Let the bit capacities of numbers a, b та E be equal to l_a, l_b and l_E, respectively. The $m^E \bmod n$ operation performed to hide the message m requires, on average, $1{,}5l_E$ multiplications. The remaining calculations to be performed after the remote platform returns the result require $1 + 1{,}5l_a + 1{,}5l_b$ multiplications. Thus, the total number of multiplication operations to be performed by the user in the method proposed by the author is

$$N_{\text{кор.}} = 1 + 1{,}5(l_a + l_b + l_E),$$

(3)

and the formula for the acceleration factor takes the form

$$k = \frac{1{,}5l_d}{1 + 1{,}5\left(l_a + l_b + l_E\right)}.$$

(4)

Based on the computational capabilities of modern computer systems, we can assume that the value of 50 for the bit capacity of the coefficients *a* and *b* will provide a high level of security for the secret exponent in most cases of practical application of the proposed method. Based on this value, and also taking into account that the most common value of the bit capacity of numbers currently used in the RSA cryptographic information protection algorithm is 2048, it is possible to calculate the acceleration factor that the developed method will provide in conditions of its practical use. Calculations performed using formula (4) allow us to conclude that when using a fairly common value of 16 for the bit capacity of the public key *E*, the calculation acceleration factor will be approximately equal to 18.

Conclusions. Theoretically substantiated, developed and experimentally investigated a method for the secure calculation of the modular exponent on remote computing power, based on the decomposition of the exponent and encryption of the base of the power using the RSA algorithm. It is shown that the proposed method makes it possible to speed up the calculation of the modular exponent by about 18 times while maintaining the security of the secret part of the data.

References

1. Boroujerdi N. Cloud Computing: Changing Cogitation about Computing/ Boroujerdi N., Nazem S. // IJCSI International Journal of Computer Science Issues, - Vol. 9, - Issue 4. -2012.- No 3.- PP. 169-180.

2. Xiaofeng Chen. New Algorithms for Secure Outsourcing of Modular Exponentiations / Xiaofeng Chen, Jin Li, Jianfeng Ma, Qiang Tang, Wenjing Lou // ESORICS 2012, LNCS 7459, - 2012.- PP. 541–556.

3. Can Xiang. Verifiable and Secure Outsourcing Schemes of Modular Exponentiations Using One Untrusted Cloud Server and Their Application // IACR Cryptology ePrint Archive 2014: PP.500 .- https://eprint.iacr.org/2014/500.pdf

4. Markovskyi O.P. Secure Modular Exponentiation in Cloud Systems./ Oleksandr P. Markovskyi, Nikolaos Bardis, Nikolaos Doukas, Sergej Kirilenko // Proceedings of The Congress on Information Technology, Computational and Experimental Physics (CITCEP 2015), 18-20 December 2015, Krakow, Poland, C. 266-269.

AUTHORS

Vladyslav Kuchin – bachelor, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: vladislavkuchin2001@gmail.com.

Alireza Miratei – PhD student, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: alirezaataei@gmail.com.

Markovskyi Oleksandr – associate professor, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: markovskyy@i.ua