INFORMATION SECURITY BASED ON THE DIFFICULTY OF FACTORIZATION OF ODD NUMBERS BY METHODS OF ELLIPTIC CURVES

G. Vostrov, *Ph. D., Associate Professor*
I. Dermenji

Odessa National Polytechnic University.

Abstract: In this article we analyze the problem of attack against cryptosystems, based on the difficulty of the factorization of odd numbers by using the Lenstra method. Effective ways of realizing this attack in modern realities and their economic feasibility were explored. In the course of the article, the reliability of systems based on the problem of factorization of large numbers in the attempts of hacking by the method of elliptic curves was proved. Specific ways of improving the already existing implementation of ECM were given; other more effective algorithms of attacks based on factorization were mentioned.

Keywords: cryptosystem, factorization, elliptic curve, public key, private key.

In modern investment projects for each period of work of the company reporting is formed. For such projects, it is extremely important to protect information from attacks from outside, since any leakage of information can lead to significant financial losses and a reduction in investment flows. Therefore, it is extremely important to have reliable means of information protection.

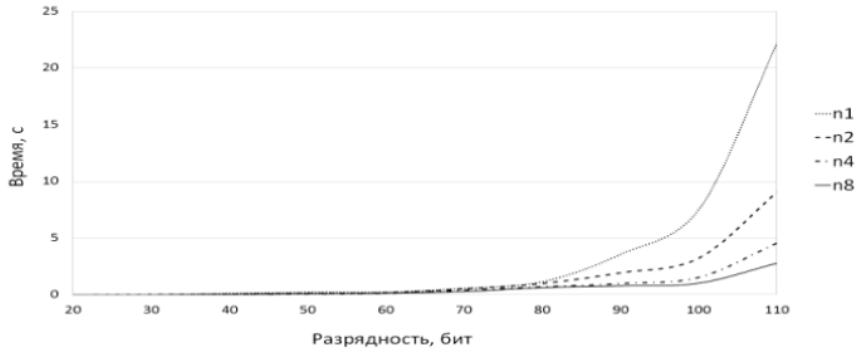
A large number of public-key cryptosystems are based on the practical difficulty of the factorization of large odd numbers, for example, cryptosystems based on the RSA algorithm. The RSA cryptosystem became the first system which is correct for both encryption and digital signature algorithms. RSA is used in a large number of cryptographic applications, including PGP, S / MIME, TLS / SSL, IPSEC / IKE and others [1]. The RSA keys of 1024 bits or more in length can be used as a reliable encryption system. Encryption key length of 1024 bit should be refused in the next three to four years [2].

The majority of attacks are associated with the misuse of the RSA algorithm as much as with the implementation of the RSA system and also with using small open or closed parameters. They all do not appear in case of competence in the implementation of the cryptosystem. The enemy can easily find the secret exponent and thereby hack the RSA by knowing the decomposition of the module into the multiplication of two prime numbers. It is currently unknown whether there exists an effective non-quantum factorization algorithm for integers. However, there is also no proof that there is no solution of this problem in polynomial time. The Lenstra elliptic-curve factorization or the elliptic-curve factorization method (ECM) is a fast sub-exponential [3] algorithm for integer factorization, which employs elliptic curves. In practice, it is mostly correct for finding small prime divisors of a number and therefore it is considered as a highly specialized algorithm. It is the best algorithm for finding simple divisors of 20 character length (64-83 bits in size), because its

difficulty basically depends on the smallest prime divisor p, and not on the factorized number [4]. This fact is a significant disadvantage of this method in this particular case, since the RSA algorithm is based on multiplying two large primes. Accordingly, the Lenstra method in this case is not optimal for attacking a cryptosystem based on it. For general-purpose factoring; ECM is the third-fastest known factoring method [5]. Let the smallest divisor of a number n equals p. Then the number of performed arithmetic operations can be estimated as: $e^{\sqrt{(2+o(1))\ln p;n(\ln p)}} \ln^2 n$

(Or $L_p\left[\frac{1}{2};\sqrt{2}\right]$ at L-notation).

It is possible to obtain an almost linear acceleration due to the parallel implementation of ECM with a distributed memory. Thus, an attacker can effectively use the opportunity to obtain a large amount of computing power with the help of cloud computing provided by a number of services, such as Amazon [7].



Pic.1. The measurement results for ECM; n1, n2, n4, n8 – using one, two, four and eight processors respectively [6].

The correct choice of boundaries $B_1 \ldots B_n$ allows getting the fastest running time of the algorithm. The Brent's table [8], which indicates the recommended boundary values for close numbers of a certain bit depth, can be used for the correct choice of such boundaries.

It is sufficient to increase the size of the RSA key to highly increase difficulty of factorization, even if an attacker is using a cloud computing. The increase in the productive capacity of the attacker, in the realities of the world, indicates an increase in the capacity of the user. Therefore, the security of the open RSA key is unquestionable without a significant breakthrough in the methods of factoring the number of its hacking in the acceptable time is not feasible.

In this case, it should be remembered about the economic feasibility of using a large amount of computing power and numbers of greater bit capacity should be used to protect more valuable information. Based on the above, the use of cryptographic applications based on the RSA algorithm, in conditions of the possibility of an attack using ECM, is economically justified. Sub-exponential algorithms are not able to provide a hacking system for an expedient time.

On the other hand, there is a question - is the use of algorithms based on the difficulty of factoring large numbers of odd is advantageous in comparison with

existing algorithms based discrete logarithm difficulty (DSA, ECDSA), which computational difficulty is defined as the exponential or even is algorithmically insoluble [9]. In particular, when it is possible to use a cryptosystem with a structure in which the computational complexity of the discrete logarithm is exponential, and at the same time a high level of resistance to unauthorized access to messages is guaranteed [10]. Also, the question remains on the security of such cryptosystems when quantum computers realize the already existing and grounded quantum Shore's factorization algorithm that can solve the factorization problem in polynomial time [11].

References:

1. Bakhtiari, Maarof [Text], 2012, p. 175.
2. Factorization of a 768-bit RSA modulus [Electronic resource] URL: https://eprint.iacr.org/2010/006
3. Parker D., Elliptic curves and Lenstra's factorization algorithm [Text] // University of Chicago: REU 2014. – 2014.
4. Lenstra Jr., Factoring integers with elliptic curves [Text]. *Annals of Mathematics* 126 (2): 649–673.
5. David Bressoud and Stan Wagon., A Course in Computational Number Theory [Text] – Key College Publishing/Springer, 2000. – C. 168-69. – 366 c. – ISBN 978-1-930190-10-8.
6. Makarenko A. Parallel implementation and comparative analysis of distributed-memory factorization algorithms/ A. Makarenko, A. Pyhteev, S. Efimov; Omsk State University. F. M. Dostoevsky – 2012 p. 94-109.
7. Easy Amazon EC2 Instance Comparison [Electronic resource] URL: http://www.ec2instances.info/
8. Brent, Richard P. (1999). "Factorization of the tenth Fermat number". Mathematics of Computation 68 (225): 429–451.
9. Manin Y., Panchishkyn A. Introduction to the modern theory of numbers [Text], - Moscow: MCCME, 2009. -552 p.
10. Vostrov G., Bezrukova J. Calculation of the discrete logarithm in modern cryptography – ELIT – 2017.
11. Shore P. W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer // Foundations of Computer-Science: Conference Publications. – 1997. – P. 1484–1509.
12. Klepikova O. A. Modeling of operational support of marketing solutions of the production enterprise by means of AnyLogic / O.A. Klepikova // Visnyk of Lviv University. Series: economical. - 2013. - Vip. 50. – p. 146-152.
13. Balan O.S. Management of the process of making investment decisions at the enterprises of production sphere: [monogr.] / O.C. Balan // Odessa: View of "VMV", 2014 - 420 p.