ISSN 1512-1232

UDC 004.02

USING GENETIC ALGORITHM FOR CRYPTANALYSIS CRYPTOALGORITHM MERKLE-HELLMAN

Lali Beselia

Sokhumi State University,Politkovskaia str., Tbilisi, Georgia

Abstract

The article considers possibility of using genetic algorithms in cryptanalysis, namely for cracking Merkle – Hellman cryptosystem. The experimental results show the efficiency using of genetic algorithms for cryptanalysis modern cryptosystems.

Index Terms: *genetic algorithms, cryptosystem, the system Merkley-Hellman cryptanalysis.*

I. Introduction

Merkle-Hellman‘s famous article was published in 1978 [1], which describes the public key (asymmetric) crypto system, based backpack task [2] One of the particular case, which can be summed up as follows: *V* the capacity knapsack and $B = \{b_1, b_2, ...b_n\}$ objects set, which have a certain capacities. Our task is to find a B subset of the abundance $B_i \subseteq B$ of the elements in the equation to be executed.

$$V = \sum_{i=1}^{n} b_i \cdot x_i$$

where, $x_i \in \{0, 1\}$, $i = 1,2,...,n$. If $x_i = 1$ it means that the i subject should be put knapsack, and if $x_i = 0$ the subject does not invest in knapsack. As is well known [2], knapsack task belongs to the group of NP complexity of the problem, but in this particular case, if the B set is the incremental sequence, the sequence of each b_i member satisfies the condition

$$b_i > \sum_{j=1}^{i-1} b_j ,$$

Then there's the task of the linear complexity of the algorithm [2].

Merkle and Hellman built the open crypto system using of this system. The encryption key $A = \{a_1, a_2, ...a_n\}$ is not reasonable for the increasing sequence, where *A* the sequence of each member a_i in the following method:

$$a_i = b_i \cdot t \pmod{m} \qquad (1)$$

and where $m, t \in Z$ the following conditions are fulfilled:

$$m > \sum_{i=1}^{n} b_i \qquad (t, m) = 1$$

Secret decryption of the key is triple (B, m, t). Open Text, which represents zeros everywhere and roll sequence, during the length of the encryption time n will be divided into a number L of blocks and performs the role $x_i \in \{0,1\}$ of abundance. Encrypted text $S_1, S_2, ..., S_L$ sums, which are calculated by the formula:

ISSN 1512-1232

$$S_j = \sum_{i=1}^{n} x_{ij} \cdot a_i \quad (2)$$

Open Text is needed to repair the problem solved backpack above option linear complexity algorithm, the known B and the Ascending and m and t options. To do this, multiplied t^{-1} by the sum of each module m

$$S_j^{/} = S_j \cdot t^{-1} (\text{mod } m) \quad (3)$$

and becomes a backpack to the solution of linear complexity algorithm separately for each S_j^i of the sum mentioned, when it is known B in the ascending sequence. In order to break the adversary system, and it will have to find the Open Text solution of NP complexity of the task, which is practically impossible, when two hundred to three hundred elements of B the sequence of the number are changed.

At a glance, this cryptosystem really was protected from any attacks and was the fastest public key system, the use of which has a large capacity to encrypt texts, but it was discovered that he had some failing[3], with which the famous scientist A.Shamir used polynomial difficulty Algorithm and broke the system [4].

From this failing first to note that the secret key from the public key reception, unlike other open-key crypto systems are not one-way function. Also, as it turned out, not necessarily to find exactly the (t_0, m_0) pair, with the help of which over increasing sequence returns the key - Not over increasing sequence. As it turned out, all of the incremental sequence, from which the additional beds A not in use can be obtained ascending secret key, or a key on the attack. These vulnerabilities by using A.Shamir us cryptoalgorithm to attack the system, which consists of two parts. In the first part of the algorithm to a whole number, which satisfied the conditions for the u / m values for some a_i of these functions is the minimum interval. Such numbers to find the algorithm Diophantine approximation method for (u, m) pairs, which will be possible to open the key to the secret key to calculate.

II. Genetic algorithms

Genetic algorithms originally used for solving optimization problems. Over time, he found the use of science in various fields. Genetic algorithms based on biological evolution is one of the basic principle: the fight to save the environment as much as possible adaptation of the population, which is achieved by strengthening and development of new generations of more and better features. Genetic algorithms for modeling of this principle is as follows: random solution set-elected candidates and the population of the genetic operators: selection, crossoving and mutation using a new generation of candidates accepted solution, which is closer to the average of the real solution, than those of the previous generation. It depends on how we use genetic algorithms and the quality of the criteria we have selected, or how to use the fitness function.

Genetic algorithms are one of the major advantages of search algorithms other than the possibility of their parallelization, which substantially reduce the attack.

III. Using genetic algorithm Merkle-Hellman algorithm is needed to spoil

Cryptographic algorithms for the analysis of genetic algorithms use a new direction, which is still unable to settle the practical cryptologists. There are dozens of works in which authors try to

ISSN 1512-1232

show the advantages of this approach can have a comparison with other methods. Our goal is to demonstrate the advantages of the use of genetic algorithms for the analysis of crypto Compared with other methods. It was at this point we took a crypto system has been broken Merkli-Hellman, the breaking of which we tried using genetic algorithms, and we compared the results obtained by the results of Shamir's algorithm.

There are some hard work, where merkli-Hellman crypto system is explained by means of genetic algorithms, but in all these cases the attack is made by means of a cipher text [7]. These studies, however, we're looking for the secret key of the public key to the technique similar to Shamir. Have developed a new heuristic methods, which resulted in the use of genetic algorithms is to make more accurate and quick. In this article, the results of the study can be used in other asymmetric crypto systems and software crypto analysis.

IV. Problems exploration

As mentioned above there are works, which discusses the use of genetic algorithms Merkle-Hellman crypto system for cryptanalysis. But our attack method is totally different from the methods used in the works. Also, we use the two different genetic algorithm, which is different from other genetic algorithms (different selection criteria and the quality of the crossover process). We carry out attacks on the cipher text of the key means. Our task is to find a (u, m) pair, the public key to find the incremental sequence of the following formula

$$b_i = a_i u (\mathrm{mod}\, m) \quad \text{Where } u=t^{-1}. \tag{4}$$

Asked to solve the problem, we have established two different algorithms. Their sale at the base of the C ++ language. Each algorithm consists of the preparation and the main part. Becoming part of the preparation of the information - Hellman algorithm to encrypt Merkle: We took $\{b_1, b_2, ...b_n\}$ an increasing sequence, **m** module base and have selected the t multiplier, which calculated the public key $a_i = b_i \cdot t (\mathrm{mod}\, m)$ and (3) the formula to be transferred encrypted information.

The first algorithm to work as follows:

1. The data from the solution set-candidates from the population, which is to take pictures of a random generator to initialize. The initial solution set-candidate of the Ascending (closed key), which is represented in a binary form. Its size is equal to **n**, where n is the size of the public key, and each of its members (the gene) Size 2d * n-1-n + i- created equal, where **d i**s the proportionality coefficient, and i for each bit rate index. Each member of the bits starts at 1.

2. With the help of a random generator for each of the Ascending introducing **m** base (binary form), whose length is equal to d * n (it must be the result of more than Ascending elements).

3. The population of each of the candidate and the corresponding solution set-**m** base resettled to the decimal system.

4. Use existing data, (1) obtained from the formula Diophantine equation $b_i t - km = a$, where k = 2,3,4; We find **t** a multitude of factors.

5**. t** multiplication of the many select only those **t**, for which **t <m** and **(t, m) = 1.**

6. The solution set-secret three candidates for (1) the formula calculating the open key.

7. Fitness function we set the quality of the selection criteria. In our case, the quality of the selection criteria, it is the 6th step of the Public key elements to match the Public key elements. If the number of items matching the key length is equal to or **n**, then the results obtained and the algorithm is complete, but if you do not match the number of key elements of the algorithm and continue moving to the next step.

8. The selection function of the selection function becomes the L / 2 (L is the initial population size), the number of candidates for the solution set-elect, whose fitness function is higher.

ISSN 1512-1232

9. The solution set-elected candidates carry out the function of crossover. Crossoving Function are as follows:

A) random generator, the solution set, choosing between two candidates, the candidate solution set (the selection is made so that breeding pairs are not repeated).

B) Each candidate solution set-divided into two parts and then adopted two new solution set-candidate. Usually, two parent solution set-up as a candidate for the four successor, but in our case we get only two successor candidate solution set, we are interested only in the read. Ascending.

10. After Crossover operation, we calculate the solution set-candidates' fitness function, if any solution set-candidate's fitness function is not equal to n, then repeat on the 2nd, 3rd, 4th, 5th, 6th, paragraph 7, 8, 9 steps 5 times, if we got the desired result, we continue to work on the algorithm, otherwise moving to the next step.

11. The use of mutation function. Then repeat the 10th steps and decide upon any results of the algorithm to work.

The algorithm of the experiments showed that most of the fitness function does not exceed n / 2-'s (half the length of the key). This result is not the best, but this result may be encrypted information from the idea.

That's why we changed our approach and established a second algorithm, which work as follows:

1. The initial population of the m base, which initialize take pictures of random generator (to be submitted in a binary system), while the population of each member (solution set-candidates) size is d * n; Where n is equal to the length of the public key, and d = 2;

2. take pictures of the solution set - Dozens of candidates for transfer to the system.

3. Shamir algorithm take the first four members of the public key and calculate the inverse of multiplication **t,** where the m base $u = p * m / a_i$, $u = t^{-1}$, $1 <= p <= a_i$, $0 <= i < 4$;

To select u multiplier, we impose certain restrictions. Besides this (um) = 1 and u <m, multiplier u, multiplied by the third member of the public key, must be greater than the base m. When you add this restriction, we reduce the set of u candidate-solutions. (u, m) pairs in all of the possible candidates for a new key to the closed (3) formula.

4. Fitness function we set the quality of the selection criteria. In this case, the quality of the selection criteria, the 4th step of the private key of the ascending. If the sequence of the rise of the fitness function value is equal to n, then the results obtained and the algorithm is complete, but if the fitness function n- than moving to the next step.

5. Crossover function solution set-elected candidates are carrying out the function of crossover. This algorithm is essentially different from the previous algorithm Crossoving function from the function.

A) Random generator, the solution set, choosing between two candidates, the candidate solution set (the selection is made so that breeding pairs are not repeated).

B) Each candidate solution set is divided into two parts (the middle point) and their Crossoving results in a four-candidate solution set.

6. The newly-adopted solution set for candidates to repeat the 2nd, 3rd, 4th steps, if any of the candidate's fitness function solution set-n- created equal, then the desired result is obtained, and the application is closed, otherwise moving to the next step.

7. the selection function selection function becomes the L (L is the initial population size) of the solution set-selection of candidates, whose fitness function is higher.

8. selected solution set-candidates for repeat steps 5 and 6.

We have indicated that the process will repeat 10 times. If this process is repeated, 10 times and we don’t get the desired result, only in this case we use a mutation, or change the function of the gene, and then repeat the 2nd, 3rd and 4th steps. When we get the desired results, we stop working. But tests showed that non f the mutations feature is not needed, and the hybridization of a maximum of 5 times using we get the desired result.

ISSN 1512-1232

Key length is 20, the population size of 50

Input data	Secret key	m	t
	123 127 268 535 1147 2312 4578 9290 18591 37187 74365 148900 297807 595612 1191223 2382452 4764898 9529801 19059651 38119281	76238574	3317
Public kay	407991 421259 888956 1774595 3804599 7668904 15185226 30814930 61666347 47110705 17952983 36469856 72962931 69680654 63119417 50020162 23781848 47580281 19084521 38099385		
The result (again after crossover operation)	Secret key	m	t^{-1}
	761565 5503131 119434097 355972053 482396448 964792896 1954429202 4357957121 9643464323 38544999919 64634809024 127922914623 296330170245 550175999421 1516290718213 2729597315878 5914568234578 12456789345670 25789087654329 223456789023456	989765234870785	2829323
Lead time	8,72 sec.		

Experimental results show that the use of genetic algorithms Merkle -Hellman crypto system is broken very quickly. Therefore, we can conclude that the use of genetic algorithms will be effective for other asymmetric crypto systems for cryptanalysis.

Reference

[1] Merkle R.C., Hellman M.E. Hidding information and signatures in trapdoor Knapsak, IEEE Trans. Inform. Theory, IT-24 (1978), pp. 535-530.

[2] Martello, S. and P. Toth, Knapsack Problems: Algorithms and Computer Implementations, John Wiley &; Sons, West Sussex, England, 1990.

[3] A. Salomaa, Public-Key Cryptography, Springer-Verlag, 1990.

[4] Garg P, Shastri A. An Improved Cryptanalytic Attack on Knapsack Cipher using Genetic Algorithm. International Journal of Information Technology, 3(3) (2006) 6.

[5] Muthuregunathan R., Vekataraman D., Rajasekaran P. Cryptanalysis of Knapsack Ciher Using Parrallel Evolutionary Computing. International Journal of Recent Trends in Engineering, Vol. 1, No 1, 2009.

[6] Shamir A. A Polinomial-Time Algorithm for Breaking the Basic Merkle-Hellman Cryptosystem. IEEE Transactions on Information Theory. Vol., IT-30, No5, 1984, pp. 699-704.

[7] R. Geetha Ramani Genetic Algorithm solution for Cryptanalysis of Knapsack Cipher with Knapsack Sequence of Size 16. International Journal of Computer Applications (0975 – 8887) Volume 35, No.11, 2011.

Article received: 2016-06-09