UDC 681.3.06.

MODELING OF DYNAMIC DATA PROTECTION PROCESSES BASED ON A DISCRETE LOGARITHM

G. Vostrov, Yu. Bezrukova
Odessa national polytechnic university

Abstract. *The primary purpose of this article is the modeling a dynamic system of an electronic digital signature based on the theory of discrete logarithms in combination with the type of protocol, such as Diffie-Hellman. In the course of the work, the main aspects of this scheme such as a hash-function, a key set and a key exchange protocol were considered. There were specified the conditions for the formation of a set of keys for an electronic digital signature algorithm; a step-by-step algorithm for computing the discrete logarithm was constructed, indicating the key aspects of solving this problem in programming implementation.*

Key words: *cryptographic stability, cryptographic analysis, electronic digital signature, key set, discrete logarithm, finite field, hash function, computational complexity.*

Introduction

The theory of the evolving three worlds system confirms that the amount of an emerging information exponentially grows in time [1]. Most of such information are represented as messages, which are organized in a certain form and located on different types of media. Messages are exposed to various influences. On the one hand, there is a group of consumers, which limits access to the information with setted conditions. On the other hand, there are people, who try to get the off-limits access to the information.

Cryptography is a science that deals with limiting the access to the information. This science is a combination of methods of protecting information from unauthorized access and at the same time one of the sections of modern mathematics. In turn, there is a cryptanalysis that deals with overcoming such limitations. At its core, this science is developing ways to uncover methods of cryptographic protection. The parallel development of these processes undoubtedly leads to their continuous improvement [2].

Cryptographic protection methods are always aimed at solving the following problems:

1. Guaranteed integrity of information in the processes of its storage and transmission in modern networks;
2. Providing access to the information exclusively to its legitimate users;
3. Absolute authentication of the information according to the selected forms of its presentation;
4. The inability of refusing the authorship;
5. Providing of a consistent analysis of messages dynamics which guarantees that the information is not monitored.;

Modern cryptography deals with the solution of the whole complex of these tasks. As a scientific direction modern cryptography has become an important area of mathematics, but at the same time it has an applied significance. Therefore, in modern cryptography computer methods and technical instruments of protecting information are applied and developed.

The primary purpose of this work is to improve methods of an electronic digital signature which represents as a requisite of an electronic document. This type of signature is created using a cryptographic transformation of the information in a message with the help of a user's secret key. By using this transformation we can check if the message was changed during its transfer. The main usage of a signature is to confirm the author

The process of forming an electronic signature has the key points that are considered in this article. One of the predetermining moments is the keys generation. Randomness is the most important key's characteristic [2]. The digital signature protocol implies the existence of private keys of both users and also the existence of a common public key. The main problem is to create a generator with a uniformly distributed random sequence. Such sequence is called a set of keys. The concrete value will be randomly get from this sequence. This type of sequence should guarantees the inability of its recovering by using its fragment. In the computational sense there should be no regularities in the sequence.

The second point is to provide a secure key exchange method. Using the Diffie-Hellman protocol is very perspective [3]. This protocol lets both of users to come up with key without personal meeting. The strong of this protocol is based on the intractable discrete logarithm problem [4]. The most important question is the existence of a such

algorithm [5]. The Diffie-Hellman protocol has a disadvantage which describes as an inability of a mutual authentication. It means that the existence of this protocol in the algorithm is not enough to provide a secure key exchange [4].

The existence of an electronic digital signature predetermines the presence of any certified message the integrity of which can be verified upon receipt. This possibility exists due to using a hash function over the message which does not allow opening the message itself. The using of cryptographic hash functions allows to create a signature scheme without restoring the message. It is much more efficient for long messages [4].

One of such schemes was presented in the RSA method [6], which is based on a sub exponential complexity of a large numbers factorization. However, in the modern world the rapid growth of computing power makes it possible to improve the effectiveness of cryptanalysts and therefore reduces the relevance of RSA schemes. Moreover, this method has some disadvantages which are brought about by additional requirements to numbers that are used in the signature algorithm. These restrictions increase the probability of disclosing the values of the electronic signature by cryptanalytics. Due to the actual drawbacks of a such scheme recently more attention has been paid to algorithms based on calculating the discrete logarithm.

The increase of cryptanalytic possibilities bring about the necessity of creating a dynamic cryptography. The system must be constructed in a way that it can monitor hacking attempts and make a decision of changing cryptographic secure parameters. It is often presumed that keys must be changed occasionally, but it is a wrong strategy because the main aspect is how effective the hacking attempt was. So, the system should control not only the number of attempts but their affectivity. This requirement is very important because cryptanalytic can decrypt the scheme using previous attempts results.

At present, the creation of dynamic algorithms for electronic digital signature based on discrete logarithm is a subject of an intensive research. The main task of this work is to develop more efficient methods of constructing an electronic digital signature in a Diffie-Hellman type system, including the generation of multiple keys, the key exchange protocol, the signature generation protocol, and the development of an algorithm for computing the discrete logarithm.

Procedures of using the electronic digital signature

There are two types of algorithms in cryptography: symmetric and asymmetric (with public key). Sometimes symmetric algorithms are called conditional algorithms in which the encryption key can be calculated by the decryption key and vice versa. The security of this algorithm is determined by a key that means that everybody who knows the key value can encrypt and decrypt the message. This is why the asymmetric algorithm is the most usable. They are constructed in a way that the encryption key is a different from decryption one, moreover there is no ability to compute the decryption key using an encryption one.

Electronic digital signature scheme is one of the most important cryptographic area with a public key. It cannot be presented as the encryption method, but only as the method of authentication of user which is the author of a concrete message.

Electronic digital signature protocol:

The scheme of an electronic digital signature with an asymmetric encryption system as in [4] has the following structure: <Message> + <Sender's secret key> = <Signature> and an authentication of a signature has the following structure: <Message> + <Signature> + <Recipient's public key> = <Yes/No>.

This scheme can only verify the authenticity of the signature. It is possible to create a system that will decrypt the message.

Electronic digital signature protocol assumes that secret keys are created in the beginning of the process and could not be published. Also it is supposed the existence of a public key that depends on a secret key value and a message hash value. After formation the signature transfers to the recipient. Each modification of a message will be detected because of signing the hash value instead of the whole message. It means that we can use an unprotected channel to transfer data. If the recipient knows the public key he can authenticate the author and check the message integrity.

The main advantages of an electronic digital signature protocol:

1. Guarantee of the authenticity of the signature due to the non-disclosure of the sender's secret key;
2. Unable to correct the signed message. When the message changes, the value of the cryptogram changes. It leads to a mismatch between the calculated and received value;
3. The sender cannot refuse the signed message, since the recipient does not need help from the sender to verify the signature;
4. The signature is a part of a document, so it can be used only once;

If documents should not be used several times we can add an information block to the message.

This block consists of data which is called a timestamp. It may include an identification information or current date and time. Due to ability of checking a timestamp, cryptanalyst cannot use this document several times.

The decryption of an electronic signature value means that cryptanalyst can replace documents and use a signature without consent of the owner. It is necessary to develop a dynamic system to provide a guaranteed scheme protection. It can be obtained by changing secret key values or a hash function depending on hacking attempts.

The key set forming mechanism

Cryptography system's strong depends on the method of a key set building.

A complex of possible keys values is called a key set. One of the main characteristic of this set is a number of possible keys.

The main characteristic of a key is its size (a length of a word in a some alphabet) [2]. It is necessary to consider next facts to define optimal size: choosing too small size can make it easier to decrypt the scheme, at the same time too big size can complicate the computational aspect of the system.

Key values must be random. The system has the best cryptographic properties if the key set is represented as a random ideal sequence of random variables with a uniform probability distribution on a given finite alphabet [2]. The fundamental task in creation the electronic digital signature is to develop the sequence generator. Known deterministic devices cannot be used, since in such cases the presence of inter-sign dependencies is possible [2].

In programming realization pseudorandom generators can be used. An ensemble is pseudorandom if and only if it is unpredictable in polynomial time. An ensemble is called unpredictable in polynomial time if for every polynomial-time algorithm A, every positive polynomial $p(\cdot)$ and all sufficiently large n:

$$P[A(1^{|x_n|}, x_n) = next_A(X_n)] < \frac{1}{2} + \frac{1}{p(n)},$$

where $next_A$ returns the $i+1$ bit of x, if on input $(1^{|x|}, x)$ algorithm A reads only $i < |x|$ bits of x and returns a uniformly chosen bit otherwise [7, 8].

Pseudo-random sequences still have a drawback due to the possible presence of regularities. This fact is defined by the iteration function which has stationary points, which determined pseudorandom sequences. This entails an internal periodicity in computer modeling. If the cryptanalyst was able to acquire any information about the device of the generator the system is not protected from hacking. In the dynamic system this problem can be solved by controlling hacking attempts. It lets programmers to rebuild completely a key set under certain conditions. Due to this ability programmers can reduce the probability of a set recovery.

One of the main problem is to create a reliable generator such as discrete exponential generator with a different variations. The safety of this generator is based on a complexity of a discrete logarithm computation [6].

We have explored different methods of generating pseudorandom sequences for different types of maps [9].

Key exchange protocol

The exchange of keys between remote users without using a secure channel is an important advantage of a public-key cryptography over symmetric cryptosystems. Diffie and Hellman offered the first scheme. Afterwards, this scheme was called the Diffie-Hellman exponential key exchange scheme [6].

Consider the key exchange protocol. It is necessary to have a plurality of pairs of values (p, g): $p-$ large prime number, g - generating group element F_p^*. Where F_p - a finite field with the number of elements equal to p. F_p^* - multiplicative group of this field with a generating element g. This group is cyclic. It means that there is a generating element in it and all other elements are obtained by exponential to the power of the generating. So, we can note that for every $a \in F_p^*$ exists n that $g^n = a \bmod p$

The result of the protocol is the element which is common to both sides.

1. The sender generates the item $a \in [1, p-1]$ calculates the number of $g_a \leftarrow g^a (\bmod p)$ and sends it to the recipient.
2. The recipient generates the item $b \in [1, p-1]$ calculates the number of $g_b \leftarrow g^b (\bmod p)$ and sends it to the sender.
3. The sender computes the value $k \leftarrow g_b^a (\bmod p)$.
4. The receiver calculates the value $k \leftarrow g_a^b (\bmod p)$.

Considering the fact that $ab \equiv ba (\bmod p-1)$ both sides calculate equals value. It guarantees the distribution of the key between a sender and a recipient [10].

The complexity of this algorithm is provided by calculating $g^{ab} (\bmod p)$. It is called the Diffie-Hellman problem. This problem essentially relies on

the complexity of computing discrete logarithm [10]. Both problems are quite difficult to be solved, but the existing algorithms show that the computational complexity of these schemes increases with the size of the field F_p and the values of the selected parameters. Despite of this fact, these problems can be solved for small values.

Many effective versions of this protocol are based on the elliptic curve group [4]. This can be explained by the fact that for points of an elliptic curve the problem of discrete logarithm is less than the complexity of this problem in the general formulation for an arbitrary group.

This protocol has a disadvantage: it does not guarantee common authentication. In this case, the probability of an attack "man in the middle" increases. This is why the algorithm must be improved. One of the possible solution of this problem is a digital certificate which can distribute public keys through the trusted channels. A system consisting of certificate centers and certificates themselves is usually called a public key infrastructure. It allows to "redistribute trust". Thus, the verification of the identity which the key belongs is replaced by the trust to the certification authority and its correct operation [4].

It is suggested to solve the problem in the following way: users have to receive a reliable public key from the certification authority and only then the secret session key will be generated using a protocol with signatures [4].

Constructing and using a hash function

Cryptographic hash function is a function that is easy to compute but hard to invert. Saying that this function is easy to compute means that there exists a polynomial-time algorithm that on x outputs $f(x)$. The second condition means that every probabilistic polynomial-time algorithm on input y to find an inverse of y under f may succeed only with negligible (in $|y|$) probability, where the probability is taken over choices of y [7].

In an electronic digital signature not the message itself is signed, but the result of the hash function over the message. The hash function must be constructed in such a way that it is not possible to get the information contained in the message using the result of the hash function. In addition, these values should not be the same for different input data blocks.

The cryptographic strength of any digital signature scheme using cryptographic hash functions depends both on the complexity of the mathematical problem, such as factorization or discrete logarithm and on the strength of the hash function [4].

Cryptographic hash function that is used in the electronic digital signature algorithm must satisfy the requirement: at the slightest change in the message the value of the hash function must be significantly different. The inability of confirmation a changed document is based on this requirement.

Building an electronic signature using the discrete logarithmic method

The problem of discrete logarithm is computationally more complicated in comparison to the factorization problem. The complexity of calculating the discrete logarithm has either exponential complexity or is algorithmically unsolvable. The using of this method in a public-key cryptography and in a digital signature algorithm is based on this fact. A specific algorithm is developed in this article. The feature of this algorithm is that the complexity of the algorithm exponentially grows with increasing the value of a prime number.

It is necessary to introduce some definitions to consider the problem of finding a discrete logarithm:

A *finite group* is an algebraic group that contains a finite number of elements which is also called the "order" of the group.

Suppose that G is a finite group, $b-$ an element of G and $y-$ an element of G, which is an exponent of b. Any integer x for which $b^x = y$ is called discrete logarithm y for base b.

Until now there is no known effective algorithm for computing the discrete logarithm (or index) $x = ind_t a$ which is determined for reversible residue $a(\mathrm{mod} q)$ according to the formula $a \equiv t^x (\mathrm{mod} q)$ where $x \in [0, q-1]$. The most important open question is the existence of such an algorithm [5].

Nevertheless, there is a way to find a discrete logarithm in finite fields, but only for the case when all prime multipliers p of $q-1$ is not a large numbers. Firstly, for every simple p we should calculate p th root of 1:

$$r_{p,j} = t^{\frac{j(q-1)}{p}} \quad \text{where} \quad j = \overline{0, p-1}$$

in a multiplicative group $(Z/qZ)^{\times}$. In this case it is convenient to calculate using the *method of repeated squaring* [11].

To calculate greater powers of $a^m (\mathrm{mod} n)$, where $m = n-1 = d_{k-1}2^{k-1} + d_{k-2}2^{k-2} + ... + d_0$ – is a binary representation of a number $n-1$ c

$d_{k-1}=1$ and $d_i=0,1$, define consistently the number $r_1=a$ and

$$r_{i+1}=\begin{cases} r_i^2 \pmod n, if\ d_{k-1-i}=0; \\ a r_i^2 \pmod n, if\ d_{k-1-i}=1; \end{cases}$$

The result is $a^{n-1}\equiv r_k \pmod n$ so that $a^{n-1}=(...((a^{2+d_{k-2}})^2 a^{d_{k-3}})^2...)a^{d_0}$

The algorithm of exponentiation is fast enough, it is polynomial and requires no more than $3[\log_2 n]$ multiplications modulo n to find the number r_k.[5]

Thus the table is constructed:

Table 1

Residues table $r_{p,j}$

	j_0	...	j_{p_0-1}	...	j_{p_s-1}
p_0	1	...	$t^{j_{p0-1}(q-1)/p_0}$	...	
...	...	...	...	...	
p_s	1	...	$t^{j_{p0-1}(q-1)/p_s}$	...	$t^{j_{ps-1}(q-1)/p_s}$

where $\{p_i\}_{i=\overline{0,s}}$ - prime divisors of a number $q-1$ $\{j_i\}_{i=\overline{0,p_s-1}}$ – ordered set $\{0,1,...,p_{s-1}\}$.

After a table constructing, where $p|(q-1)$ and $(q-1)=\prod_{p|q-1} p^{\alpha_p}$ – decomposition of the number $(q-1)$ use the *Chinese theorem on residues* [6]:

Suppose that $m_0...m_{r-1}$ - positive pairwise relatively prime modules and $M=\prod_{i=0}^{r-1} m_i$, let also be given r corresponding deductions n_i Then the system of r equations and inequality $n\equiv n_i \pmod{m_i}$, $0\le n<M$ has the only solution. Moreover, this solution is exactly the smallest nonnegative residue modulo M of a number $\sum_{i=0}^{r-1} n_i v_i M_i$ where $M_i=\frac{M}{m_i}$ and v_i - inverse elements which determine from the relations $v_i M_i=1 \pmod{m_i}$ [6].

Following the theorem, it is suffice to find $x \pmod{p^{\gamma_p}}$ for each $p|(q-1)$. Some primary p which divide $(q-1)$ and $\gamma=\gamma_p>0$ must be fixed. Then the algorithm of finding $x \pmod{p^{\gamma}}$ must be described.

Suppose that:

$$x\equiv x_0+x_1 p+...+x_{\gamma-1}p^{\gamma-1} \pmod{p^{\gamma}},$$

and $0\le x_i<p$. Firstly, x_0 value must be determined. To do this we should calculate the value $a^{(q-1)/p}\in(Z/qZ)^{\times}$, which is the p th root of 1, since $a^{q-1}\equiv 1 \pmod q$. From equality $a\equiv t^x \pmod q$ follows that $a^{(q-1)/p}=t^{x(q-1)/p}=t^{x_0(q-1)/p}=r_{p,x_0}$. Thus, comparing the value $a^{(q-1)/p}$ to $\{r_{p,j}\}$, where $0\le j<p$ we can suppose that x_0 is equals to the value of j at which $a^{(q-1)/p}=r_{p,j}$ [11].

To find x_1 we should change a to $a_1=\frac{a}{t^{x_0}}$. Then a_1 has a discrete logarithm such as $x-x_0=x_1 p+...+x_{\gamma-1}p^{\gamma-1} \pmod{p^{\gamma}}$. Since a_1 is a p th power we find that $a_1^{(q-1)/p}\equiv 1 \pmod q$ and $a_1^{(q-1)/p^2}\equiv t^{\frac{(x-x_0)(q-1)}{p^2}}\equiv t^{\frac{x_1(q-1)}{p}}\equiv r_{p,x_1}$. Thus, comparing the value $a_1^{(q-1)/p^2}$ with $\{r_{p,j}\}$ we can suppose that x_1 is equals to the value of j at which $a_1^{(q-1)/p^2}=r_{p,j}$ [11].

Next, to get the entire sequence x_i, $i=\overline{0,\gamma-1}$, we may suppose that $a_i=\frac{a}{t^{x_0+x_1 p+...+x_{i-1}p^{i-1}}}$. Thus, a_i is a p^i th power, so $a_i^{(q-1)/p^i}=1$ and $a_i^{(q-1)/p^{i+1}}=t^{(x_1+x_{i+1}p+...)(q-1)/p}=t^{x_1(q-1)/p}=r_{p.x_i}$. Therefore, we assume that x_i is equals to the value of j at which $a_i^{(q-1)/p^{i+1}}=r_{p,j}$ [11].

The result of this process is a value of $x \pmod{p^{\gamma}}$. Repeating such calculations for all $p|(q-1)$ we can find x using the *Chinese theorem on residues* [11].

As mentioned earlier, this algorithm is applicable only for small values of the prime divisors of a number q. In the computational sense, the most time-consuming step is to compile a table. The main problem is that we are not allowed to use one table for several times. When we change the value of q it should be completely rebuilt by recalculating all the values. It should be noted that for an arbitrary q with a large order the size of the table becomes incommensurably large.

The following problems were solved for the program implementation of this algorithm:

1. Development of an effective method for selecting the type of numbers;
2. Construction of methods for controlling the performance of arithmetic operations with respect to rounding of numbers. Since at the slightest loss of the number order the result of the calculations can differ substantially;
3. Defining a table storage option $r_{p,j}$ so that the insertion and indexing operations are carried out with the least amount of computer time.

Conclusion

In this article a solution of the problem of constructing an electronic digital signature scheme based on the method of discrete logarithm was found. For this method a step-by-step algorithm for calculating the discrete logarithm, indicating the key aspects and features of the solution of this problem in software implementation was constructed.

The problem of the development of the discrete logarithm method was solved in combination with the problems of generating a key set and choosing a cryptographic hash function that meets the requirements of the task.

A way of key exchange during signature generation was defined. The practical significance and merits of using the Diffie-Hellman protocol were determined. Modification of this protocol allows to build reliable schemes for electronic digital signatures. It was defined that this protocol is the most perspective to be used in the electronic digital signature scheme.

Also, the problem of the computational complexity of algorithms in describing methods for constructing a key set was considered. When constructing an algorithm for an electronic digital signature, one should evaluate both the computational complexity of each structural component and the entire algorithm as a whole. In addition to generating a key set and the algorithm of discrete logarithm, a great attention should be payed to the estimation of the computational complexity of the hash function algorithm. In the course of the work, it was proved that using a hash function over a message makes it possible to significantly reduce the computational complexity of the entire algorithm. It can be explained by the fact that the messages can be quite large while the hash value has a constant character.

In the process of developing an electronic digital signature scheme, it was proved that this method based on the discrete logarithm is more reliable for signing documents that are of financial, legal or economic nature than the discrete logarithmic algorithm based on elliptical curves.

References

1. Penrose, R. (2007), The path to reality or laws governing the universe, - Moscow, Izhevsk, R&C Dynamics, Institute of Computer's Research, 911 p.
2. Fomichev, V. (2010) , Methods of Discrete Mathematics in Cryptology, - Moscow: Dialogue-MEPhI, 424 p.
3. Diffie W., Hellman, M.E. (1976), New directions in cryptography, - IEEE Trans. Info. Theory, IT-22(6):644-654.
4. Smart, N. (2005), Cryptography, - Moscow: Techno sphere, 528 p.
5. Manin, Yu. , Panchishkin, A. (2009), Introduction to the modern theory of numbers, - Moscow: MSC-MO, 552 p.
6. Crandall, R., Pomerance, K. (2011), Prime numbers: cryptographic and computational aspects, Transl. from English / Ed. and with a preface by V. Chubarikova, - Moscow: URSS: Book House "LIBROKOM", 664 p.
7. Goldreich, O. (2004), Foundation of Cryptography. Basic Tools, Cambridge university press, 393 p.
8. Goldreich, O. (1999), Modern Cryptography, Probabilistic Proofs and Pseudorandomness, Springer, 199p.
9. Vostrov, G., Khrinenko, A. (2017) Computer modeling of the processes of chaos formation in nonlinear dynamic maps, - ELTECS.
10. Mao, V. (2005), Modern Cryptography: Theory and Practice, Trans. from English. - Moscow: Publishing house "Williams", 768 p.
11. Koblitz, N. (2001), Course of number theory and cryptography, - Moscow: Scientific publishing house PTA, 254 p.

МОДЕЛИРОВАНИЕ ДИНАМИЧЕСКИХ ПРОЦЕССОВ ЗАЩИТЫ ИНФОРМАЦИИ НА ОСНОВЕ ТЕОРИИ ДИСКРЕТНОГО ЛОГАРИФМА

Г. Н. Востров, Ю. С. Безрукова
Одесский национальный политехнический университет

Аннотация. *В данной работе созданы методы построения динамической системы электронной цифровой подписи, основанной на дискретном логарифмировании, в сочетании с протоколом типа Диффи-Хеллмана. Определены преимущества использования такого метода и его криптографическая стойкость в сравнении с другими схемами. Построен метод нахождения значения дискретного логарифма в конечных полях с указанием ключевых аспектов и особенностей решения данной задачи при программной реализации. Был определен способ обмена ключами при формировании подписи, основанный на протоколе Диффи-Хелмана, для которого дополнительно указан метод взаимной аутентификации. Задача о развитии метода дискретного логарифмирования решалась в комплексе с проблемами генерации ключевого множества и выбора криптографической хэш-функции, соответствующей требованиям поставленной задачи.*

Ключевые слова: *криптографическая стойкость, криптоанализ, электронная цифровая подпись, ключевое множество, дискретный логарифм, конечное поле, хеш-функция, вычислительная сложность.*

Received 27.04.2017

George Vostrov, Ph. D. of Technical Sciences, Associate Professor of the Department of Applied Mathematics and Information Technologies, Odessa National Polytechnic University. Shevchenko ave., 1, Odessa, Ukraine.

E-mail: vostrov@gmail.com, mob. +380503168776

Востров Георгий Николаевич, кандидат технических наук, доцент кафедры прикладной математики и информационных технологий Одесского национального политехнического университета. Просп. Шевченко, 1, Одесса, Украина.

ORCID ID: 0000-0003-3856-5392

Yulia Bezrukova, Student of the Department of Applied Mathematics and Information Technologies, Odessa National Polytechnic University. Shevchenko ave., 1, Odessa, Ukraine.

E-mail: bezrukovajulia961@gmail.com, mob. +380500341872

Безрукова Юлия Сергеевна, студент кафедры прикладной математики и информационных технологий Одесского национального политехнического университета. Просп. Шевченко, 1, Одесса, Украина.

ORCID ID: 0000-0002-0577-2216