Харьковский национальный университет радиоэлектроники

Академия наук прикладной радиоэлектроники

ПРИКЛАДНАЯ РАДИОЭЛЕКТРОНИКА

Научно-технический журнал

И.о. главного редактора

Чурюмов Г.И.

Зам. главного редактора

Дохов А.И.

Редакционный совет

Гузь В.И., Довбня А.Н., Егоров А.М., Калугин В.В., Кравченко В.И., Назаренко И.П. (Россия), Неклюдов И.М., Пресняк И.С., Симонов К.Г. (Россия), Симанков В.С. (Россия), Слипченко Н.И., Чабдаров Ш.М. (Россия), Яковенко В.М., Ярошенко В.С. (Россия)

Редакционная коллегия

Абрамович Ю.И. (США), Бодянский Е.В., Борисов А.В., Буц В.А., Бых А.И., Гомозов В.И., Жуйков В.Я., Зарицкий В.И., Кипенский А.В., Кульпа К. (Польша), Леховицкий Д.И., Литвинов В.В., Лукин К.А., Мачехин Ю.П., Модельский Й. (Польша), Нерух О.Г., Поляков Г.А., Ролинг Г. (Германия), Седышев Ю.Н., Серков А.А., Сухаревский О.И., Чурюмов Г.И., Шифрин Я.С., Шкварко Ю.В. (Мексика)

Адрес редакции:
Редакция журнала «Прикладная радиоэлектроника»
Харьковский национальный университет радиоэлектроники
просп. Науки, 14, 61166, Харьков, Украина
Тел.: + 38 (057) 702 10 57
Факс: + 38 (057) 702 10 13
E-mail: are@kture.kharkov.ua
http://www.anpre.org.ua

© Харьковский национальный университет радиоэлектроники, 2015

СОДЕРЖАНИЕ

МЕТОДЫ И СРЕДСТВА АССИМЕТРИЧЕСКИХ КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

Gorbenko I., Kachko O., Pogrebnyak K. Features of parameters calculation for NTRU algorithm 272

Бессалов А.В., Цыганкова О.В. Классификация кривых в форме Эдвардса над простым полем 277

Горбенко Ю.І., Ганзя Р.С., Акользіна О.С. Електронні підписи на основі ідентифікаторів та бінарного відображення 284

Акользіна О.С., Горбенко І.Д. Аналіз захищеності алгоритму цифрового підпису ЕС KCDSA від атак на зв'язаних ключах 291

Кулібаба В.А., Тоцький О.С., Горбенко І.Д. Аналіз захищеності алгоритмів електронних підписів ГОСТ Р 34.10 2012 та ДСТУ 4145 2002 від атаки на зв'язаних ключах 296

Єсіна М.В. Математична модель протоколу сліпого електронного підпису на еліптичних кривих 300

Ковальова Н.В., Горбенко Ю.І., Бакликов О.О. Аналіз електронних підписів з відновленням повідомлення згідно з ДСТУ ISO/IEC 9796-2 306

МЕТОДЫ И СРЕДСТВА СИММЕТРИЧНЫХ КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

Rodinko M.Yu., Oliynykov R.V., Hrinenko T.O. Improvement of the method for optimal S-boxes generation 315

Кузнецов О.О., Іваненко Д.В., Мордвінов Р.І. Дослідження потокових алгоритмів шифрування – переможців міжнародного проекту ESTREAM 321

Лисицкий К.Е. О методике оценки законов распределения вероятностей максимумов полных дифференциалов и смещений линейных оболочек блочных симметричных шифров 335

Лавровская Т.В., Вотяков О.И., Рассомахин С.Г. Оценка эффективности псевдослучайных кодов, сгенерированных с помощью LFSR 339

Кузнецов А.А., Белозерцев И.Н., Андрушкевич А.В. Анализ и сравнительные исследования нелинейных узлов замены современных блочных симметричных шифров 343

Кузнецов А.А., Горбенко Ю.И., Иваненко Д.В. Комбинаторные свойства ключевого расписания блочных симметричных шифров 351

Котух Е.В., Халимов Г.З. Универсальное хеширование по кривым, ассоциированным с группой Судзуки 361

МЕТОДЫ И СРЕДСТВА ЭЛЕКТРОННОЙ ИДЕНТИФИКАЦИИ

Потій О.В., Козлов Ю.Н., Дьяченко І.І. Базові принципи електронної ідентифікації. Концепція інфраструктури електронної ідентифікації України 366

Потій О.В., Брошеван Є.В. Методи формування унікального електронного ідентифікатора і процес видачі електронних посвідчень: досвід Євросоюзу 370

Потій О.В., Піддубний К.О. Протоколи надання довірчих послуг у національних архітектурах електронної ідентифікації країн Євросоюзу 377

Кузнєцов О.О., Тимченко В.А., Дубровний Б.С. Аналіз та порівняльні дослідження методів мережної стеганографії 384

Пономар В.А. Побудова моделі порушника для загрози підміни чи модифікації апаратного засобу КЗІ та аналіз способів протидії 390

Аулов І.Ф. Комплексний метод порівняння, оцінки та вибору механізмів управління ключовими даними користувачів у хмарних ІТС 397

К 70-летию со дня рождения Ивана Дмитриевича Горбенко 403

УВАЖАЕМЫЕ ЧИТАТЕЛИ!

Настоящий выпуск журнала «Прикладная радиоэлектроника» является тематическим и посвящен проблемным вопросам кибербезопасности, в основном в части криптографической защиты информации. Ряд статей, которые представлены в журнале, являются заказными, их тематика определяется практическими приложениями. В основном они подготовлены специалистами и молодыми соавторами по тематике, ориентируясь на задачи, которые решаются спонсором издания журнала – ПАТ «Институт информационных технологий», г. Харьков.

При отборе статей делалась ориентация авторов на оценку стойкости существующих криптографических преобразований для их применения в постквантовый период, а также на разработку предложений относительно совершенствования криптографических преобразований с учетом возникающих требований. Анализ и оценка, в первую очередь, были направлены на принятые в Украине, в качестве национальных, международные стандарты.

Дело в том, что в среде криптографов возникли волнения, связанные с наличием прогнозов относительно создания в перспективе квантового компьютера. Так, в случае появления квантового компьютера, на котором будут эффективно запущены квантовые алгоритмы криптоанализа Шора или алгоритм поиска Гроувера и др., могут возникнуть большие проблемы относительно стойкости как симметричных, так и ассиметричных криптографических преобразований.

Учитывая указанное, сегодня в технологически развитых государствах развернуты серьезные исследования и ведутся разработки в двух направлениях – создании квантового компьютера, а также разработки квантовых алгоритмов осуществления квантового криптоанализа.

Существенный импульс этим работам, по нашему мнению, придала публикация в Интернет статьи «A RIDDLE WRAPPED IN AN ENIGMA», авторы NEAL KOBLITZ AND ALFRED J. MENEZES. По мнению авторов, опубликованное агентством национальной безопасности правительства США (NSA) крупное политическое заявление о необходимости разработки стандартов для постквантовой криптографии, является мощным толчком для теоретических исследований и практических разработок в криптологии. В заявлении также указывается, что стандарты для новых постквантовых алгоритмов еще не изобретены.

Среди кандидатов на постквантовую криптографию авторами статьи называются:

- симметрические криптографические преобразования (блочные и поточные);
- криптопреобразования, основанные на алгебраических решетках;
- преобразования, основанные на хеш-функциях;
- криптография, основанная на кодах исправления ошибок;
- криптография на основе трудности вычисления изогения определенной степени между двумя изогенными суперсингулярными эллиптическими кривыми над конечным полем.

Публикуемые в этом журнале статьи, конечно без особых претензий авторов, в определенной мере ориентированы на решение задач синтеза и анализа постквантовых преобразований.

Так, в первом разделе представлены статьи, посвященные ассиметричным криптопреобразованиям направленного шифрования и электронной подписи. Они, в определенной мере, по крайней мере для авторов, отвечают на вызовы относительно требований к постквантовой криптографии, а также позволяют сделать определенные шаги к ней.

Во втором разделе мы продолжаем публикации, посвященные алгоритмам симметричных блоковых преобразований. Мы считаем, что принятием в Украине национальных стандартов ДСТУ 7624:2014 и ДСТУ 7564:2014, уже создана постквантовая криптография в части симметричных блочных преобразований и функций хеширования.

В третьем разделе представлены в основном статьи, посвященные электронной идентификации и и аутентификации.

С уважением и пожеланиями успехов,
профессор ***И.Д. Горбенко***

МЕТОДЫ И СРЕДСТВА АССИМЕТРИЧЕСКИХ КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

UDK 004 056 55

FEATURES OF PARAMETERS CALCULATION FOR NTRU ALGORITHM

IVAN GORBENKO, OLENA KACHKO, KOSTYANTYN POGREBNYAK

The paper considers the practical aspects of parameters generation for the NTRU algorithm (ANSI X9.98). The obtained results will allow to generate new sets of parameters to meet the growing requirements for cryptographic security and the availability of new attacks on the lattice.

Keywords: NTRU parameters, cryptographic complexity, error decryption, combined attack.

INTRODUCTION

The NTRU algorithm is used for encryption and digital signatures, and has much better speed characteristics than the rest of the algorithms of this class [3, 9]. In spite of problems of using this algorithm for digital signature, its application to asymmetric encryption is based on standard [1] (the Standard). The aim of this work is the analysis of NTRU encryption algorithm parameters proposed in [1]. It is this choice of parameters that ensures the reliability of the algorithm in terms of repayment calculation operations of a private key, encryption and decryption and the required cryptographic security.

The fullest justification for the choice of parameters is considered in [2]. It is the parameters, which are proposed in this paper, are used in [1]. We have considered the formation of some parameters [2] for the purpose of their inspection and provision of the generation of other parameters which are not inferior to the proposed ones.

The parameters which are used in the NTRU encryption algorithm are as follows:

N is the polynomial degree that determines its the maximum exponent which ensures multiplicative group maximum order [5];

q is a "big" module, a positive integer number for which one can easily calculate the module value ($q = 2048$ for the Standard);

p is a "small" module, usually a small positive integer number or a polynomial with small coefficients of small degree, p and q are relatively prime ($p = 3$ for the Standard).

The designations d_f, d_g are used to set the amount of 1 and –1 in polynomials. These values are used to generate a private key; they actually determine the power of a set of private keys, $d_g = \left\lfloor \frac{N}{3} \right\rfloor$, $d_f < \frac{N}{3}$ for the Standard;

The designation d_r is used to set the amount of 1 and –1 in the blinding polynomial. It is applied for encrypting ($d_r = d_f$ for the Standard).

The remaining parameters depending on N, d_f and security level are not considered in the paper.

Thus, it is sufficient to choose the following parameters: N, d_f .

The reference [2] proposes the following criteria for choosing parameters:

– ensuring of the required security level;

– ensuring of minimal computational complexity;

– minimum memory requirement for storing key data and encrypted text.

According to the criteria listed three classes of parameters are defined.

1. Using minimal storage for keys and the encrypted block (N*11 bits) – size metric is achieved due to the minimum order of the polynomial for ensuring a given security level. In [1, 2] this class includes polynomials ((401, 113) (449.134) (677, 157) (1087, 120)). The first digit of the pair specifies the value of N, and the second one gives the value of d_f .

2. By reducing the number of non-zero elements for private keys one decreases the time required to perform a multiplication of polynomials (computational complexity is equal to $N * d_f$) – performance metric. In [1, 2] this class includes polynomials of (659, 38), (761, 42), (1087, 63), (1499, 79)) orders.

3. The combination of size and performance metrics makes it possible to obtain parameters that at small dimensions of additional memory enable to get a significant speed boost. In [1, 2] this class includes polynomials of ((541, 49), (613, 55), (887, 81), (1171, 106) orders.

The paper deals with the generation parameters that are determined by all the metrics.

CHOOSING PARAMETER *N*

This parameter considerably affects the resistance of the cryptographic algorithm. Since the very beginning of using this parameter it has been recommended to choose N as a prime number, which increases the probability of the existence of an inverse element [6] for the private key. Furthermore, [7] shows the availability of additional attacks if N is not a prime number. The probability of having an inverse element and effectiveness of attacks, based on the factorization of parameter q in module $X^N - 1$, depends on the group order n of number N. The degree of polynomial N is chosen if the appropriate group order of $n \geq \frac{N-1}{2}$.

The value of the group order for N (401, 449, 761, 887, 1087) is equal to $\frac{N-1}{2}$. The value of the group order for N (541, 613, 659, 677, 1171, 1499) is equal to N–1. Thus, the condition for the group order is performed for all values of N. 169 primes are in the range [401, 1499], of which 108 prime numbers satisfy the demand for the group order. Thus, in addition to the Standard preset one can choose other primes.

CHOICE OF PARAMETERS IN VIEW OF DECODING ERRORS

In order to determine the conditions of decoding error appearance it is enough to consider the decryption algorithm. The ambiguity appears when the coefficients of the polynomial exceed $q/2$. If one foresees the normal low of distribution of the coefficients values, the probability of exceeding a given value of c for at least one polynomial coefficient is determined according to the formula (1) with due account of $d_r = d_f$ [2]:

$$p_{err} = N * erfc\left(\frac{c}{\sigma\sqrt{2}}\right), \quad (1)$$

where $c = \frac{q-2}{2p}, \sigma = \sqrt{\frac{8d_f}{3}}$.

To provide a given cryptographic security level, this probability shouldn't exceed 2^{-S}, where S, respectively, equals {112, 128, 192, 256}.

The maximum values d_f, which provide the required decoding error absence probability, are shown in Table 1. The values of d_f, that are used in the Standard, are given in parentheses.

Table 1

Maximum values of d_f to ensure correct decoding

S/N	401	449	677	1087	541	613
112	133 (113)				180 (49)	
128		149 (134)				204 (55)
192			161 (157)			
256				122 (120)		
S/N	887	1171	659	761	1087	1499
112			219 (38)			
128				236 (42)		
192	161 (81)				161 (63)	
256		120 (106)				120 (79)

Solving the problem of finding private keys by public ones depends on the method of forming the private key. In the Standard the private key is formed according to the formula: $f = pF + 1$. In this case, the problem of finding the private key refers to Closest Vector Problem (CVP) [9]. The most effective solution to the problem is the hybrid attack [2].

To perform the hybrid attack the whole lattice is divided into 2 parts.

The first part includes y_2 first rows of the lattice ($y_2 = [N, 2N]$). The reduction operation is performed for it.

For the second part of the lattice the number of rows is equal to $2N - y_2$. The attack Meet-In-The-Middle (MITM) is performed for it [8]. The value of c – numbers 1 (–1) remaining in the second part of the lattice, is determined. It is assumed that the numbers 1 (–1) are identical. This is what provides the worst conditions for the attack.

Recovery is performed for the lattice, the number of rows is determined by values: $[y_1, y_2]$, where

$$y_1 = [0, N-1], y_2 = [N, 2N-1],$$

$$y_1 = \frac{2N - y_2(1+\alpha)}{1-\alpha}.$$

Obviously, the quality of recovery depends on the number of rows for which the reduction is performed, with increasing their number, the quality of recovery increases. To increase their number requires increasing the value of $\alpha = (0,1)$.

According to [2], to assess the time required to recover the specified number of lattice rows uses the equation:

$$w = \frac{2m(y_2 - N)}{(1-\alpha)^2} + 3\ln\frac{2(y_2 - N)}{(1-\alpha)} + c, \quad (2)$$

where $m = 0.2$, $c = -50$ constants determined experimentally; y_2 determines the number of the last lattice row, for which the reduction is realized; α is the factor that determines the number of rows that are takes into account for recovery; w determines the time needed for such an attack. Actually, the time is equal to 2^w.

The value of α can be found for a given cryptographic security level, solving equation (2) with respect to α. The maximum value of α can be found using a formula that determines the value y_1 by solving the imparity: $\frac{2N - y_2(1+\alpha)}{1-\alpha} \geq 0$.

So, $\alpha_{\max} = \frac{2N - y_2}{y_2}$. Due to the fact that the attack on the lattice is performed in parallel with a MITM attack the value of α may be less than it is necessary to obtain the current security level. In this case, the time required for the MITM attack, should not be less than it is necessary to ensure the security level. The total time required for a hybrid attack, is determined by the maximum time. Fig. 1 shows a graph of the dependence of the value α on y_2 ($\alpha_{\max}$).

Besides, a graph of values α is specified under which the required security level is achieved (labeled AlphaMin) to attack on the lattice. As shown in Fig. 1, the relative value range α (Delta graph) increases, as the value of y_2 increases. The graphs for different values of N are of similar character.

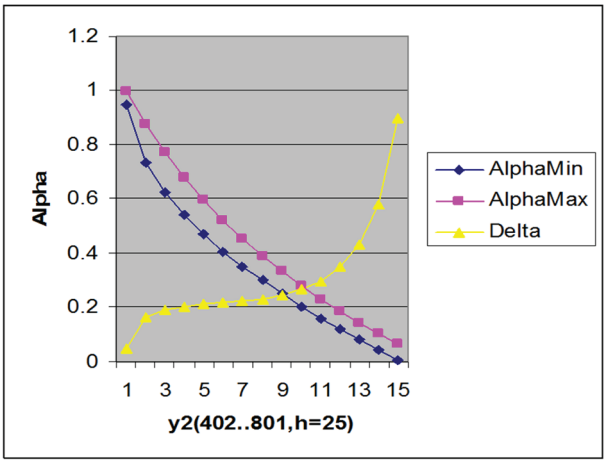


Figure 1. Graf of the dependencies of the α on y_2 ($N = 401$)

Figure 2 shows a graph of w-α, relationship obtained at the following values (N = 401, y_2 =693, α = [0.0668–0.1573]).

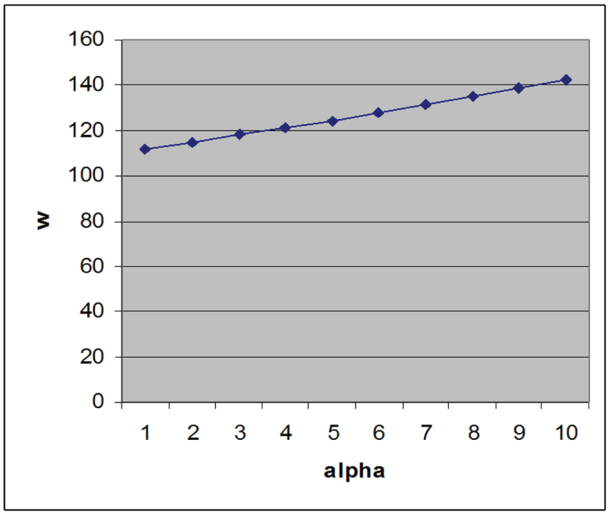


Fig. 2. Graph of w (α)

The initial value α is selected to get $w = 112$. The graph shows that the increase α leads practically to the linear increase w. Changing the value of α allows to balance lattice reduction time and recovery time of the rest of the private key.

As it can be seen from (2), the maximum value of α corresponds to the minimum y_2. With increasing y_2 it is necessary to reduce α to ensure the necessary security level.

Table 2 shows the calculated values of α for y_2 compared to the anticipated ones according to [2].

The values are determined only for those prime numbers that comply with the security level. As can be seen from Table 2, calculated and anticipated values practically coincide. The optimum values of parameters y_2,α will be determined forth. Time is calculated for this which is required to perform a MITM attack, realized for $2N - y_2$ rows of the matrix.

To calculate the parameters with respect to a MITM attack, one should consider the following:

– the numbers 1(–1) in the remaining rows ($c = 2N - y_2$) must be correct, i.e. with regard to their total (d_f). The author [2] designates the probability of this fact as p_{Split};

– the result obtained by means of the MITM attack must be consistent with the result that has been obtained with the help of the previous attack after finding the nearest lattice vector, i.e. the total number of 1 and –1 must be correct. The probability of this fact is denoted as p_s [2];

Table 2

Minimum values of α for specified security level

S	N	y_2	Anticipated values α	Obtained values α
112	401	693	0.095	0.0949443
	541	800	0.149	0.148151
	659	902	0.175	0.175155
128	449	770	0.100	0.0993571
	613	905	0.142	0.141385
	761	1026	0.183	0.182416
192	677	1129	0.096	0.0960767
	887	1294	0.143	0.142555
	1087	1464	0.175	0.174973
256	1087	1630	0.127	0.126437
	1171	1693	0.144	0.143584
	1499	1984	0.174	0.174653

The total time required to perform the attack is defined by the formula [2]:

$$w_2 = \frac{t\Pi}{p_{Split}}, \qquad (3)$$

where: t is the time required to perform one iteration; Π is the number of iterations.

Calculation of individual components of the formula (3) will be further examined.

p_{Split} CALCULATION

The designation $\begin{pmatrix} n \\ r_1 \quad r_2 \end{pmatrix}$, which means the number of variations that can be placed in n positions of r_1 data of one type (i.e. +1) and r_2 data of other type, for example (-1) will be used in the subsequent formulas [2]. All definitions below assume that $r_1 = r_2$, that is, the designation $\begin{pmatrix} n \\ r \quad r \end{pmatrix}$ will be actually used. The corresponding number of variations is calculated using the formula $C_n^r * C_{n-r}^r = \frac{n!}{(r!)^2 (n-2r)!}$.

p_{Split} is determined by the formula:

$$p_{Split} = \begin{pmatrix} y_2 - N \\ d_f - c \quad d_f - c \end{pmatrix} * \begin{pmatrix} 2N - y_2 \\ c \quad c \end{pmatrix} * \begin{pmatrix} N \\ d_f \quad d_f \end{pmatrix}^{-1}. \qquad (4)$$

The expression between the first brackets determines the number of options that can accommodate $d_f - c$ (1) and (–1) in the first $y_2 - N$ rows. The expression between the second brackets determines the number of options that can accommodate c (1) and

(–1) in the $2N - y_2$ rows. The expression between the third brackets determines the total number of options that can accommodate all (1) and (–1) (d_f) in N rows.

With the possibility of further use of the key rotation the formula to calculate p_{Split} take the form: $p_{Split} = \left(1 - p_{Split}\right)^N$.

Let us specify the limits at which the formula (4) makes sense considering restrictions on y_2, c: $y_2 \in [N, 2N-1]$, $c \in [0, d_f]$:

$$c_{\min} = \begin{cases} d_f - 0.5(y_2 - N) & 0.5(y_2 - N) < d_f \\ 0 & 0.5(y_2 - N) < d_f \end{cases};$$

$$c_{\max} = \begin{cases} 0.5(2N - y_2) & 0.5(2N - y_2) < \ d_f \\ d_f & 0.5(2N - y_2) \geq d_f \end{cases}.$$

The values of all components of (4) must be integer.

Table 3 shows set values of N, d_f, y_2, c and corresponding value of p_{Split} calculated by the formula (4) for the security levels 112, 128, 192 and 256 respectively. Besides, Table 3 shows the anticipated values according to [2]. The values obtained in the calculation and anticipated ones practically coincide.

The influence of the value c on the p_{Split} is considered using an example of calculations for $S = 112$, $N = 541$, $d_f = 49$, $y_2 = 800$. In this case, $c_{\min} = 0, c_{\max} = 49$. The calculation results have

Table 3

Calculated and anticipated values of p_{Split}

Sec	N	y_2	c	$\log_2 p_{Split}$ (anticipated)	$\log_2 p_{Split}$ (calculated)
112	401	693	27	–0.6	–0.58093
	541	800	15	–13.1	–13.0804
	659	902	13	–17.7	–17.6503
128	449	770	35	–0.3	–0.299504
	613	905	17	–14.9	–14.9203
	761	1026	15	–20.9	–20.8762
192	677	1129	45	–2.0	–2.03799
	887	1294	27	–21.9	–21.8938
	1087	1464	23	–31.9	–31.9323
256	1087	1630	39	–24.9	–24.9227
	1171	1693	37	–28.7	–28.7098
	1499	1984	29	–47.8	–47.8223

shown that at the beginning the probability increases monotonically, has a maximum in the set interval and then decreases monotonically.

p_s CALCULATION

In fact, the probability p_s determines that the norm for the calculation result obtained by the reduction (key G and part of key F without c (1) and (–1)) does not exceed the anticipated norm $2d_g + 2(d_f - c)$. At that it is assumed that the values distribution law for key G and key F is normal, in this case the error function can be used for the calculation.

To calculate this probability according to [2], one can use the formula:

$$p_s = \left(1 - \frac{2}{3q}\right)^{y_1} * \Pi_{i=0}^{y_2 - y_1} \left(1 - \overline{f_{D,\sigma}}\right), \quad (5)$$

where

$$y_1 = \frac{2N - y_2(1+\alpha)}{1-\alpha};$$

$$D = q^{\frac{\alpha(y_2 - y_1) + i(1-\alpha)}{y_2 - y_1}};$$

$$\overline{f_{D,\sigma}} = erfc\left(\frac{D}{\sigma\sqrt{2}}\right) - \frac{\sigma\sqrt{2}}{D\sqrt{\pi}}\left(e^{\frac{D^2}{2\sigma^2}} - 1\right);$$

$$\sigma = \sqrt{\frac{2d_g + 2(N-c)}{y_2}}.$$

The value of σ determines the part of non-zero elements among y_2 elements.

From (5) we get:

$$\log_2 p_s = y_1 \log_2\left(1 - \frac{2}{3q}\right) + \sum_{i=0}^{y_2 - y_1} \log_2\left(1 - \overline{f_{D,\sigma}}\right). \quad (6)$$

The results of the calculation according to (6) compared with the values obtained in [2] are presented in Table 4.

Table 4

Probability of selected key correctness according to norm

N	y_2	d_f	C	$\log_2 p_s$ (anticipated)	$\log_2 p_s$ (calculated)
401	693	113	27	–45.4	–45.4676
541	800	49	15	–26.9	–26.8954
659	902	38	13	–21.9	–22.0056
449	770	134	35	–49	–48.9491
613	905	55	17	–31.5	–31.5344
761	1026	42	15	–23.1	–23.0858
677	1129	157	45	–67.4	–67.6896
887	1294	81	27	–43.9	–43.851
1087	1464	63	23	–34.2	–34.3427
1087	1630	120	39	–64.1	–64.2834
1171	1693	106	37	–56	–55.1001
1499	1984	79	29	–44.4	–44.6091

As the results of the calculation have shown, the results obtained are close to these obtained in [2].

This paper presents the results of the generation of the main components that are necessary to determine the parameter d_f for the selected value N. The obtained results agree with those given in [2]. This will allow to generate and explore the parameters for the NTRU algorithm.

References

[1] American National Standard for Financial Services ANSI X9.98 – 2010. Lattice-Based Polynomial Public Key Establishment Algorithm for the Financial Services Industry.

[2] *P. Hirschhorn, J. Hoffstein, N. Howgrave-Graham3, W. Whyte.* Choosing NTRUEncrypt Parameters in Light of Combined Lattice Reduction and MITM Approaches, https://www.securityinnovation.com/uploads/Crypto/params.pdf / (visited 22.12.2015).

[3] *Горбенко И.Д., Качко Е.Г., Балагура Д.С.* Применение криптосистем NTRU в инфраструктуре от-

крытых ключей. Міжнародна наукова конференція «Питання оптимізації обчислень (ПОО-XL)», Україна, Крим, Велика Ялта, смт. Кацивелі, 30.09.2013–04.10.2013.

[4] *Качко Е.Г., Балагура Д.С., Горбенко Ю.И.* Обоснование и исследование практической реализации улучшенного алгоритма цифровой подписи NTRUSIGN/ Прикладная радиоэлектроника, т.11, № 2, 2012 г. , с. 195-200.

[5] NTRU Cryptosystems Technical Report. http://www.securityinnovation.com/uploads/Crypto/NTRUTech014.pdf / (visited 22.12.2015).

[6] NTRU Cryptosystems Technical Report. Report #009, Version 1. Invertibility in Truncated Polynomial Rings https://www.securityinnovation.com/uploads/Crypto/NTRUTech009.pdf / (visited 22.12.2015).

[7] *Craig Gentry*. Key Recovery and Message Attacks on NTRU-Composite http://www.iacr.org/archive/eurocrypt2001/20450181.pdf / (visited 22.12.2015).

[8] *Nick Howgrave-Graham*. A Hybrid Lattice-Reduction and Meet-in-the-Middle Attack Against NTRU. https://www.securityinnovation.com/uploads/Crypto/hybrid.pdf / (visited 22.12.2015).

[9] *Jeff Hoffstein, Nick Howgrave-Graham, Jill Pipher, William Whyte*. Practical lattice-based cryptography: NTRUEncrypt and NTRUSign. https://www.securityinnovation.com/uploads/Crypto/lll25.pdf/(visited 22.12.2015).

Manuscript received October, 30, 2015

Gorbenko Ivan, professor, doctor of technical sciences, State Prize Laureate (science and technology), chief designer of the private corporation «Institute of Information Technologies». Research interests: cryptology and cyber security.

Kachko Olena, Ph.D., assistant professor of the Department of Software Engineering of KhNURE. Research interests: cryptography, cryptanalysis, parallel computing.

Pogrebnyak Kostyantyn, Ph.D., assistant professor of the Department of Information Technology Security of KhNURE. Research interests: application of methods of algebraic geometry in cryptology, asymmetric cryptanalysis.

УДК 004 056 55

Особливості обчислення параметрів для алгоритму NTRU / І.Д. Горбенко, О.Г. Качко, К.А. Погребняк // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 272–276.

Розглянуто практичні аспекти генерації параметрів для алгоритму NTRU (ANSI X9.98). Отримані результати дозволять генерувати нові набори параметрів з урахуванням зростаючих вимог до криптографічної складності та наявності нових атак на решітку.

Ключові слова: параметри NTRU, криптографічна складність, помилка розшифрування, комбінована атака.

Табл.: 4. Іл.: 2. Бібліогр.: 9 найм.

УДК 004 056 55

Особенности вычисления параметров для алгоритма NTRU / И.Д. Горбенко, Е.Г. Качко, К.А. Погребняк // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 272–276.

Рассмотрены практические аспекты генерации параметров для алгоритма NTRU (ANSI X9.98). Полученные результаты позволят генерировать новые наборы параметров с учетом растущих требований к криптографической сложности и наличия новых атак на решетку.

Ключевые слова: параметры NTRU, криптографическая сложность, ошибка дешифрования, комбинированная атака.

Табл.: 4. Ил.: 2. Библиогр.: 9 назв.

УДК 681.3.06

КЛАССИФИКАЦИЯ КРИВЫХ В ФОРМЕ ЭДВАРДСА НАД ПРОСТЫМ ПОЛЕМ

А.В. БЕССАЛОВ, О.В. ЦЫГАНКОВА

Дан анализ свойств точек малых порядков скрученной кривой Эдвардса. Введена арифметика для групповых операций с особыми точками этих кривых. Предложена классификация кривых в форме Эдвардса на 3 непересекающихся класса. Получены точные формулы для числа кривых разных классов минимального порядка. Дан критический анализ результатов в работах других авторов.

Ключевые слова: эллиптическая кривая, скрученная кривая Эдвардса, параметры кривой, порядок точки, сложение точек, изоморфизм, квадратичное кручение, квадрат, неквадрат.

ВВЕДЕНИЕ

Эллиптические кривые в форме Эдвардса над простым полем, без сомнения, весьма перспективны в современной криптографии. Как мы доказали в работе [3], их производительность в среднем не менее чем в 1.5 раза превышает производительность кривых в форме Вейерштрасса. При троичном NAF(k) представлении числа k точки kP выигрыш в скорости вычислений достигает 1.6 раза. Арифметика этих кривых проще в связи с наличием нейтрального элемента группы как неособой точки кривой (1,0). Исключая бесполезные изоморфные кривые, в кривых Эдвардса достаточно использовать один параметр d вместо обычных двух параметров a и b кривой в канонической форме Вейерштрасса.

В работе [2] авторы обобщили и расширили класс кривых Эдвардса [1] введением нового параметра a и снятием ограничения на неквадратичность параметра d кривой. Они назвали этот класс скрученными кривыми Эдвардса (Twisted Edwards Curves), а кривые, определенные в [1] – полными кривыми Эдвардса. В работе [2] был дан детальный анализ некоторых свойств этих кривых, доказан ряд важных теорем, сделана попытка дать классификацию кривых в форме Эдвардса и привести некоторую статистику распределения порядков кривых, относящихся к разным классам этих кривых при небольших значениях модуля $p = 1009$ и $p = 1019$. Мы обнаружили, что кривые в форме Эдвардса разбиты в этой работе на пересекающиеся классы, в результате чего в статистических таблицах раздела 4 одни и те же кривые попадают в разные классы, что дает недостоверную статистику.

В данной работе мы даем критический анализ свойств скрученных кривых Эдвардса. В первом разделе мы предлагаем арифметику для групповых операций с особыми точками этих кривых, даем анализ точек малых порядков и формулы, связывающие их с другими точками кривой. Во втором разделе мы обсуждаем и обосновываем некорректность некоторых утверждений и классификации кривых в [2], предлагаем классификацию кривых в форме Эдвардса с разбиением на три непересекающиеся класса в зависимости от свойств параметров a и d кривой. Мы анализируем свойства кривых всех трех классов и возможные значения порядков этих кривых. Наконец, в третьем разделе нами получены точные формулы для числа кривых различных классов с минимальным кофактором 4 и дан критический анализ результатов, приведенных в [2].

1. ТОЧКИ МАЛЫХ ПОРЯДКОВ СКРУЧЕННЫХ КРИВЫХ ЭДВАРДСА

В работе [2] *скрученные кривые Эдвардса* (twisted Edwards curves) определены как обобщение кривых Эдвардса $x^2 + y^2 = 1 + dx^2y^2$ путем ввода нового параметра a в уравнение

$$\mathrm{E}_{\mathrm{E},a,d}: ax^2 + y^2 = 1 + dx^2y^2,$$

$$a \neq d,\ a, d \in \mathrm{F}_p^*,\ d \neq 1,\ p \neq 2.$$

Наряду с вводом параметра a авторы [1, 2] сняли ограничения на пару параметров a и d, допуская любые значения $\left(\frac{ad}{p}\right) = \pm 1$. При $a = 1$ получаем *кривую Эдвардса*, а если у нее d – неквадрат или $\left(\frac{d}{p}\right) = -1$, то в [2] она названа *полной кривой Эдвардса*. Этот термин связан с полнотой закона сложения точек кривой [1]. В работе [4] мы предложили поменять местами x и y координаты в форме кривой Эдвардса с целью сохранения горизонтальной симметрии обратных точек, принятой в криптографии. Опираясь на это свойство, определим скрученную кривую Эдвардса уравнением

$$\mathrm{E}_{\mathrm{E},a,d}: x^2 + ay^2 = 1 + dx^2y^2,$$

$$a,d \in \mathrm{F}_p^*,\ d(d-a) \neq 0,\ p \neq 2. \qquad (1)$$

Тогда модифицированный универсальный закон сложения точек имеет вид

$$(x_1, y_1) + (x_2, y_2) = \left(\frac{x_1x_2 - ay_1y_2}{(1 - dx_1x_2y_1y_2)}, \frac{x_1y_2 + x_2y_1}{(1 + dx_1x_2y_1y_2)} \right). \qquad (2)$$

При совпадении двух точек получим из (2) закон удвоения

$$2(x_1, y_1) = \left(\frac{x_1^2 - ay_1^2}{(1 - dx_1^2y_1^2)}, \frac{2x_1y_1}{(1 + dx_1^2y_1^2)} \right). \qquad (3)$$

Использование модифицированных законов (2), (3) позволяет сохранить горизонтальную симметрию (относительно оси x) обратных точек, общепринятую в теории эллиптических кривых. Определяя теперь обратную точку как $-P = (x_1, -y_1)$, получаем согласно (1) $(x_1, y_1) + (x_1, -y_1) = \mathbf{O} = (1, 0)$. На оси x также всегда лежит точка $\boldsymbol{D_0} = (-1, 0)$ второго порядка, для которой в соответствии с (3) $2\boldsymbol{D_0} = (1, 0) = \mathbf{O}$. В зависимости от свойств параметров a и d можно получить еще 2 особые точки второго порядка и 2 или 4 точки 4-го порядка. Как следует из (1), на оси y могут лежать точки $\pm\boldsymbol{F_0} = (0, \pm 1/\sqrt{a})$ 4-го порядка, для которых $\pm 2\boldsymbol{F_0} = \boldsymbol{D_0} = (-1, 0)$. Ясно, что эти точки существуют над полем F_p, если параметр a является квадратом.

Из уравнения (1) определим квадраты

$$x^2 = \frac{1-ay^2}{1-dy^2},\ \ y^2 = \frac{1-x^2}{a-dx^2}, \qquad (4)$$

порождающие в ряде случаев особые точки на бесконечности (знак «∞»мы ставим при делении на 0):

$$\boldsymbol{D_{1,2}} = \left(\pm\sqrt{\frac{a}{d}}, \infty\right),\ \ \pm\boldsymbol{F_1} = \left(\infty, \pm\frac{1}{\sqrt{d}}\right). \qquad (5)$$

Они возникают в случаях $\left(\frac{ad}{p}\right) = 1$ и $\left(\frac{d}{p}\right) = 1$ соответственно. По правилам обычного предельного перехода и закона (3) легко проверить, что $2\boldsymbol{D_{1,2}} = \mathbf{O}$, $\pm 2\boldsymbol{F_1} = \boldsymbol{D_0} = (-1, 0)$. Иными словами, при выполнении условий их существования особые точки $\boldsymbol{D_{1,2}}$ есть точки 2-го порядка, а особые точки $\pm\boldsymbol{F_1}$ – точки 4-го порядка. Нейтральный элемент $\mathbf{O}$ и точки 2-го и 4-го порядков кривой Эдвардса здесь и далее выделяются жирным шрифтом.

Кроме перечисленных, точки 4-го порядка могут существовать как неособые при ненулевых координатах x и y. Положим, например, $2\boldsymbol{F_2} = 2(x_1, y_1) = \boldsymbol{D_1}$. Тогда согласно (2) и (5) запишем два уравнения

$$\frac{x_1^2 - ay_1^2}{\left(1 - dx_1^2y_1^2\right)} = \sqrt{\frac{a}{d}},\ \ \frac{2x_1y_1}{\left(1 + dx_1^2y_1^2\right)} = \infty.$$

Отсюда

$$\left(1 + dx_1^2y_1^2\right) = 0,\ \ \Rightarrow\ x_1^2 + ay_1^2 = 0, \Rightarrow\ \ x_1^2 = -ay_1^2.$$

Согласно первому из записанных выше уравнений имеем

$$\frac{2x_1^2}{1 + \frac{d}{a}x_1^4} = \sqrt{\frac{a}{d}} \Rightarrow \frac{d}{a}x_1^4 - 2\sqrt{\frac{d}{a}}x_1^2 + 1 = 0 \Rightarrow$$

$$\Rightarrow x_1^2 = \sqrt{\frac{a}{d}},\ \ y_1^2 = -\frac{1}{\sqrt{ad}}.$$

Итак, координаты возможных точек 4-го порядка $\pm\boldsymbol{F_2} = \left(\sqrt[4]{\frac{a}{d}}, \pm\sqrt{\frac{-1}{\sqrt{ad}}}\right)$. $\pm\boldsymbol{F_3} = \left(-\sqrt[4]{\frac{a}{d}}, \pm\sqrt{\frac{-1}{\sqrt{ad}}}\right)$. Необходимыми и достаточными условиями существования таких точек 4-го порядка являются:

$$\left(\frac{ad}{p}\right) = 1,\ \ p \equiv 3 \bmod 4. \qquad (6)$$

Для этого случая множества всех квадратов и всех 4-х степеней элементов совпадают, т.е. если $\left(\frac{a}{p}\right) = 1$, то $a^2 \neq a$ для всех $a \neq 1$. При этом любой из квадратов имеет 2 квадратных корня и 2 корня 4-й степени. Действительно, если α – примитивный элемент мультипликативной группы F_p^*, и α^2 – квадрат этой группы, то $\alpha^2\alpha^{p-1} = \alpha^{2+4k+2} = \alpha^{4(k+1)}$. Значит, любой квадрат имеет и корни 4-й степени.

Например, для кривой $x^2 + 6y^2 = (1 + 3x^2y^2) \bmod 7$ (здесь $a = -1$ и $d = 3$ – неквадраты при $p = 7$) точки 4-го порядка имеют координаты $\boldsymbol{F_{2,3}} = (\pm 2, \pm 2)$. При проверке согласно (3) получим $2\boldsymbol{F_2} = \left(\sqrt{\frac{a}{d}}, \infty\right) = \boldsymbol{D_1}$. Порядок N_E этой кривой, включающей точки $\mathbf{O}$, $\boldsymbol{D_{0,1,2}}$, $\pm\boldsymbol{F_{2,3}}$, равен 8, группа точек нециклическая с типом $T = (2, 2^2)$.

Найдем условия существования точек 8-го порядка. Пусть $S = (x_1, y_1)$ – точка 8-го порядка, тогда $2S_1 = \boldsymbol{F_0} = (0, 1/\sqrt{a})$ – точка 4-го порядка. Согласно (2)

$$\frac{x_1^2 - ay_1^2}{\left(1 - dx_1^2y_1^2\right)} = 0, \qquad \frac{2x_1y_1}{\left(1 + dx_1^2y_1^2\right)} = \frac{1}{\sqrt{a}}.$$

Тогда

$$x_1^2 = ay_1^2 \Rightarrow \frac{d}{a}x_1^4 - 2x_1^2 + 1 = 0 \Rightarrow x_{1,2}^2 = \frac{a}{d}\left(1 \pm \sqrt{1 - \frac{d}{a}}\right).$$

Так как справедливо

$$\left(1 + \sqrt{1 - \frac{d}{a}}\right)\left(1 - \sqrt{1 - \frac{d}{a}}\right) = \frac{d}{a},$$

то возникают следующие необходимые и достаточные условия существования точек 8-го порядка:

1. При $\left(\frac{ad}{p}\right) = -1$: $\left(\frac{a}{p}\right) = 1$ и $\left(\frac{a-d}{p}\right) = 1$;
2. При $\left(\frac{ad}{p}\right) = 1$: $\left(\frac{a}{p}\right) = 1$,

$$\left(\frac{1 - \frac{d}{a}}{p}\right) = 1 \text{ и } \left(\frac{1 + \sqrt{1 - \frac{d}{a}}}{p}\right) = 1.$$

Например, в приведенном выше примере кривой с $a = -1$ и $d = 3$ при $p = 7$ оба параметра – неквадраты и нарушаются условия $\left(\frac{a}{p}\right) = 1$ и $\left(\frac{a-d}{p}\right) = 1$. Хотя порядок кривой равен 8, точек 8-го порядка она не содержит, т. к. группа нециклическая.

При условии существования особых точек (5) вместе с точками $\boldsymbol{D_0}$, $\pm\boldsymbol{F_0} = (0, \pm 1/\sqrt{a})$, принимая правила предельного перехода в (2), можно найти координаты сумм:

$$(x_1, y_1) + (-1, 0) = (-x_1, -y_1),$$

$$(x_1,y_1)+\left(\sqrt{\frac{a}{d}},\infty\right)=\left(\sqrt{\frac{a}{d}}x_1^{-1},\frac{1}{\sqrt{ad}}y_1^{-1}\right),$$

$$(x_1,y_1)+\left(-\sqrt{\frac{a}{d}},\infty\right)=\left(-\sqrt{\frac{a}{d}}x_1^{-1},-\frac{1}{\sqrt{ad}}y_1^{-1}\right),$$

$$(x_1,y_1)+\left(\infty,\frac{1}{\sqrt{d}}\right)=\left(-\frac{1}{\sqrt{d}}y_1^{-1},\frac{1}{\sqrt{d}}x_1^{-1}\right).,$$

$$(x_1,y_1)+\left(\infty,-\frac{1}{\sqrt{d}}\right)=\left(\frac{1}{\sqrt{d}}y_1^{-1},-\frac{1}{\sqrt{d}}x_1^{-1}\right).$$

Все найденные суммы удовлетворяют уравнению (1) при подстановке, т.е. являются точками кривой. Сумма $(x_1,y_1)+D_0=P^*=(-x_1,-y_1)$ меняет знаки координат точки P, тогда как сложение с особыми точками 2-го порядка инвертирует их с весами, сложение же с особыми точками 4-го порядка инвертирует с весами и меняет координаты местами.

Важно заметить, что использование правил предельного перехода сохраняет операцию сложения любых пар точек, включая особые. Это позволит говорить об изоморфизме кривых в различной форме.

2. КЛАССИФИКАЦИЯ КРИВЫХ В ФОРМЕ ЭДВАРДСА

В работе [2], как нам представляется, допущен ряд некорректных утверждений и результатов, которые мы выносим на обсуждение. Основные теоремы в этой работе опираются на бирациональную эквивалентность между кривыми (1) и кривыми в форме Монтгомери, заданными уравнением

$$\mathrm{E}_{\mathrm{M,A,B}}: Bv^2=u^3+Au^2+u,\ A=2\frac{a+d}{a-d},\ B=\frac{4}{a-d},$$

$$a=\frac{A+2}{B},\ d=\frac{A-2}{B},\ \mathrm{A}^2\neq 4. \qquad (7)$$

Она основана на замене координат с помощью рациональных функций

$$y=\frac{u}{v},\ x=\frac{u-1}{u+1}\ \ u=\frac{1+x}{1-x},\quad v=\frac{u}{x}. \qquad (8)$$

В работе [2] доказывается теорема 3.2: *любая скрученная кривая Эдвардса* (1) *бирационально эквивалентна кривой* (7) *в форме Монтгомери*.

Так как нам придется далее обращаться к паре квадратичного кручения (quadratic twist [2]), мы также проведем отображение точек (7) в точки кривой (1) .

Разделим (7) на v^2 и с учетом (8) получим

$$\frac{4}{(a-d)}\frac{1}{y^2}=u+u^{-1}+2\frac{a+d}{a-d},\Rightarrow$$

$$\Rightarrow\frac{2}{(a-d)}\frac{1}{y^2}=\frac{1+x^2}{1-x^2}+\frac{a+d}{a-d}.$$

Отсюда

$$\frac{2\left(1-x^2\right)}{y^2}=\left(1+x^2\right)(a-d)+\left(1-x^2\right)(a+d),$$

и, наконец, получаем изоморфную кривой (7) кривую в форме (1)

$$\mathrm{E}_{\mathrm{M,A,B}}\sim\mathrm{E}_{\mathrm{E},a,d}:\left(1-x^2\right)=y^2\left(a-dx^2\right). \qquad (9)$$

Нетрудно с помощью (8) осуществить и обратное преобразование. Имеет место взаимно однозначное отображение точек $(u_1,v_1)\leftrightarrow(x_1,y_1)$. Если для любой пары точек принять операцию сложения (2) с включением особых точек (см. раздел 1), то можно утверждать, что кривые $\mathrm{E}_{\mathrm{M,A,B}}$ и $\mathrm{E}_{\mathrm{E}a,d}$ изоморфны. Этот изоморфизм сохраняет порядки всех точек.

Перейдем теперь к парам квадратичного кручения. Пусть $\left(\frac{c}{p}\right)=-1$, тогда кривая кручения в форме Монтгомери имеет вид

$$\mathrm{E}^{\mathrm{t}}{}_{\mathrm{M,A,B}}\sim\mathrm{E}_{\mathrm{M,A,\,cB}}: cBv^2=u^3+Au^2+u,$$

$$A=2\frac{a+d}{a-d},\ B=\frac{4}{a-d}. \qquad (10)$$

Изоморфная ей скрученная кривая Эдвардса, как можно видеть из выполненных выше преобразований, записывается как

$$\mathrm{E}^{\mathrm{t}}{}_{\mathrm{E},a,d}\sim\mathrm{E}_{\mathrm{E},ca,cd}:$$

$$\left(1-x^2\right)=cy^2\left(a-dx^2\right)=y^2\left(ca-cdx^2\right). \qquad (11)$$

Иначе говоря, для построения пары квадратичного кручения к кривой в форме (1) необходимо перейти к новым параметрам кривой $a=ca, d=cd$, т.е. квадраты обращаются в неквадраты и обратно.

Во втором разделе пионерской работы [2] утверждается, что кривая $\mathrm{E}_{\mathrm{E},1,d/a}$ есть пара квадратичного кручения (quadratic twist) кривой $\mathrm{E}_{\mathrm{E},a,d}$, т.е. $\mathrm{E}^{\mathrm{t}}{}_{\mathrm{E},a,d}\sim\mathrm{E}_{\mathrm{E},1,\,d/a}$. Видимо, следует признать это утверждение в общем случае некорректным. Как следует из нашего анализа, оно справедливо лишь при $\left(\frac{a}{p}\right)=-1$, если принять $c=a^{-1}$. При $\left(\frac{a}{p}\right)=1$ кривые $\mathrm{E}_{\mathrm{E},a,d}$ и $\mathrm{E}_{\mathrm{E},1,d/a}$ изоморфны.

Чтобы классифицировать кривые в форме Эдвардса с разбиением на непересекающиеся классы, рассмотрим всевозможные сочетания для пар параметров a и d кривой (1).

1. $\left(\frac{ad}{p}\right)=-1$.

1.1. $\left(\frac{a}{p}\right)=1,\left(\frac{d}{p}\right)=-1$. Согласно (1) и (2) в этом случае на кривой (1) имеется единственная точка $\boldsymbol{D_0}=(-1,0)$ 2-го порядка и 2 точки 4-го порядка $\pm\boldsymbol{F_0}=(0,\pm1/\sqrt{a})$. В соответствии с (8) им отвечают точки кривой Монтгомери (7) $D_{\mathrm{M0}}=(0,0)$ и $\pm F_{\mathrm{M0}}=(1,\pm\sqrt{a})$. Этот случай определен в работе [1]. Здесь заменой $(x,y)\to(X,\ Y/\sqrt{a})$ получаем изоморфную кривой (1) полную кривую Эдвардса $X^2+Y^2=1+d'X^2Y^2$, $d'=d/a\Rightarrow\left(\frac{d'}{p}\right)=-1$. Итак, имеет место изоморфизм $\mathrm{E}_{\mathrm{E},a,d}\sim\mathrm{E}_{\mathrm{E},1,d/a}$.

1.2. $\left(\frac{a}{p}\right)=-1,\left(\frac{d}{p}\right)=1$. Здесь также нет осо-

бенностей, т. к. параметры a и d просто меняются местами. С помощью замены $(x,y) \to (1/X, Y)$ получим изоморфную кривую $X^2 + dY^2 = 1 + aX^2Y^2$. Ее квадратичное кручение образуется смещением параметров $d' = cd$, $a' = ca$, $\left(\frac{c}{p}\right) = -1$, при этом попадаем в условия п.1.1. Далее аналогично строим изоморфную кривую Эдвардса $\bar{x}^2 + \bar{y}^2 = 1 + \left(\frac{a}{d}\right)\bar{x}^2\bar{y}^2$. Таким образом, пара кривых $E^t_{E,1,d/a} \sim E_{E,1,a/d}$ образуют пару квадратичного кручения. Этот результат известен [1].

Итак, рассмотренные в п.1 условия для a и d порождают *полные кривые Эдвардса*.

2. $\left(\frac{ad}{p}\right) = 1$. Именно этот случай образует класс скрученных кривых Эдвардса и класс кривых Эдвардса с квадратичным параметром d. Как мы показали, квадратичное кручение образуется смещением параметров $d' = cd$, $a' = ca$, где c – неквадрат. Поэтому кривые в п.2.1. и 2.2. образуют пару квадратичного кручения. Свойства одной из кривых пары кручения полезны для определения свойств другой.

2.1. $\left(\frac{a}{p}\right) = -1, \left(\frac{d}{p}\right) = -1$. Согласно (7) имеем $(Bad)^2 = (A + 2)(A - 2)$ и, следовательно, дискриминант квадратного уравнения в правой части (7) $(A^2 - 4)$ является квадратом. Тогда кубическое уравнение $u^3 + Au^2 + u = 0$ имеет 3 корня в поле F_p: $\{0, -(A \pm \sqrt{A^2 - 4})/2\}$, а кривая Монтгомери содержит 3 точки 2-го порядка: $D_{M0} = (0,0)$, $D_{M1,2} = (-(A \pm \sqrt{A^2 - 4})/2, 0)$, с координатами $v_{0,1,2} = 0$. Преобразованием координат (8) точка D_{M0} кривой (7) переходит в точку $\boldsymbol{D_0} = (-1, 0)$ кривой (1), а две другие точки $D_{M1,2}$ трансформируются в 2 точки 2-го порядка $\boldsymbol{D_{1,2}} = \left(\pm\sqrt{\frac{a}{d}}, \infty\right)$ с делением на 0 y-координаты $y = u/v$. Так как при $x = 0$ из (1) следует $ay^2 = 1$, решения для y-координаты нет и точки 4-го порядка на оси y для этого случая не существуют. Согласно (6), такие точки могут лежать не на оси кривой при выполнении $\left(\frac{ad}{p}\right) = 1$ и $p \equiv 3 mod 4$. Итак, данный случай характерен наличием трех точек 2-го порядка (из них две точки – особые точки на бесконечности) и наличием точек 4-го порядка лишь при $p \equiv 3\text{mod}4$. Заметим, что в данном случае изоморфизм на основе замены $(x,y) \to (X, Y/\sqrt{a})$ (см. п.1.1) построить нельзя из-за несуществования элемента $\sqrt{a}$. В то же время для кривой квадратичного кручения он существует.

Рассмотренный здесь случай наиболее интересен для практических приложений, т. к. кривая $E_{E,a,d}$ имеет минимальное число особенностей.

2.2. $\left(\frac{a}{p}\right) = 1, \left(\frac{d}{p}\right) = 1$. Как и в предыдущем случае, при $v = 0$ дискриминант уравнения (7) $(A^2 - 4) = (Bad)^2$ является квадратом и вновь мы имеем 3 точки 2-го порядка с теми же координатами, что и в п. 2.1. Две из них преобразованием (8) переходят в особые точки 2-го порядка кривой Эдвардса с квадратичным параметром. В отличие от п. 2.1., здесь всегда имеются точки 4-го порядка, в частности, точки $\pm\boldsymbol{F_0} = (0, \pm 1/\sqrt{a})$ на оси y кривой (1). Кроме того, кривая Монтгомери (7) содержит две точки 4-го порядка с координатой $u_1 = -1$, которые отображением (8) порождают особые точки кривой Эдвардса (1). Действительно, из уравнения касательной к кривой (3) в точке 4-го порядка $P_M = (u_1, v_1)$, проходящей через точку (0,0) 2-го порядка, имеем

$$\left.\frac{dv}{du}\right|_{u=u_1} = \frac{3u_1^2 + 2Au_1 + 1}{2Bv_1} = \frac{v_1}{u_1}.$$

Тогда с учетом (7) получим $u_1^2 = 1 \Rightarrow u_1 = \pm 1$. Одна из пар точек 4-го порядка имеет координаты $\pm F_M = (-1, \pm\sqrt{(A-2)B})$. Как следует из (8), эти две точки кривой Монтгомери с координатой $u_1 = -1$ преобразуются в особые точки 4-го порядка кривой (1) с x-координатами $\pm\sqrt{(A-2)B}$, и с делением на 0 y-координаты $y = (u_1 - 1)/(u_1 + 1)$. В итоге в рассматриваемом случае получили 4 особые точки (на бесконечности): по две точки 2-го и 4-го порядков. Для данного случая преобразование координат $(x, y) \to (X/\sqrt{a}, Y)$ дает изоморфную кривой (1) кривую Эдвардса с квадратичным параметром $X^2 + Y^2 = 1 + d'X^2Y^2$, где $d' = d/a \Rightarrow \left(\frac{d'}{p}\right) = 1$ и имеет место изоморфизм $E_{E,a,d} \sim E_{E,1,d/a}$. Кривые этого пункта мы относим к *кривым Эдвардса* (с квадратичным параметром d в поле F_p). У них наихудшие свойства из рассмотренных.

Важно отметить, что для кривых п.2 при обращении параметров $\left(\frac{a}{d}\right) \to \left(\frac{d}{a}\right)$ имеет место изоморфизм $E_{E,a,d} \sim E_{E,d,a}$. Это следует из квадратичности $\left(\frac{ad}{p}\right) = 1$.

Итак, можно разбить все кривые в форме Эдвардса на непересекающиеся классы:

- полные кривые Эдвардса (с условиями п. 1: $\left(\frac{ad}{p}\right) = -1$.);
- кривые Эдвардса с квадратичным параметром (с условиями п.2.2: $\left(\frac{a}{p}\right) = 1 . \left(\frac{d}{p}\right) = 1$.);
- скрученные кривые Эдвардса (с условиями п. 2.1: $\left(\frac{a}{p}\right) = -1 . \left(\frac{d}{p}\right) = -1$.

В работе [1] доказано (теорема 3.3), что закон сложения для кривых Эдвардса является

полным, т.е при любых входах знаменатели в (2) $1+dx_1x_2y_1y_2 \neq 0,\ 1-dx_1x_2y_1y_2 \neq 0$, если параметр d есть квадратичный невычет: $\left(\frac{d}{p}\right)=-1$. Очевидно, что для кривых Эдвардса с квадратичным параметром нарушается полнота закона сложения.

Докажем полноту закона сложения (2) для скрученных кривых Эдвардса.

Теорема 2.1. *При* $\left(\frac{a}{p}\right)=-1\,.\left(\frac{d}{p}\right)=-1$ *и любых входах знаменатели в* (2) $1+dx_1x_2y_1y_2 \neq 0$, $1-dx_1x_2y_1y_2 \neq 0$.

Доказательство. Допустим обратное: *а*) $1+dx_1x_2y_1y_2=0$ или *b*) $1-dx_1x_2y_1y_2=0$.

Пусть справедливо равенство *а*). Тогда $x_1y_1=\frac{-1}{dx_2y_2}$. Рассмотрим квадрат

$$\left(x_1+\sqrt{a}y_1\right)^2=x_1^{\,2}+ay_1^{\,2}+2\sqrt{a}x_1y_1=$$

$$1+dx_1^{\,2}y_1^{\,2}+2\sqrt{a}x_1y_1=$$

$$=1+\frac{1}{dx_2^{\,2}y_2^{\,2}}-\frac{2\sqrt{a}}{dx_2y_2}=(dx_2^{\,2}y_2^{\,2})^{-1}\left(x_2-\sqrt{a}y_2\right)^2.$$

Аналогично получим

$$\left(dx_2^{\,2}y_2^{\,2}\right)\left(x_1-\sqrt{a}y_1\right)^2=\left(x_2+\sqrt{a}y_2\right)^2.$$

При $\left(\frac{d}{p}\right)=-1$ эти равенства справедливы лишь при $x_i=y_i=0$, но такая точка не существует, либо при $x_i=\pm\sqrt{a}y_i$, что невозможно при $\left(\frac{a}{p}\right)=-1$. Итак, допущение *а*) привело к противоречию. Тот же результат получим при допущении *b*). Теорема доказана.

Для криптографических приложений следует искать кривые порядка $N_E=4n$ с минимальным кофактором 4 при нечетном n, из которых отбираются кривые с простым n. Среди полных кривых Эдвардса (условия пп. 1.1 и 1.2) практически половина имеют порядок $4n$ (n – нечетное). Они являются циклическими, и их порядки пробегают все кратные 4-м числа в границах Хассе. Кривые Эдвардса с квадратичным параметром d (п. 2.2) являются нециклическими с тремя точками 2-го порядка и четырьмя точками 4-го порядка. Отсюда следует, что они содержат нециклическую подгруппу, изоморфную $Z/2 \times Z/4$ порядка 8, а порядок этих кривых имеет минимальный кофактор 8. Поэтому кривые порядка $N_E = 4n$ наряду с полными кривыми Эдвардса можно искать лишь среди скрученных кривых в условиях п. 2.1.

В работе [2] доказаны теоремы 3.3–3.5 о бирациональной эквивалентности кривых Эдвардса и Монтгомери. Кривая Эдвардса в этой работе определена как $E_{E,1,d}$ без ограничений на параметр d. По нашей классификации это объединяет полные кривые Эдвардса и кривые Эдвардса с квадратичным параметром. Предлагаемая нами классификация кривых в форме Эдвардса с разбиением их на непересекающиеся классы является логичной и исключающей возможные недоразумения.

В теореме 3.3 [2] доказано, что кривые Эдвардса $E_{E,1,d}$ и Монтгомери $E_{M,A,B}$ бирационально эквивалентны лишь при наличии в них точек 4-го порядка. Далее в теореме 3.4 доказана бирациональная эквивалентность этих кривых и наличие в них точек 4-го порядка при $p \equiv 3\bmod 4$. В частности, для скрученных кривых Эдвардса (с условиями п. 2.1) порядок кривой $N_E^t \equiv 0\bmod 8$. Действительно, для нее парой квадратичного кручения является кривая с условием п. 2.2, имеющая подгруппы 8-го порядка. Следовательно, ее порядок $N_E \equiv 0\bmod 8$. Тогда сумма числа точек пары кривых кручения при $p \equiv 3\bmod 4$ равна

$$N_E + N_E^t = 2(p + 1) = 2(4k+3 + 1) \equiv 0\bmod 8.$$

Отсюда следует $N_E^t \equiv 0\bmod 8$.

При $p \equiv 1\bmod 4$ мы получим

$$N_E + N_E^t = 2(p + 1) = 2(4k+1 + 1) \equiv 0\bmod 4,$$

т.е. с учетом $N_E \equiv 0\bmod 8$ для скрученной кривой Эдвардса порядок $N_E^t \equiv 0\bmod 4$. Ясно, что в этом случае она имеет три точки 2-го порядка и не имеет точек 4-го порядка. Это подтверждается условием (6). Конечно, при этом нет изоморфизма скрученной кривой Эдвардса с кривой $E_{E,1,d}$, имеющей точки 4-го порядка (теорема 3.5[2]). Итак, скрученные кривые Эдвардса с минимальным кофактором порядка $N_E = 4n$ существуют лишь для половины возможных значений модуля $p \equiv 1\bmod 4$.

Условия, порождающие скрученные кривые Эдвардса с минимальным кофактором порядка $N_E = 4n$, можно найти также, опираясь на свойства кривой Монтгомери (7). Мы знаем, что на кривой Монтгомери $E_{M,A,B}$ для первой координаты точки 4-го порядка выполняется равенство $u_1^2=1 \Rightarrow u_1=1$. Подставляя эти значения в (7), получим $Bv^2 = A + 2 = Ba$ и $Bv^2 = A - 2 = Bd$. Следовательно, в условиях п.2.1 при $p \equiv 1\bmod 4$ точек 4-го порядка на кривой Монтгомери и изоморфной ей скрученной кривой Эдвардса не существует. Любая такая кривая имеет порядок $4n$ при нечетном n и $p \equiv 1\bmod 4$.

Обращаясь к примеру кривой

$$E_{M,9,1}: v^2 = u^3 + 9u^2 + u,\ p = 17$$

приведенному в [2] и отвечающего согласно (7) условиям п.2.1., получаем уравнение скрученной кривой $E_{E,11,7}: x^2 + 11y^2 = 1 + 7x^2y^2$ (здесь параметры $a = 11$ и $d = 7$ являются квадратичными невычетами по модулю 17). Кривая Монтгомери $v^2 = u(u + 3)(u + 6)$ имеет порядок $N_E = 20$, содержит три точки 2-го порядка и не имеет точек 4-го порядка. Она является нециклической с типом $T = (2,2,5)$ и представляется прямой суммой циклических подгрупп 2-го и 10-го порядков. Ясно, что она содержит две различные подгруппы простого порядка 5 (всего имеется 8 точек 5-го и

8 точек 10-го порядков). Если уравнение $E_{E,11,7}$ записать как $x^2 = (y^2 - 1)/(7y^2 - 11)$, то и числитель, и знаменатель здесь обращаются в 0 при соответственно $y^2 = 1$ и $y^2 = 4$. Особые точки для координаты x возникают при $y = \pm 2$. Согласно (8) $x = (u - 1)/(u + 1)$ и эти значения отвечают корням $u_{2,3} \in \{-3, -6\}$ кубического уравнения в $E_{M,9,1}$, т.е. особым точкам 2-го порядка $\boldsymbol{D_{2,3}} = (\pm 2, \infty)$. Например, примем $P = (8,1)$, тогда $2P = (3, -5,)$, $4P = (-4,6)$, $8P = (3,5)$. Так как $8P = -2P$, то $10P = O$ и $\mathrm{Ord}P = 10$. Но в подгруппу $<P>$ входит особая точка 2-го порядка $5P = 4P + P = (2, \infty)$. Приняв генератором подгруппы 5-го порядка точку $G = 2P$ простого порядка 5, можно в подгруппе точек $<G>$, не включающей особых точек, использовать арифметику кривых с групповой операцией (2) без особенностей. Это справедливо для любых точек нечетного порядка.

3. ЧИСЛО СКРУЧЕННЫХ КРИВЫХ ЭДВАРДСА ПОРЯДКА 4*n*

Изучив свойства скрученных кривых Эдвардса, нет необходимости обращаться к статистике порядков этих кривых, как это сделано в [2]. На основе нашей классификации, данной во втором разделе, и свойств кривых, мы можем найти точное число этих кривых с минимальным четным кофактором 4-го порядка $4n$ кривой (n – нечетное). Для этого мы рассматриваем лишь случай $p \equiv 1 \bmod 4$, при котором они существуют.

Для полных кривых Эдвардса с условием п.1 число всех кривых равно числу неквадратов (квадратичных невычетов [7]) $(p - 1)/2$. Так как для пары квадратичного кручения справедливо $N_E + N^t_E = 2(p + 1) \equiv 0 \bmod 4$, то из $N_E = p + 1 - t \equiv 0 \bmod 4$ и $p + 1 \equiv 2 \bmod 4$ имеем $\pm t \equiv 2 \bmod 4$. При этом $N^t_E \equiv 0 \bmod 8$. Итак, если порядок одной из кривых имеет минимальный кофактор 4, то порядок кривой кручения имеет минимальный кофактор 8 и наоборот. Поскольку каждой кривой отвечает одна кривая кручения с инверсией $d \to d^{-1}$, то число полных кривых Эдвардса с минимальным кофактором $M_1 = M_{11} + M_{12} = (p - 1)/4$.

Для кривых с условиями п.2 классификации кривые Эдвардса с квадратичным параметром (п. 2.2) строятся с помощью квадратов $\left(\frac{d}{a}\right)$, из которых выбрасываются квадраты 1 и -1, так что остается $(p - 5)/2$ квадратичных вычетов. Так как инверсия $\left(\frac{a}{d}\right)\left(\frac{d}{a}\right)$ дает изоморфную кривую, мы отбрасываем половину значений квадратов и получаем число кривых с минимальным кофактором 8 $M_{2.2} = (p - 5)/4$. Переход к скрученным кривым Эдвардса с минимальным кофактором 4 как квадратичному кручению кривых п.2.2 дает то же число кривых $M_{2.1} = (p - 5)/4$. Все скрученные кривые Эдвардса при $p \equiv 1 \bmod 4$ имеют минимальный кофактор 4-го порядка кривой.

Итак практически половина всех кривых Эдвардса имеет минимальный кофактор 4. Их число для полных кривых Эдвардса и скрученных кривых Эдвардса также приблизительно одинаково.

В этом свете не может не удивить статистика порядков кривых, приведенная в [2]. Она, разумеется, возникла в связи с пересечением классов кривых, определенных в этой работе (например, кривые Эдвардса и полные кривые Эдвардса наполовину пересекаются). В таблицах распределения порядков протестированных кривых они записаны как разные классы. Такое же пересечение мы видим между классами скрученных и полных кривых Эдвардса. Иначе откуда при $p \equiv 3 \bmod 4$ в таблице порядков кривых ($p = 1019$) возникает 236 скрученных кривых Эдвардса с минимальным кофактором 4? Согласно теореме 3.5 [2] в классе скрученных кривых с условиями п. 2.1 таких кривых не существует. Значит, они заимствованы из кривых, изоморфных полным кривым Эдвардса. Ситуация напоминает двойное налогообложение, и одни и те же кривые регистрируются в разных классах.

В теории вероятностей для построения распределений вероятностей событий используются несовместные события. Классификация кривых в форме Эдвардса с непересекающимися классами, предложенная нами в разделе 2, исключает подобные вышеописанные недоразумения. Кроме того, она позволила найти точное решение для числа скрученных кривых Эдвардса с минимальным кофактором 4, что конструктивно для криптографии и делает бессмысленным обращение к статистике.

Заметим, что введение нового параметра a в определение скрученной кривой Эдвардса практически не дает новых полезных свойств и лишь вдвое расширяет множество кривых в форме Эдвардса с минимальным кофактором 4. Вместе с тем такие скрученные кривые существуют лишь при $p \equiv 1 \bmod 4$. Кроме того, в групповой операции появляется дополнительная операция умножения на параметр a, что лишь замедляет вычисления. Правда, этот аргумент несущесвенный, если принять a как минимальный неквадрат в поле F_p и варьировать параметром $\left(\frac{d}{a}\right)$ при поиске простого n. Позитивным аргументом в пользу скрученных кривых Эдвардса является то, что при $p \equiv 1 \bmod 4$ все они имеют порядок $4n$, что упрощает поиск полезных для криптосистем кривых.

Литература

1. *Bernstein Daniel J., Lange Tanja.* Faster addition and doubling on elliptic curves. IST Programme under Contract IST–2002–507932 ECRYPT, 2007. – P. 1–20.
2. *Bernstein Daniel J. , Birkner Peter , Joye Marc , Lange Tanja, Peters Christiane.* Twisted Edwards Curves. IST Programme under Contract IST–2002–507932 ECRYPT,and in part by the National Science Foundation under grant ITR–0716498, 2008. – P. 1–17.
3. *Бессалов А.В., Цыганкова О.В.* Производительность групповых операций на скрученной кривой Эд-

вардса над простым полем. Радиотехника №181, 2015. – С. 58–63.

4. *Бессалов А.В., Цыганкова О.В.* Взаимосвязь семейств точек больших порядков кривой Эдвардса над простым полем. Проблемы передачи информации. – Том 51, вып 4, 2015. – С. 103–109.
5. *Бессалов А.В.* Число изоморфизмов и пар кручения кривых Эдвардса над простым полем. Радиотехника, вып. 167, 2011. – С. 203–208.
6. *Бессалов А.В., Дихтенко А.А., Третьяков Д.Б.* Сравнительная оценка быстродействия канонических эллиптических кривых и кривых в форме Эдвардса над конечным полем. Сучасний захист інформації, №4, 2011. – С. 33–36.
7. *Бессалов А.В., Телиженко А.Б.* Криптосистемы на эллиптических кривых: Учеб. пособие. – К.: ІВЦ «Політехніка», 2004. – 224 с.

Поступила в редколлегию 3.11.2015

Бессалов Анатолий Владимирович, доктор технических наук, профессор, профессор кафедри ММЗИ ФТИ НТУУ «КПИ». Научные интересы: криптография, теория корректирующего кодирования.

Цыганкова Оксана Валентиновна, аспирант кафедры ММЗИ ФТИ НТУУ «КПИ». Научные интересы: эллиптические кривые в форме Эдвардса.

УДК 621. 3.06

Класифікація кривих у формі Едвардса над простим полем / А.В. Бессалов, О.В. Циганкова // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 277–283.

Дано аналіз властивостей точок малих порядків скрученої кривої Едвардса. Введено арифметику для групових операцій з особистими точками цих кривих. Запропоновано класифікацію кривих у формі Едвардса на 3 непересічних класи. Отримано точні формули для кількості кривих різних класів мінімального порядку. Дано критичний аналіз результатів у работах інших авторів.

Ключові слова: еліптична крива, скручена крива Едвардса, параметри кривої, порядок точки, додавання точок, ізоморфізм, квадратичне кручення, квадрат, неквадрат.

Бібліогр.: 7 найм.

UDC 621. 3.06

Classification of curves in the Edwards form over a prime field / A.V. Bessalov, O.V. Tsygankova // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 277–283.

The paper analyzes the properties of points of small orders of the twisted Edwards curve. Arithmetic for group operations with critical points of these curves is introduced. A classification of curves in the Edwards form into three nonoverlapping classes is suggested. Exact formulas for the number of curves of differen classes of minimum order are obtained. A critical analysis of the results of the other authors' works is provided.

Keywords: elliptic curve, twisted Edwards curve, curve parameters, point order, addition of points, isomorphism, square twist, square, nonsquare.

Ref.: 7 items.

УДК 004.056.55

ЕЛЕКТРОННІ ПІДПИСИ НА ОСНОВІ ІДЕНТИФІКАТОРІВ ТА БІНАРНОГО ВІДОБРАЖЕННЯ

Ю.І. ГОРБЕНКО, Р.С. ГАНЗЯ, О.С. АКОЛЬЗІНА

Розглядаються вимоги до криптографічних перетворень типу електронний підпис (ЕП). В основу новітніх вимог покладено необхідність забезпечення криптографічної стійкості (ЕП) у постквантовий період. Даються оцінки криптографічної стійкості стандартам на (ЕП), що визначені в ДСТУ ISO/IEC 14888- 3:2014 (7 та 8 алгоритми). Розглядаються сутності та властивості ЕП, а також умови і перспективи застосування ЕП на ідентифікаторах, що ґрунтуються на бінарних відображеннях (спарюванні) точок еліптичних кривих(ЕК).

Ключові слова: асиметричне криптографічне перетворення, алгоритм Міллера, бінарне відображення(спарювання) точок еліптичних кривих, Вейля та Тейта перетворення, електронний підпис, криптографічна стійкість, механізм електронного підпису, секретний та відкритий ключ.

ВСТУП

У серпні 2015 року Агентство національної безпеки США (АНБ) зробило заяву про необхідність розробки стандартів криптографічного захисту інформації для постквантової криптографії [1–3]. По суті, АНБ безпосередньо вважає, що настав час розробки стандартів, у тому числі на електронний підпис (ЕП), на основі перспективних асиметричних криптографічних перетворень [3]. Ще раніше [1–2] розглядались питання оцінки стану та можливостей створення у найближчий час «істинно» квантового комп'ютера, а також наявності та можливості застосування алгоритмів квантового криптоаналізу. Також в [2] зроблені оцінки криптографічної стійкості ЕП, криптоперетворення в яких здійснюються на основі застосування математичного апарату кілець, скінченних полів, груп точок еліптичних кривих (ЕК), груп точок гіпереліптичних кривих, ідентифікаторів та бінарного відображення (спарювання) точок ЕК, фактор-кілець [1–3, 6, 7, 8] тощо.

Зважаючи на особливу важливість в застосуванні криптографічних перетворень типу ЕП, на міжнародному рівні зроблено чергове корегування відповідних стандартів, у першу чергу ЕП з додатком згідно з ДСТУ ISO/IEC 14888-3 2014 [6]. Стандарт ДСТУ ISO/IEC 14888-3 2014 прийнятий в Україні методом погодження на основі міжнародного стандарту ISO/IEC 14888-2, друге видання 2008-04-15 «Інформаційні технології. Методи захисту. Цифрові підписи з додатком (доповненням) – Частина 2: Базові механізми на основі сертифікатів». Номер посилань ISO/IEC 14888-2:2006. Він, по суті, включає дванадцять механізмів, два з яких були в ISO/IEC 14888-3 1998 (DSA), а три були у ISO/IEC 15946-2: 2002 (EC DSA, EC GDSA, EC RC DSA). В стандарт включено корейський алгоритм електронного підпису, що реалізований в скінченному полі (KC DSA), три алгоритми ЕП Шнорра, стандарт ЕП РФ ГОСТ Р 34.10-2001, а також два механізми, що ґрунтуються на криптографічних перетвореннях на ідентифікаторах зі спарюванням (бінарним відображенням) точок еліптичних кривих.

Третє майбутнє видання ISO/IEC 14888-3 сьогодні готується до введення в дію. Воно відміняє та замінює друге видання (ISO/IEC 14888-3:2006 (E).

В табл. 1 викладається сутність основних вказаних ЕП, аналізуються їх властивості та, а надалі

Таблиця 1

Характеристики алгоритмів ЕП ISO/IEC 14888-3: 2015

Характеристика / алгоритм	Мат. апарат	Загальні параметри	Допустимі розміри параметрів	Ключова система	Допустимі розміри ключів	Стійкість
DSA	Скінченне поле	P, q ,g	$P \geq 2^{2048}$ $q \geq 2^{256}$	(xi,Yi)	$(xi, ki) \geq 2^{256}$ $Yi \geq_2 2^{2048}$	субекспоненційна
DSA1	Математ. кільце	P, q ,g				субекспоненційна
EC DSA	Група точок EC	(a, b, G, n, P, f, u, h)		(di,Qi)	$(di,ki) \geq 2^{160}$	експоненційна
EC G DSA	Група точок EC	(a,b,G, n, P, f,u,h)		(di,Qi)	$(di,ki) \geq 2^{160}$	експоненційна
EC KC DSA	Група точок EC	(a,b,G, n, P, f,u,h)		(di,Qi)	$(di,ki) \geq 2^{160}$	експоненційна
KC DSA	Скінченне поле	P, q ,g	$P \geq 2^{2048}$ $q \geq 2^{256}$	(xi,Yi)	$(xi, ki) \geq 2^{256}$ $Yi \geq_2 2^{2048}$	субекспоненційна
CT EK1	Спарюв. точок EC					експоненційна
CT EK2	Спарюв. точок EC					експоненційна

обговорюються умови і можливості застосування з урахуванням розвитку криптоаналізу та інших вимог. Також в табл. 1 наведені загальні оцінки та допустимі параметри алгоритмів ЕП, що включені в ISO/IEC 14888-3:2015.

Результати аналізу перших шести алгоритмів електронних підписів з додатком наведені в значному числі джерел, у тому числі у наших [2, 7]. В цій статі наводяться результати викладення сутностей, стану стандартизації, дослідження властивостей ЕП з додатком, що ґрунтуються на перетвореннях зі спарюванням точок еліптичних кривих. Увагу до цих перетворень можна пояснити, по-перше, значним періодом досліджень в цьому напрямі та затраченому інтелекту, а, по-друге, спробою визначити – чи можуть такі ЕП застосовуватися у постквантовий період. Зрозуміло, за умови, що застосовуватимуться функції гешування, що будуть стійкими та матимуть допустиму складність гешування, у постквантовий перід. Очевидно, теоретичне доведення криптографічної стійкості механізмів ЕП на ідентифікаторах зі спарюванням точок ЕК, може ґрунтуватись на основі алгебраїчних решіток.

Наші дослідження показали [2, 7–10], що в першу чергу заслуговують уваги 2 методи ЕП, що засновані на криптографічних перетвореннях на ідентифікаторах зі спарюванням точок ЕК.

На основі детального аналізу виявлено два ґрунтовних механізми, які позначимо як IBS1 [4] та IBS2 [5], причому відносно механізму IBS1 існують модифікації.

Метою цієї статі є аналіз сутностей, основних складових механізмів та властивостей ЕП на ідентифікаторах зі спарюванням точок ЕК по критеріях складність – стійкість, в тому числі порівняння перспективних існуючих, але за умови що до них висуваються вимоги забезпечення стійкості проти квантових методів криптоаналізу у постквантовий період, а також розробка відповідних рекомендацій з їх вибору та використання.

1. ЗАГАЛЬНІ ПОЛОЖЕННЯ ЕП НА ІДЕНТИФІКАТОРАХ ЗІ СПАРЮВАННЯМ ТОЧОК ЕК

Сьогодні визначено 2 нових механізми ЕП на ідентифікаторах, побудування яких ґрунтується на використанні спарювання точок еліптичних кривих [2–7]. Доведення їх стійкості ґрунтується на доведенні складності вирішення задачі Діффі – Геллмана у моделі випадкового оракула [2–4]. Використання у таких механізмах спарювання точок ЕК забезпечує нові доказово стійкі механізми ЕП [4–6]. Крім того, на відміну від традиційних ЕП з додатком, що засновані на застосуванні сертифікатів відкритих ключів, в ЕП на ідентифікаторах як відкритий ключ використовуються ідентифікаційні дані користувача. Розглянемо та проведемо аналіз формального механізму ЕП, що ґрунтується на ідентифікаторах та спарюванні точок ЕК, поклавши за основу [3, 2, 7].

Сутність ЕП на ідентифікаторах та криптопримітиви.

Нехай $(G,+)$ та $(V,*)$ – циклічні групи з простим порядком P та P^{β}, а G базова точка ЕК. Нехай також $e: G\times G\to V$ є перетворення спарювання точок ЕК, яке задовольняє такі умови:

– білінійність: $e(x_1+x_2,y)=e(x_1,y)e(x_2,y)$ та $e(x,y_1+y_2)=e(x,y_1)e(x,y_2)$;

– невиродженість: існує $x\in G$ та $y\in G$ такі, що $e(x,y)\neq 1$;

– обчислюваність, коли існує ефективний алгоритм для обчислення спарювання e.

Нехай також циклічна група $(V,*)$ містить у загальному вигляді елементи скінченного поля $F(P^{\beta})$, де P – порядок базової точки, а β – степінь розширення поля [4, 7]. Також, нехай існують функції гешування [2, 4]

$$h:\{0,1\}^{*}\times V\to (Z/lZ)^{\times}, H:\{0,1\}^{*}\to G^{*}, \quad (1)$$

де $G^{*}:=\{0\}$, $V^{*}:=V\setminus 1$.

Звичайно механізми ЕП на ідентифікаторах будуються на основі використання чотирьох алгоритмів : встановлення асиметричної пари ключів третьою довіреною стороною(ТДС); отримання користувачем від ТДС секретного ключа; підписування та перевіряння даних (повідомлення). При чому, в процесі реалізації механізму ЕП взаємодіють три сторони – підписувач, перевірник та ТДС.

Для механізму IBS1, що розглядається, можливими згідно з [4] потенційними є п'ять механізмів – IBS1-1 та IBS1-2, варіації механізму IBS1-2 Ель-Гамаля та Шнорра, а також механізм з декількома довіреними сторонами. Для IBS-2 у [4] визначена одна схема.

2. СУТНІСТЬ ТА ВЛАСТИВОСТІ МЕНХАНІЗМІВ ЕП IBS1-1

Механізм IBS1-1 ґрунтується на чотирьох алгоритмах: встановлення асиметричної пари ТДС, отримання підписувачем свого секретного ключа, підписування та перевіряння.

Алгоритм встановлення асиметричної пари ТДС. Алгоритм виконується для встановлення асиметричної пари ключа для ТДС. Під час його виконання ТДС генерує чи обирає випадкове ціле число $t\in(Z/lZ)^{\times}$, розраховує відкритий ключ $Q_{\text{ТДС}}=tG$ та оприлюднює $Q_{\text{ТДС}}$, де t є секретним ключем ТДС. Таку третю довірену сторону визнано називати уповноваженим на генерацію.

Алгоритм отримання підписувачем свого таємного ключа. Алгоритм встановлення виконується ТДС, коли відповідний підписувач запитує для себе секретний ключ підпису S_{ID}. Нехай ідентифікаційні дані підписувача подаються рядком ID. У відповідь на запит ТДС обчислює для підписувача секретний ключ $S_{ID}=tH(ID)$ та ви-

дає його підписувачу захищеним шляхом. Крок отримання, як правило, виконується один раз для кожного підписувача, при чому використовуються однакові вхідні дані для різних ідентифікаційних даних *ID*.

Алгоритм підписування. В ході підписування підписувач виконує такі кроки:

1. Генерує чи обирає випадкові точку ЕП $P_1 \in G^*$ та ціле *таємне разове* $k \in (Z/lZ)^\times$.

2. Використовуючи спарювання Тейта чи Вейля *e* обчислює відкритий ключ сеансу

$$r = e(P_1, P)^k. \quad (2)$$

3. Використовуючи узгоджену геш-функцію обчислює геш-значення

$$v = h(M, r). \quad (3)$$

4. Обчислює ЕП u, використовуючи точку ЕК P_1, ідентифікатор *ID*, таємний ключ S_{ID} та відкрите значення r у вигляді

$$u = vS_{ID} + kP_1. \quad (4)$$

Електронним підписом повідомлення M є пара $(u, v) \in (G, (Z/lZ)^\times)$.

Алгоритм перевіряння ЕП. Після отримання підписаного повідомлення M^* та підпису (u^*, v^*), перевіряння підпису виконується при знанні загальних параметрів { P,ID}, відкритого ключа ТДС – $Q_{\text{ТДС}}$ та значення отриманого підпису u^* у такій послідовності.

1. Згідно з [2–6] засобом спарювання точок ЕК обчислюється відкритий ключ r^*

$$r^* = e(u^*, P) \cdot e(H(ID), -Q_{\text{ТДС}})^{v^*}. \quad (5)$$

2. З використанням отриманого M^* та обчисленого згідно з (5) r^* обчислюється значення

$$v = h(M^*, r^*). \quad (6)$$

3. Виконується порівняння обчисленого перевірником значення v та отриманого v^*. Якщо

$$v = v^*,$$

то підпис і, відповідно повідомлення M*, приймаються як цілісні та підтверджується справжність підписувача, інакше повідомлення відхиляється.

Можливість та обґрунтованість застосування порівняння випливає із наведеного нижче [4].

Попередній аналіз ЕП механізму IBS1-1

Аналіз наведеного вище підпису показує, що операція підпису може бути оптимізована засобом виконання попередніх обчислень. Так, можна заздалегідь обчислити перетворення Вейля $e(P_1, P)$ для P_1, наприклад, $P_1 = S_{ID}$, та зберігати це значення з підписаним ключем. Для $P_1 = S_{ID}$ це означає, що операція підпису вимагає одного експоненціювання в групі V, одного гешування, та одного множення з використанням елемента групи G.

Перевірка вимагає одного експоненціювання у групі *V*, одне гешування,та двох спарювань. При цьому одне спарювання можна не виконувати, якщо велика кількість перевірок здійснена для однієї особи, заздалегідь розрахувавши $e(H(ID), -Q_{\text{ТДС}})$.

Безпечність механізму IBS1-1. Для доведення безпечності механізму в [4] наводиться теорема, яку можна використати для оцінки криптографічної стійкості.

Теорема 1. Нехай є випадкова модель оракула та криптоаналітик Kp, який може здійснювати щонайменше $n_1 \geq 1$ запитів на геш-значення та отримання стану оракула, а також що найбільше здійснення екзистенційної підробки $n_2 \geq 1$ за час T_A з імовірністю

$$\varepsilon_A \geq a \frac{n_1 n_2^2}{l} \quad (7)$$

для деякої константи $a \in Z^{\geq 1}$. Тоді існує ще один ймовірнісний алгоритм C та константа $c \in Z^{\geq 1}$, такі, що C розв'язує задачу Діффі – Гелмана з урахуванням

$$(P, Q_{\text{ТДС}}, R)$$

для будь-якого даного $R \in G^*$ на вході, за очікуваний час

$$T_C \leq \frac{c n_1 n_2 T_A}{\varepsilon_A}.$$

Якщо G подана як підгрупа іншої групи $\hat{G}$, визначення H може бути складним на практиці.

3. СУТНІСТЬ ТА ВЛАСТИВОСТІ МЕХАНІЗМУ IBS1-2

Загальна схема ЕП. Алгоритм встановлення асиметричної пари ТДС.

Під час встановлення ТДС обирає мономорфізми та обчислює асиметричну пару (*s*, *q*), де *s* – секретний ключ ТДС, а *q* відкритий. Вказаний механізм можна подати у вигляді

$$s: G_1 \to G \text{ та, } q: G_1 \to V$$
$$\text{з } p(s(x)) = q(x) \quad (9)$$

для всіх $x \in G_1$.

Алгоритм отримання підписувачем свого таємного ключа. Алгоритм отримання виконується ТДС, коли підписувач робить запит на отримання секретного ключа ЕП. Відповідно до запиту ТДС, використовуючи ідентифікатор *ID користувача*, ТДС обчислює та надсилає йому захищеним шляхом(довіреним каналом) секретний ключ

$$d = s(H(ID)). \quad (10)$$

Крок отримання, як правило, виконується один раз для кожного користувача (підписувача), з використанням однакових вхідних даних ТДС для різних ID. Відкритий ключ підписувача

$$y = q(H(ID)), \quad (11)$$

як правило, розраховується перевірником з використанням ID підписувача згідно з відображенням *q*.

Алгоритм підписування. В ході підписування підписувач виконує такі кроки:

1. Генерує чи обирає *таємне разове* $k \in (Z/lZ)^\times$.

2. Використовуючи спарювання *e*, обчислює відкритий ключ сеансу

$$r = p(k). \quad (12)$$

3.Використовуючи узгоджену геш-функцію, обчислює геш-значення

$$(v,w)=h(m,r). \quad (13)$$

Обчислює ЕП і, *використовуючи точку ЕК* P_1, *ідентифікатор ID*, *таємний ключ* S_{ID} та *відкрите значення сеансового ключа v*, у вигляді

$$u=va+wk. \quad (14)$$

Електронним підписом повідомлення M є пара

$$(u,r)\in G\times V^*. \quad (15)$$

Алгоритм перевіряння ЕП. Після отримання підписаного повідомлення M^* та підпису (u^*,r^*), перевіряння підпису виконується при знанні загальних параметрів {P, ID}, відкритого ключа ТДС – $Q_{\text{ТДС}}$ та значення отриманого підпису u^* у такій послідовності.

1. Згідно з [4] обчислюються параметри

$$(v,w)=h(m,r). \quad (16)$$

2. З використанням отриманого ідентифікатора ID підписувача, узгодженої геш-функції та відображення q обчислюється значення відкритого ключа.

$$y=q(H(ID)). \quad (17)$$

3.Виконується перевірка співвідношення

$$p(u)=y^v r^w. \quad (18)$$

4. Якщо (18) виконується, то повідомлення M^* приймається як цілісне та підтверджується справжність підписувача, інакше повідомлення відхиляється. Можливість та обґрунтованість застосування порівняння випливає із наведених нижче алгебраїчних перетворень [5]

$$p(u)=p(va+wk)=p(a)^v\,p(k)^w=$$
$$=p(s(H(ID)))^v\,p(k)^w=$$
$$=q(H(ID))^v\,p(k)^w=y^v r^w.$$

4. МОДИФІКАЦІЯ МЕХАНІЗМУ IBS1-2

В ході реалізації механізму IBS1 – 2 можна скористатися такими модифікаціями [4,5]. Так підписувач та перевірник можуть попередньо замість $(v,w)=h(m,r)$ обчислити $(w,v)=h(m,r)$. Далі, якщо $G=(Z/lZ)^+$, то можна додатково задати $h(m,r)$ будь-якої з шести комбінацій

$$(u,v),(u,w). \quad (19)$$

Перевірник у цьому випадку буде вирішувати відносно змінної, що залишилась, рівняння

$$u=va+wk.$$

Тоді ЕП складатиметься зі значення цієї змінної, що лишилась, та r. Також необхідно зазначити, що якщо $G=(Z/lZ)^+$, то необхідні додаткові уточнення. Але головне відносно модифікації це те, що наведені перетворення є однаково безпечні в моделі випадкового оракула [4]. Зазначене пояснюється тим, що відповідні елементи кортежу (u,v,w) є обчислювано нерозрізнювані, бо a, $k\neq 0$.

Також можливі модифікації відносно застосування геш-функції. Так, якщо

$$h_1:\{0,1\}^*\to(Z/lZ)^\times, h_2:V\to(Z/lZ)^\times$$

та $h_3:\{0,1\}^*\times V\to(Z/lZ)^\times$

є геш-функціями, то можна вважати, що:

$$h(m,r):=(h_1(m),h_2(r))$$
або $h(m,r):=(h_3(m,r),1)$.

Вибір $h(m,r):=(h(m),r)$, очевидно не буде допустимим. Випадок $h(m,r):=(h_3(m,r),1)$ представляє окремий інтерес. Дійсно, для механізму IBS1-2 при $w=1$ для рівняння перевірки маємо, що

$$p(u)=y^u r.$$

Це означає, що можна розрахувати $r=p(u)y^{-v}$, яке еквівалентно тому, що якщо замість (u,r) задати (u,v) як підпис, то залежно від V, можна зменшити просторову складність.

Необхідно відмітити, що наведені модифікації підписання та перевірки є простими, але вони не порушують визнаний стандартний механізм Шнорра [4].

5. МЕХАНІЗМИ З ДЕКІЛЬКОМА ТДС

Наш аналіз, проведений з урахуванням [2, 7–10], дозволив виявити декілька проблемних питань.

Першим з них є необхідність повної довіри до ТДС, оскільки вона генерує таємні ключі для усіх користувачів–підписувачів. По суті, такий механізм не дозволяє повною мірою забезпечити послугу неспростовності відправника, оскільки ТДС має доступ до секретних ключів підписувачів. Але існують модифікації такого механізму, що дозволяють забезпечити секретність на основі розподілення таємниці [2].

Іншою проблемою є необхідність передачі від ТДС секретних ключів ЕП користувачам? Для її вирішення кожен підписувач повинен мати конфіденційний зв'язок з ТДС, тобто секретний канал зв'язку. При чому має бути довіра до таких каналів зв'язку, особливо в частині неспростовності відправника. Також проблемними є питання управління ключами в такій конфігурації системи.

Певне вирішення названих проблем може бути досягненим у ході застосування декількох ТДС [11]. Наприклад, таким чином.

Нехай $s_i:G_1\to G$ та $q_i:G_1\to V$ є секретним та відкритим мономорфізмом ТДС$_i$ для $1\le i\le n$. Для такої умови можна використовувати віртуальну довірену сторону ТДС, визначену мономорфізмами

$$s(x):=\sum_{i=1}^{n}s_i(x) \text{ та } \prod_{i=1}^{n}q_i(x), \quad (20)$$

такими, що

$$p(s(x))=q(x)$$

для всіх $x\in G_1$.

В цьому випадку, оскільки s_i є випадкові та незалежні, то ймовірність того, що s та q не будуть ін'єктивні є $1/l$. Підписувач обчислює свій особистий ключ

$$s\left(H\left(ID\right)\right)=\sum_i s_i\left(H\left(ID\right)\right), \qquad (21)$$

причому $s_i\left(H\left(ID\right)\right)$ він отримує від кожної ТДС$_i$. Перевірник визначає відкритий ключ підписувача також на основі запитів до розподіленого ТДС

$$q\left(H\left(ID\right)\right)=\prod_i q_i\left(H\left(ID\right)\right), \qquad (22)$$

причому він розраховує $q_i\left(H\left(ID\right)\right)$, використовуючи відкритий відповідний мономорфізм q_i для кожної із ТДС$_i$.

6. БЕЗПЕЧНІСТЬ МЕХАНІЗМІВ IBS1

В ході оцінки безпечності механізму ***IBS1*** покладемо, що зловмисник С є ймовірнісною поліноміальною машиною Тьюрінга, яка приймає q як глобальні параметри механізму та випадкову стрічку. Метою порушника є спроба здійснити екзистенційну підробку ЕП при відомому ID підписувача, як найбільш загрозливу атаку. Також, для полегшення вирішення порушником задачі екзистенційної підробки вважатимемо, що його система криптоаналізу може звертатись з запитами до таких чотирьох оракулів:

– *оракул обчислення геш-значення для заданого ідентифікатора*: – для будь-якого ID оракул обчислює відповідне геш-значення $H\left(ID\right)$;

– *оракул отримання користувачем секретного ключа* – для будь-якого ID оракул виробляє відповідний таємний ключ $s\left(H\left(ID\right)\right)$;

– *оракул обчислення геш-значення повідомлення* – для будь-якого повідомлення m та $r\in V^*$ оракул обчислює відповідне геш-значення $h\left(m,r\right)$;

– *оракул ЕП* повідомлення m: для будь-якого повідомлення m та ідентифікатора ID оракул виробляє для повідомлення m ЕП користувача з ідентифікатором ID.

За вказаних умов безпечність механізму IBS1 відносно екзистенційної підробки ЕП можна оцінити, використовуючи нижченаведену теорему 2.

Теорема 2 [4]. Нехай існує випадкова модель оракула та адаптивна друга сторона С, яка здійснює щонайбільше $n_1\geq 1$ запитів на геш-значення ID та отримання стану оракула, а також щонайбільше $n_2\geq 1$ запитів на геш-значення повідомлення та підпису за час здійснення екзистенційної підробки T_A з імовірністю

$$\varepsilon_A\geq a\frac{n_1{n_2}^2}{l} \qquad (23)$$

для деякої константи $a\in Z^{\geq 1}$. Тоді існує ще один ймовірнісний алгоритм B та константа $b\in Z^{\geq 1}$, такі, що B знаходить $s\left(R\right)$;
для будь-якого даного P, $R\in G^*$ та $Q\in G^*$ з $Q=s\left(P\right)$ на вході, за очікуваний час

$$T_C\leq\frac{cn_1n_2T_A}{\varepsilon_A}. \qquad (24)$$

Необхідно відмітити, що доведення безпеки IBS-1 та IBS-2 базується на вирішенні задачі Діффі – Геллмана в групі G. Як показано в [4–8] немає формального методу зменшення складності вирішення задачі Діффі – Геллмана в групі G.

Таким чином, у теоремі 2 стверджується, що для забезпечення захищеності механізму ЕП із заданими p,q та одним або декількома рівняннями обчислення $Q=s\left(P\right)$, експоненційно складно (неможливо) обчислити інші значення ключів $s\left(P\right)$. Якраз у цьому випадку, знання секрету ТДС про s, забезпечує узагальнену односпрямованість для p та q.

7. ПОРІВНЯННЯ МЕХАНІЗМУ IBS1 З ІНШИМИ МЕХАНІЗМАМИ ЕП

Наведені вище дані та попередній аналіз відносно механізму IBS1, дозволяють зробити такі висновки.

1. Алгоритми встановлення та отримання ключа є віртуально однаковими для всіх чотирьох схем та сумісні із схемою на ідентифікаторах. У табл. 2 наведені значення складності підпису та перевірки для механізму IBS1 та механізмів, взятих із [4, 5], а також вид відображення. При чому, E позначає складність експоненціювання в V, M складність скалярного множення в G, SM складність одночасного скалярного множення виду $\lambda P+\mu Q$ у групі G, а P – складність обчислення спарювань. В ході оцінки складність обчислення геш-значення не береться до уваги.

Таблиця 2

Порівняння складності механізму IBS1

	Схема 1	[5]	[17]
Підписування	1E+1M	2M	2M
Перевірка ЕП	1E+2P / 1E+1P	1M+2P	3P / 2P
Вид відображення	$G\times(Z/lZ)^{\times}$	$G\times G$	$G\times G$

2. Проведений аналіз та дані, що взяті із [4, 9–10], дозволяють зробити висновок про те, що операція спарювання має найбільшу складність, тобто потребує найбільше процесорного часу. Крок перевірки механізму IBS1 може бути оптимізований до 1E+1P. Механізм, що наведений в [5], потребує 3P у звичайній версії та 2P в оптимізованій версії, він є найбільш складним.

3. Доведення безпеки механізму IBS1 та механізму із [5] базується на вирішенні проблеми Діффі – Геллмана в групі G. При цьому, немає формального методу зменшення складності вирішення задачі Діффі – Геллмана в групі G.

4. Основним висновком, який можна зробити, є висновок про те, що механізм IBS1 може забезпечити переваги у швидкодії, вимогах до з'єднання та безпеки, що може бути доведеною.

На наш погляд, механізмом, що може конкурувати з механізмом IBS1, є механізм із [5], який

позначатимемо як IBS2. В табл. 2 наведено відповідні оцінки складності ЕП для IBS1 та IBS2. При цьому, механізм IBS1, порівняно з механізмом IBS2, має приблизно вдвічі менше складність перетворення.

8. ПРОБЛЕМНІ ПИТАННЯ БЕЗПЕКИ ОТРИМАННЯ СЕКРЕТНОГО КЛЮЧА

Вище зазначалося, що основним недоліком механізмів на ідентифікаторах є можливість доступу до секретного ключа зі сторони ТДС. Це означає, що вона може виконуватися за користувача ЕП з такими можливостями як і користувач – власник секретного ключа, що запитує його у ТДС. Це дуже важливий недолік і він певним чином має виключатися.

Основним механізмом протидії такій атаці є розподілення таємниці, коли секретний ключ виробляється користувачем на основі відповідного числа приватних таємниць. Більш детально викладення сутності та оцінки таких протоколів наведено, наприклад, у [11, 12].

Розглянемо такий протокол більш детально. Нехай $ТДС_i \in i$ із n незалежних сторін, $1 \le i \le n$. Розподілимо головний секрет t серед n довірених сторін таким чином. Кожна із $ТДС_i$ генерує незалежно свій особистий ключ t_i та оприлюднює $Q_{ТДС} = t_i P$. Особистий майстер ключ ТДС визначається як

$$t = \sum_{i=1}^{n} t_i , \qquad (25)$$

причому кількість учасників може бути невідомою для жодного учасника схеми. Відкритий майстер ключ ТДС визначається як

$$Q_{ТДС} = \sum_{i=1}^{n} Q_{ТДС_i} . \qquad (26)$$

Тоді підписувач, отримавши свою частку секретного ключа по захищеному каналу від кожної довіреної сторони $ТДС_i$ з

$$S_{ID}^{(i)} = t_i Q_{ID} , \qquad (27)$$

розраховує на основі (27) ключ підписувача у вигляді

$$S_{ID} = \sum_{i=1}^{n} S_{ID}^{(i)} . \qquad (28)$$

В цьому випадку для компрометації особистого ключа підписувача, приватні довірені сторони повинні домовитися згідно з умовами (n, n) чи навіть (n, k) сторонньої схеми розподілу секрету, де $k \le n$. При цьому, підписувач у процесі обчислення секретного ключа може перевірити цілісність, справжність та авторство частки секретного ключа на основі застосування ЕП та в подальшому визначити, яка саме $ТДС_i$ видала хибний чи з помилкою приватний t_i ключ.

ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ

1. Механізм ЕП IBS-1 на ідентифікаторах зі спарюванням точок ЕК відповідає вимогам, що висуваються до перспективних ЕП. Під час його застосування забезпечується захист від найбільш загрозливої атаки – екзистенційної підробки на основі атаки з адаптивним вибором повідомлення.

2. Для реалізації ЕП на ідентифікаторах зі спарюванням необхідно знайти підходящі групи G,V та спарювання $e: G \times G \to V$. Як такі групи можна використати групи, утворені скінченими полями та еліптичними кривими над скінченими полями, а спарювання – отримані зі спарювань Вейля чи Тейта.

3. Більш точно як G можна використати підгрупу точок еліптичної кривої над скінченим полем, а V – підгрупа циклічної групи більшого скінченого поля.

4. В механізмі IBS1 ЕП складається замість $r \in V$ з $v \in (Z/lZ)^{\times}$, що дає більш ефективну схему.

5. Спарювання Вейля і Тейта ефективно обчислюються з рекомендованого алгоритму Міллера. При цьому спарювання Тейта порівняно зі спарюванням Вейля має приблизно на п'ятдесят відсотків меншу складність, ніж спарювання Вейля [4, 5].

6. В ході реалізації ЕП можуть бути використаними тільки суперсингулярні еліптичні криві, які утворюють не F_q-раціональний ендоморфізм.

7. Будь-яка суперсингулярна крива повинна мати вигляд

$$y^2 + y = x^3 + ax + b$$

над двійковим полем. Через існування атаки Вейля, при розширенні скінченного поля краще вибирати m непарне. В цьому випадку всі суперсингулярні еліптичні криві є ізоморфізмами для однієї з трьох кривих спеціальної форми для суперсингулярних еліптичних кривих.

8. Таким чином, у статті наведено сутність, огляд та аналіз механізмів ЕП на ідентифікаторах з використанням спарювань IBS1 та визначено проблемні питання. Їх потрібно врахувати в процесі впровадження та застосування алгоритмів 7 та 8 ДСТУ ISO/IEC 14888-3.

9. Основною перевагою механізмів IBS1 є захищеність від найбільш загрозливої атаки – від екзистенційної підробки на основі атаки з адаптивним вибором повідомлення.

Література

[1] *Горбенко Ю. І.* Аналіз шляхів розвитку криптографії після появи квантових комп'ютерів / Ю. І. Горбенко, Р. С. Ганзя // Компютерні системи та мережі : Вісник національного університету «Львівська політехніка». – 2014. – № 806. – С. 40–48.

[2] *Горбенко Ю.І.* Методи побудування та аналізу, стандартизація та застосування криптографічних систем. Монографія. – Харків: Форт, 2015. – 959 с.

[3] A RIDDLE WRAPPED IN AN ENIGMA. NEAL KOBLITZ AND ALFRED J. MENEZES Department of Mathematics, Box 354350, University of Washington, Seattle, WA 98195 U.S.A.

[4] Efficient Identiti Based Signature Schemes based on Pairings. Florian Hess. Dept. Computer Science. University of Bristol. Florian @cs.bris.ac.uk

[5] An Identity-Based Signature from Gap Diffie-Hellman Groups Jae Choon Cha1 and Jung Hee Cheon2 1 Department of Mathematics Korea Advanced Institute of Science and Technology Taejon, 305–701, Korea jccha@knot.kaist.ac.kr

[6] ISO/IEC 14888-3:1998 Information technology – Security techniques – Digital signatures with appendix – Part 3: Certificate-based mechanisms.

[7] *Горбенко І. Д., Погребняк К. А.* Схема цифрового підпису із використанням парних відображень на основі стандарту ДСТУ4145-2002 / І. Д. Горбенко, К. А. Погребняк // Прикладная радиоэлектроника : Тематический выпуск, посвященный проблемам обеспечения информационной безопасности. – 2009. Том 8. № 3. – С. 290–295.

[8] *Горбенко І. Д.* Метод побудування випадкових бітів на основі спарювання точок еліптичних кривих / І. Д. Горбенко, Н. В. Шапочка, К. А. Погребняк // Прикладна радіоелектроніка. – 2010. Том 9, №3. – С. 386–393.

[9] *Горбенко І.Д., Мелецький А.П., Погребняк К.А., Шевченко Д.В.* Билинейное спаривание точек эллиптических кривых и его теоретические основы. // Прикладная радиоэлектроника. – 2006. Том 5. № 1. – С. 3–13.

[10] *Горбенко І.Д., Мелецький А.П., Погребняк К.А., Шевченко Д.В.* Методы выполнения билинейных спариваний точек эллиптических кривых в криптографических приложениях. // Прикладная радиоэлектроника. – 2006. Том 5. – № 1. – С. 13–18.

[11] Б*ондаренко М.Ф., Кравченко П.О.* Комбінована інфраструктура відкритих ключів // Прикладная радиоэлектроника. – 2009. – Том 8. № 3. – С. 327–330.

[12] *Бондаренко М.Ф., Горбенко І.Д., Кравченко П.О., Мелецький О.П.* Аналіз та перспективи сучасних протоколів видання та генерації ключів на базі ідентифікаторів // Прикладная радиоэлектроника. – 2007. – Том 6. № 3. – С. 256–263.

Надійшла до редколегії 12.11.2015

Горбенко Юрій Іванович, провідний науковий співробітник ХНУ ім. В.Н. Каразіна, канд. техн. наук, лауреат державної премії в галузі науки та техніки. Наукові інтереси: інфраструктури відкритих ключів, захищені комп'ютерні системи та мережі, криптологія.

Ганзя Роман Сергійович, випускник кафедри безпеки інформаційних технологій ХНУРЕ. Наукові інтереси: дослідження стійкості асиметричних криптосистем.

Акользіна Ольга Сергіївна, студентка кафедри БІКС ХНУ ім. В.Н. Каразіна. Наукові інтереси: алгоритми та стандарти цифрового підпису, атаки на алгоритми цифрового підпису.

УДК 004.056.55

Электронные подписи на основе идентификаторов и бинарного отображения / Ю.И. Горбенко, Р.С. Ганзя, О.С. Акользина // Прикладная радиоэлектроника: науч.-техн. журнал. – 2015. – Том 14. – № 4. – С. 284–290.

Приводятся результаты оценки криптографической стойкости электронных подписей (ЭП), которые определены в ДСТУ ISO/IEC 14888- 3:2014 (7 и 8 алгоритмы). Рассматриваются свойства, условия и перспективы применения ЭП на идентификаторах и бинарных отображениях (спариваниях) точок эллиптический кривых.

Ключевые слова: ассиметрическое криптографическое преобразование, бинарное отображение (спаривание) точок эллиптических кривых, Вейля и Тейта преобразования, электронная подпись, криптографическая стойкость.

Табл. 2. Бібліогр.: 12 найм.

UDC 004.056.55

Electronic signatures based on identities and binary image / Yu.I. Gorbenko, R.S. Ganzya, O.S. Akolzina // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 284–290.

The paper provides the results of evaluation of the cryptographic security of electronic signatures (ES) that are defined in the State Standard ISO / IEC 14888-3: 2014 (7-th and 8-th algorithms). The properties, the conditions and the prospects of ES on the identities and binary mapping (pairings) of points of elliptic curves are considered.

Keywords: asymmetric cryptographic transformation, binary mapping (pairing) of points of elliptic curves, Weil and Tate transformation, electronic signature, cryptographic resistance.

Tab. 2. Ref.: 12 items.

УДК 681.3.06

АНАЛІЗ ЗАХИЩЕНОСТІ АЛГОРИТМУ ЦИФРОВОГО ПІДПИСУ EC KCDSA ВІД АТАК НА ЗВ'ЯЗАНИХ КЛЮЧАХ

О. С. АКОЛЬЗІНА, І. Д. ГОРБЕНКО

Розглядаються сутність, умови та можливості здійснення порушниками атак з метою підробки електронних підписів на основі зв'язування особистих ключів електронного підпису, наводяться рекомендації відносно забезпечення захисту від атак на основі зв'язування особистих ключів.

Ключові слова: алгоритм електронного підпису згідно з ДСТУ ISO/IEC 14888-3:2014, атака на основі зв'язування ключів, особисті сеансові та довгострокові ключі.

ВСТУП

Алгоритм електронного підпису (ЕП) EC KCDSA вважається одним з найбільш надійних криптографічних алгоритмів ЕП на світовому рівні, він також реалізований у національному стандарті ДСТУ ISO/IEC 14888-3. Вказаний стандарт введено в Україні в дію наказом Мінекономрозвитку №1493 з 1 січня 2016 року. Незважаючи на його рівень стандартизації, існує необхідність детального вивчення його на предмет криптографічної стійкості від існуючих та потенційних атак.

Метою цієї статті і є аналіз захищеності алгоритму електронного підпису EC KCDSA від атак на зв'язаних ключах та розробка рекомендацій відносно протидії їм.

1. СУТНІСТЬ, ПРИЗНАЧЕННЯ ТА ОСНОВИ ЗАСТОСУВАННЯ EC KCDSA

Алгоритм EC KCDSA – це алгоритм, що розроблений командою Korea Internet & Security Agency, він є дещо аналогом алгоритму KCDSA, але на еліптичних кривих. В алгоритмі реалізується механізм вироблення ЕП з додатком (доповненням) [1, 2]. Цей алгоритм відкликано з [1] та введено в [2].

ЕП повідомлення M згідно з EC KCDSA, що виробляється підписувачем, складається з двох складових – цілих чисел r та s, обчислених за модулем простого числа – порядку базової точки.

В процесі підписування повідомлення M згідно з EC KCDSA підписувач виконує такі кроки:

1. Обчислюється геш-значення повідомлення M та даних сертифіката z_A, тобто

$$e = h(z_A \| M).$$

2. Генерується (вибирається) випадкове ціле число – ключ сеансу k.

3. Обчислюється геш-значення

$$r = h(\pi(kG)) = h(c),$$

де π є функцією, що виділяє x – координату точки kG та перетворює її у рядок байтів і присвоює йому значення змінної c, а також обчислюється геш-значення $h(c)$.

4. Здійснюється зв'язування значення r та геш-значення e у вигляді $w = r \, XOR \, e$.

5. З використанням особистого ключа d, таємного ключа сеансу k та значення w обчислюється значення ЕП s за модулем порядку базової точки n:

$$s = d(k - w) \bmod q. \quad (1)$$

На останок до повідомлення M додається значення ЕП – (r, s).

Перевіряння ЕП повідомлення M', наприклад, об'єкта A об'єктом B, здійснюється у такій послідовності:

1. Перевіряється виконання умов, що $0 < s' < n$ та $len_{r'} \le len_{h()}$, якщо хоч одна умова не виконується, тоді повідомлення відхиляється.

2. Обчислюється геш-значення

$$e' = h(z_A \| M').$$

3. Обчислюється значення $w' = r' \, XOR \, e'$.

4. Обчислюється значення точки еліптичної кривої у вигляді

$$(x_1', y_1') = s'Q_A + w'G. \quad (2)$$

5. Виділяється та перетворюється значення координати x_1' в рядок байтів, якому присвоюється значення c.

6. Обчислюється геш-значення $v = h(c)$.

Якщо $r' = v$, тоді ЕП і відповідно повідомлення М приймаються перевірником як справжні та цілісні, інакше – відхиляється.

Примітка. Якщо використовують несправжнє значення відкритого ключа Q або несправжнє значення Z_A, тоді процес перевіряння ЕП повідомлення M буде з великою ймовірністю невдалим.

2. СТІЙКІСТЬ АЛГОРИТМУ ДО ОСНОВНИХ ВИДІВ АТАК

Основною вимогою до ЕП є забезпечення криптографічної стійкості проти усіх відомих та потенційних атак, причому складність має носити експоненційний характер [2, 3, 5–7].

На сьогодні існує значне число атак на ЕП, достатньо детально вони описані та аналізуються в [5–7].

Як правило, атаки реалізуються в процесі здійснення криптоаналізу. При здійсненні криптоаналізу вважається, що криптоаналітику відомі методи, алгоритми й протоколи ЕП, загально-системні параметри та відкриті ключі. Вважається також, що криптоаналітик володіє усім необхідним програмним та апаратним забезпеченням ЕП, невідомим є тільки особистий ключ. Додатково може бути відомий інтервал

або клас, якому належить особистий ключ, вага Хеммінга ключа, або його ймовірнісний розподіл. Звичайно ці відомості спрощують задачу знаходження дискретного логарифма.

Доведено [1–3, 5–6], що агоритм ЕП ЕС KCDSA захищений від таких типів атак, як: «повне розкриття», на основі підписаних даних, «екзистенційна підробка», «селективна підробка». Така захищеність забезпечується, в тому числі, за рахунок обов'язкового використання колізійно-стійких функцій гешування [4–5].

Також обов'язковими умовами забезпечення криптографічної захищеності від атак є забезпечення конфіденційності, цілісності, справжності (автентичності), доступності, надійності та неспростовності особистого ключа. Відносно відкритого ключа не вимагається забезпечення його конфіденційності, обов'язково відносно нього мають забезпечуватися цілісність, справжність (автентичність), доступність, та неспростовність.

Зазначені послуги, як правило, забезпечуються засобом захищеного генерування особисто власником (підписувачем) асиметричних пар ключів, безпечним зберіганням та застосуванням особистого ключа, виготовленням та обслуговуванням сертифікатів відкритих ключів у третьої довіреної сторони, як правило в центрі сертифікації ключів відповідної інфраструктури відкритого ключа [2–4].

До цих пір у доступних джерелах вважалось [1, 5–6], що ЕП ЕС KCDSA захищений від атаки на зв'язаних ключах. Наші дослідження показали, що атака вказаного ЕП на зв'язаних ключах існує і є критичною за певних умов для підписувача. Тому розглянемо це питання детально.

3. СУТНІСТЬ ТА УМОВИ ЗДІЙСНЕННЯ АТАКИ НА ЗВ'ЯЗАНИХ СЕАНСОВИХ КЛЮЧАХ

При атаці на зв'язаних ключах, на відміну від інших атак, порушник певним чином може мати доступ до ключа сеансу чи довгострокового особистого ключа, наприклад, за рахунок влаштування спеціальних лазівок у програмне чи програмно-апаратне забезпечення, яке використовується в ході вироблення ЕП.

У подальшому вважатимемо, що порушником є криптоаналітик третього рівня [5–6], який під час виконання криптоаналізу може спиратися на науково-технічний ресурс, що прирівнюється до науково-технічного ресурсу спецслужби економічно розвиненої держави.

Розглянемо санкціонованого користувача, який здійснює такі зловмисні дії. Робиться спроба для інформації (повідомлень) M_i та M_j виробити однакові ЕП. Якщо це буде зроблено, то далі зловмисник може маніпулювати цими підписаними повідомленнями, висуваючи або передаючи в ході реалізації загроз те чи інше повідомлення, але обидва вони прийматимуться отримувачем як справжні та цілісні з правильним підписом [5–6].

Нехай зв'язаними є ключі k_1 та k_2 сеансів, тобто які мають змінюватися при кожному виробленні ЕП. В ході аналізу вважатимемо, що особистий ключ d_a є дійсним протягом деякого часу ΔT, а k_1 і k_2 є зв'язаними ключами відповідних сеансів ЕП, причому $k_1 + k_2 = n$. В цьому випадку $k_2 = n - k_1$. Результати аналізу вироблення підписів для повідомлень M_i та M_j наведені в табл. 1.

Аналіз даних табл. 1 показує, що відкриті сеансові ключі r_1 та r_2 в алгоритмі ЕП ЕС KCDSA в ході застосування їх для підпису різних повідомлень M_i та M_j зв'язаних ключів, співпадають. Вказане є значним недоліком алгоритму ЕП ЕС KCDSA. Але на кроці 6 ми отримуємо, що з великою ймовірністю $w_1 \neq w_2$, оскільки геш-функція, що застосовується в алгоритмі вироблення ЕП, є колізійно стійкою.

Надалі розглянемо, чи існує можливість, за якої для різних повідомлень M_i та M_j співпадатимуть самі підписи s_1 та s_2 та яка ймовірність такої події. Тобто визначимо умови, за яких компоненти s_1 та s_2 ЕП ЕС KCDSA для двох довільних різних повідомлень M_i та M_j, співпадуть. Для цього прирівняємо праві частини s_1 та s_2 сьомих рядків табл. 1. В результаті маємо.

$$d_a(k_1 - w_1) \bmod n = d_a(k_2 - w_2) \bmod n \qquad (3)$$

або

$$(k_1 - w_1) \bmod n = (n - k_1 - w_2) \bmod n = \\ = -(k_1 + w_2) \bmod n. \qquad (4)$$

Далі отримаємо, що:

$$2k_1 = (w_1 - w_2) \bmod n$$

або

$$k_1 = \frac{w_1 - w_2}{2} \bmod n = \frac{r_1 \oplus h_1 - r_2 \oplus h_2}{2} \bmod n. \qquad (5)$$

Аналіз співвідношень (4) та (5) щодо захищеності ЕС KCDSA дозволяє зробити такі висновки.

Таблиця 1

Алгоритм вироблення ЕП для ЕС-КС DSA зі зв'язаними сеансовими ключами

Для повідомлення M_i	Для повідомлення M_j
1. $h_1 = H(Z_a \parallel M_1)$	1. $h_2 = H(Z_a \parallel M_2)$
2. $k_1 \in [1, n-1]$	2. $k_2 = n - k_1, k_2 \in [1, n-1]$
3. $(x_1, y_1) = k_1 G$	3. $(x_2, y_2) = k_2 G = (n - k_1)G = nG(\bmod n) - k_1 G(\bmod n) = 0 - k_1 G(\bmod n) = (x_1, -y_1)$
4. $x_1 \to c_1$	4. $x_2 = \pi(x_1, -y_1) = x_1 \to c_2 = c_1$
5. $r_1 = H(c_1)$	5. $r_2 = H(c_2) = H(c_1) = r_1$
6. $w_1 = r_1 \oplus h_1$	6. $w_2 = r_1 \oplus h_2$
7. $s_1 = d_1(\mathrm{k}_1 - w_1) \bmod n$	7. $s_2 = d_2(\mathrm{k}_2 - w_2) \bmod n$

1. Значення відкритих сеансових ключів r_1 та r_2 відповідно для k_1 та $k_2 = n = k_1$ особистих сеансових ключів співпадають, незалежно від повідомлень M_i та M_j, які підписуються. Це безпосередньо випливає з третього і п'ятого рядків табл. 1.

2. Співвідношення (5) вказує на умову здійснення атаки на зв'язаних ключах. Для здійснення атаки необхідно обчислити k_1 згідно з (5). Це можна зробити, оскільки r_1 порушнику відоме, а геш-значення повідомлень $h_1(M_1)$ та $h_2(\mathrm{M}_2)$ він може обчислити.

3. Таким чином, алгоритм ЕП ЕС KCDSA незахищений від атаки на зв'язаних сеансових ключах, а співвідношення (5) визначає умову підробки підпису для повідомлення M_j при відомих даних відносно.

4. Для захисту алгоритму ЕП ЕС KCDSA від атаки на зв'язаних сеансових ключах необхідно виключити можливість зв'язування сеансових ключів у процесі підписування повідомлень. Основним підходом до забезпечення вищезазначеного є практичне виключення можливостей запису викривленого програмного забезпечення, що дозволяє зв'язувати сеансові ключі, наприклад, засобом використання апаратно-програмних чи апаратних засобів криптографічних перетворень під час підписування повідомлень.

4. СУТНІСТЬ ТА УМОВИ ЗДІЙСНЕННЯ АТАКИ НА ДОВГОСТРОКОВИХ ЗВ'ЯЗАНИХ КЛЮЧАХ

Розглянемо також можливість та умови здійснення засобом зв'язування довгострокових ключів. Таке зв'язування може бути здійснене на етапах генерування асиметричних пар ключів ЕП.

Нехай зв'язування довгострокових ключів здійснюється таким чином

$$d_2 = n - d_1,\ k_1 = k_2 = k. \qquad (6)$$

У табл. 2 наведений алгоритм вироблення ЕП для EC-KCDSA на довгострокових зв'язаних ключах.

Далі, використовуючи сьомі рядки табл. 2, визначимо умову співпадання інших складових ЕП s_1 та s_2 для зв'язаних довгострокових ключів, наприклад коли $s_1 = s_2$. В результаті маємо, що

$$d_1(k - w_1) = -d_1(k - w_2) \bmod n;$$
$$k - w_1 = -k - w_2 (\bmod n);$$
$$k = (w_1 - w_2)/2 (\bmod n). \qquad (7)$$

На основі аналізу (6) можна зробити висновок, що при зв'язуванні довгострокових ключів за правилом (6) не вдається визначити умову, за якої – $s_1 = s_2$. Тому модифікуємо залежність між довгостроковими ключами таким чином

$$d_2 = n - d_1 \pm 1. \qquad (8)$$

Нехай знову виконується рівність $k_1 = k_2 = k$.

У табл. 3 наведений алгоритм вироблення ЕП для ЕС KCDSA на довгострокових зв'язаних ключах згідно з (8).

Визначимо умову вироблення однакових s – компонент ЕП, тобто: прирівняємо праві частини сьомого рядка табл. 3. В результаті отримуємо

$$d_1(k - w_1) \bmod n =$$
$$= (-d_1 \pm 1)(k - w_2) d_1 (k - w_1) \bmod n = \qquad (9)$$
$$= (-d_1 \pm 1)(k - w_2)$$

або

$$\frac{k - w_1}{k - w_2} = \frac{-d_1 \pm 1}{d_1} (\bmod n).$$

Таблиця 2

Алгоритм вироблення ЕП для EC-KCDSA на довгострокових зв'язаних ключах

Для повідомлення M_i	Для повідомлення M_j
1. $h_1 = H(Z_a \parallel M_1)$	1. $h_2 = H(Z_a \parallel M_2)$
2. $k \in [1, n-1]$	2. $k \in [1, n-1]$
3. $(x_1, y_1) = kG$	3. $(x_2, y_2) = kG = (x_1, y_1)$
4. $x_1 \to c_1$	4. $x_2 = \pi(x_1, y_1) = x_1 \to c_2 = c_1$
5. $r_1 = H(c_1)$	5. $r_2 = H(c_2) = H(c_1) = r_1$
6. $w_1 = r_1 \oplus h_1$	6. $w_2 = r_1 \oplus h_2$
7. $s_1 = d_1(k - w_1) \bmod n$	7. $s_2 = d_2(k - w_2) \bmod n = (n - d_1)(k - w_2) \bmod n = -d_1(k - w_2) \bmod n$

Таблиця 3

Алгоритм вироблення ЕП для EC-KCDSA зі зв'язаними довгостроковими ключами згідно з (8)

Для повідомлення M_i	Для повідомлення M_j
1. $h_1 = H(Z_a \parallel M_1)$	1. $h_2 = H(Z_a \parallel M_2)$
2. $k \in [1, n-1]$	2. $k \in [1, n-1]$
3. $(x_1, y_1) = kG$	3. $(x_2, y_2) = kG = (x_1, y_1)$
4. $x_1 \to c_1$	4. $x_2 = \pi(x_1, y_1) = x_1 \to c_2 = c_1$
5. $r_1 = H(c_1)$	5. $r_2 = H(c_2) = H(c_1) = r_1$
6. $w_1 = r_1 \oplus h_1$	6. $w_2 = r_1 \oplus h_2$
7. $s_1 = d_1(k - w_1) \bmod n$	7. $s_2 = d_2(k - w_2) \bmod n = (n - d_1 \pm 1)(k - w_2) \bmod n = (-d_1 \pm 1)(k - w_2) \bmod n$

Далі отримуємо, що

$$\frac{k-w_1}{k-w_2}=\pm\frac{1}{d_1}-1(\mathrm{mod}n)$$

$$\pm d_1=\frac{k-w_2}{2k-w_1-w_2}(\mathrm{mod}n) \qquad (10)$$

та

$$d_1=\mp\frac{k-w_2}{2k-w_1-w_2}(\mathrm{mod}n) \qquad (11)$$

при варіантах зв'язування

$$d_2=n-d_1+1(\mathrm{mod}n)d_2=n-d_1+1(\mathrm{mod}n)$$

та $d_2=n-d_1-1(\mathrm{mod}n)d_2=n-d_1-1$.

Вираз (11) визначає умову співпадання ЕП при зв'язуванні довгострокових ключів. Якщо враховувати, що термін дії довгострокового ключа щонайменше один рік, то порушник зможе використати зв'язані довгострокові ключі лише один раз. Цього буде достатньо для здійснення підробки.

Отже, якщо буде здійснене зв'язування довгострокових ключів у розглянутий спосіб, то він досягне своєї мети – обидві компоненти підпису (r, s) для різних повідомлень M_1 та M_2 співпадуть, що дозволить підробити ЕП

5. СУТНІСТЬ ТА УМОВИ ЗДІЙСНЕННЯ АТАКИ НА ЗВ'ЯЗАНИХ ДОВГОСТРОКОВИХ ТА СЕАНСОВИХ КЛЮЧАХ

Розглянемо можливість здійснення атаки на основі поєднання двох попередніх атак – зв'язування як довгострокових, так і короткострокових ключів. У цьому випадку не потрібно, щоб $k_1=k_2=k$.

Нехай короткострокові ключі виробляються з порушенням порушником у такий спосіб:

$$k_2=n-k_1 \bmod n, \qquad (12)$$

а довгострокові таким чином:

$$d_2=n-d_1\pm1. \qquad (13)$$

Алгоритм вироблення ЕП при зв'язуванні довгострокових та короткострокових ключів наведений у табл. 4.

Визначимо умову співпадіння s_1 та s_2: прирівнявши праві частини(7) табл. 4.

$$d_1(k_1-w_1)=(-d_1\pm1)(-k_1-w_2)\bmod n; \qquad (14)$$

або

$$\frac{k_1-w_1}{-k_1-w_2}=\frac{-d_1\pm1}{d_1}(\mathrm{mod}n),$$

$$\frac{k_1-w_1}{-k_1-w_2}=\pm\frac{1}{d_1}-1(\mathrm{mod}n),$$

оскільки порушнику відомі k_1, w_1 та w_2, то він може обчислити

$$\pm d_1=\frac{-w_1-w_2}{k_1-w_2}(\mathrm{mod}n) \qquad (15)$$

або

$$d_1=\pm\frac{w_1+w_2}{k_1-w_2}(\mathrm{mod}n). \qquad (16)$$

Зробимо уточнення відносно реалізації атак зі зв'язуванням довгострокових ключів. Так аналіз (10), (11), (15) та (16) показує, що довгостроковий ключ d_1 та сенсовий k_1 можуть бути розраховані тільки під час вироблення ЕП, оскільки значення w_1 та w_2 залежать від самих повідомлень M_i, M_j і обчислюються у процесі підпису. При цьому для реалізації атаки порушник повинен закласти в систему генерування асиметричних пар ключів відповідну закладку, яка дозволяла б у певний момент здійснити зв'язування довгострокових та короткострокових ключів (або встановити однакові короткострокові ключів).

6. ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ

1. Проведені дослідження алгоритму ЕП EC-KCDSA, що входить в ДСТУ ISO/IEC 14888-3:2014, та отримані результати дозволяють зробити висновки про те, що відносно нього існують ефективні атаки підробки ЕП, які грунтуються на зв'язуванні особистих ключів. Також у процесі досліджень отримано нові умови зв'язування особистих ключів та комбінацій сеансових та довгострокових ключів.

2. Одним з найбільш ефективних методів захисту від атак на зв'язаних ключах є використання криптографічних протоколів та надійних апаратно-програмних чи апаратних засобів ЕП, для яких практично не можливо зв'язування ключів.

3. Значно ускладнюється можливість здійснення атаки зв'язування ключів засобом використання в процесі вироблення ЕП геш-значення різних констант, наприклад, залежних від параметрів криптографічних перетворень.

Таблиця 4

Алгоритм вироблення ЕП для EC-KCDSA зі зв'язаними довгостроковими та сеансовими ключами

Для повідомлення M_i	Для повідомлення M_j
1. $h_1=H(Z_a \parallel M_1)$	1. $h_2=H(Z_a \parallel M_2)$
2. $k_1\in[1,n-1]$	2. $k_2=n-k_1 \bmod n$
3. $(x_1,y_1)=kG$	3. $(x_2,y_2)=-k_1G=(x_1,-y_1)=(x_1,\mathrm{y}_1)$
4. $x_1\to c_1$	4. $x_2=\pi(x_1,y_1)=x_1\to c_2=c_1$
5. $r_1=H(c_1)$	5. $r_2=H(c_2)=H(c_1)=r_1$
6. $w_1=r_1\oplus h_1$	6. $w_1=r_2\oplus h_2=r_1\oplus h_2$
7. $s_1=d_1(k-w_1)\bmod n$	7. $s_2=d_2(k-w_2)\bmod n=(n-d_1\pm1)(-k_1-w_2)\bmod n=(-d_1\pm1)(-k_1-w_2)\bmod n$

4. При спробах підробки ЕП відносно алгоритму ЕП EC-KCDSA, що входить в ДСТУ ISO/IEC 14888-3:2014, для порушника більш раціональним є здійснення атаки на основі зв'язування сеансових ключів, тому необхідно в першу чергу здійснювати захист від таких дій порушників.

Література

[1] ISO/IEC 15946-2:2002, Information technology – Security techniques – Cryptographic techniques based on elliptic curves – Part 2: Digital Signatures.

[2] ISO/IEC 14888-3:2006 Information technology – Security techniques – Digital signatures with appendix – Part 3: Discrete logarithm based mechanisms.

[3] ДСТУ ISO/IEC 14888-3:2014 (проект) «Інформаційні технології. Методи захисту. Цифрові підписи з доповненням. – Частина 3: Механізми на основі сертифікатів» [на заміну ДСТУ ISO/IEC 14888-3:2002].

[4] ISO/IEC 10118-3. Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions.

[5] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Теорія. Практика. Застосування. // Монографія. Видання 2-ге, перероблене й виправлене. – Харків. Видавництво «ФОРТ», 2012. – 878 с.

[6] *Горбенко Ю.І.* Побудування та аналіз систем, протоколів та засобів криптографічного захисту інформації. Монографія. – Харків. Форт. 2015. – 959 с.

[7] *Акользіна О.С, Бакликов О.О.* Порівняльний аналіз перспективних стандартів ЕП в групі точок еліптичних кривих // Радіотехніка. – 2015. – 181. – С. 101–109.

Надійшла до редколегії 17.11.2015

Акользіна Ольга Сергіївна, фото та відомості про автора див. на стор. 290.

Горбенко Іван Дмитрович, професор, доктор технічних наук, лауреат державної премії в галузі науки і техніки, головний конструктор ПАТ «ІІТ». Наукові інтереси: криптологія та кібербезпека.

УДК 681.3.06

Анализ защищенности алгоритма цифровой подписи EC-KCDSA от атак на связанных ключах / О.С. Акользина, И.Д. Горбенко // Прикладная радиоэлектроника: науч.-техн. журнал. – 2015. – Том 14. – № 4. – С. 291–295.

Приводятся результаты анализа защищенности алгоритма ЦП EC-KCDSA от атак на связанных ключах. Разработаны варианты атак на связанных долгосрочных ключах.

Даются рекомендации по применению данного алгоритма.

Ключевые слова: цифровая подпись, EC-KCDSA, атака на связанных ключах.

Табл.: 4. Библиогр.: 7 назв.

UDC 681.3.06

Analysis of EC-KCDSA electronic signature algorithm protectability from key-related atacks / O.S. Akolzina, I.D. Gorbenko // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 291–295.

The results of analyzing the EC-KCDSA electronic signature algorithm protectability from key-related attacks are provided. Some methods of long-time related key attacks have been developed. Application recommendations of the said algorithm are given.

Keywords: electronic signature, EC-KCDSA, key-related attack.

Tab.: 4. Ref.: 7 items.

УДК 681.3.06

АНАЛІЗ ЗАХИЩЕНОСТІ АЛГОРИТМІВ ЕЛЕКТРОННИХ ПІДПИСІВ ГОСТ Р 34.10 2012 ТА ДСТУ 4145 2002 ВІД АТАКИ НА ЗВ'ЯЗАНИХ КЛЮЧАХ

В.А. КУЛІБАБА, О.С.ТОЦЬКИЙ, І.Д.ГОРБЕНКО

Запропоновано атаку, що дозволяє визначити особисті ключі електронного підпису ГОСТ Р 34.10 2012 та ДСТУ 4145-2002 шляхом залежності (зв'язування) сеансових ключів. Наводяться математичні та програмні моделі вироблення і перевірки підпису, що передбачає використання для двох різних повідомлень залежних (зв'язаних) сеансових ключів. За допомогою програмної моделі демонструється реальна атака.

Ключові слова: атака на залежних (зв'язаних) ключах, електронний підпис, математична та програмні моделі атак, електронний підпис згідно з ГОСТ Р 34.10 2012 та ДСТУ 4145-2002.

ВСТУП

У зв'язку зі значним поширенням електронних довірчих послуг та їх транскордонного характеру, суттєво розширюється застосування електронних (цифрових) підписів (ЕП). Розроблено та застосовуються ряд стандартів ЕП, число яких уже складає більше двох десятків. В процесі застосування ЕП, які ґрунтуються на різних математичних методах, особливе місце зайняли ЕП у групах точок еліптичних кривих(ЕК). Згідно з основними вимогами ЕП мають забезпечувати криптографічну стійкість від усіх відомих та потенційних атак, а також задовольняти вимоги складності перетворень. Такі вимоги суттєво задовольняють криптографічні перетворення в групах точок еліптичних кривих [1–4]. Вони, як випливає із [1], уже отримали за це почесне звання «золотого щита». Тому знайшли широке розповсюдження та застосування ЕП ГОСТ Р 34.10-2012 та ДСТУ 4145-2002 [3, 4], які побудовані на застосуванні математики еліптичних кривих.

Водночас важливим є подальше дослідження їх захищеності від потенційних атак. До потенційних атак необхідно віднести атаки, що ґрунтуються на зв'язуванні ключів – сеансових та довгострокових.

Метою цієї статі є аналіз захищеності національних стандартів Російської Федерації та України від атак на залежних (зв'язаних) ключах засобом математичного та програмного їх моделювання, а також визначення механізмів захисту від них.

При викладенні отриманих результатів досліджень опиратимемося на результати досліджень, що наведені в [1, 2].

1. МОДЕЛЬ АТАКИ НА АЛГОРИТМ ГОСТ Р 34.10-2012

В процесі математичного та програмного моделювання атаки відносно ГОСТ Р 34.10-2012 вважатимемо, що відомими є відповідні загально-системні параметри та кожен підписувач має асиметричні ключові пари з необхідним рівнем захисту особистого ключа та виготовив сертифікат, використавши послугу третьої довіреної сторони (ТДС).

Алгоритми вироблення та перевіряння ЕП для ГОСТ 34.10-2012 наведені в табл. 1. В таблиці прийнято такі позначення: H – ідентифікатор геш-функції; q – порядок циклічної підгрупи групи точок еліптичної кривої; d – особистий довгостроковий ключ – випадкове ціле число, що задовольняє умову $1 < d < q$; k –ключ сеансу, $1 < k < q$; Q – відкритий ключ перевірки підпису, що задовольняє умову $dP = Q$; P – базова точка еліптичної кривої, що задовольняє умову $qP = 0$; M_i – повідомлення, що підписується.

Проведемо аналіз захищеності алгоритму ГОСТ Р 34.10-2012 від атаки на залежних(зв'язаних) сеансових ключах. Вважатимемо, що протягом дії довгострокового ключа d_i порушник шляхом активації програмної закладки змушує систему використати зв'язаний сеансовий ключ

Таблиця 1

Алгоритми вироблення та перевіряння ЕП для ГОСТ Р 34.10-2012

Вироблення підпису	Перевіряння підпису
1. Обчислення геш-значення повідомлення $h_1' = H(M)$ 2. Перетворення його на ціле число $h_1 = h_1' (\mathrm{mod} q)$ 3. Генерування сеансового ключа $0 < k_1 < q$ 4. Обчислення точки на ЕК $R = (kP) = (x_R, y_R)$ 5. Обчислення відкритого ключа сеансу $r = x_R (\mathrm{mod} q)$	1. Перевірка $0 < r < q$, $0 < s < q$ 2. Обчислення геш-значення повідомлення $h_1' = H(M)$. 3. Перетворення його на ціле число $h_1 = h_1' (\mathrm{mod} q)$ 4. Обчислити $v = h_1^{-1} (\mathrm{mod} q)$. 5. Обчислити значення $z_1 = sv \pmod q$, $z_2 = -rv \pmod q$ 6. Обчислити точку на ЕК $C = (z_1 P + z_2 Q) \bmod q$ 7. $R = x_c \pmod q$
Вихідний результат:	
6. $S_1 = (r_1 d + k_1 h_1) (\mathrm{mod} q)$ 7. Підпис (r, s)	8. Якщо $R = r$, то підпис приймається, в іншому випадку підпис вважається неправильним.

ЕП k для двох різних повідомлень M_i та M_j. Метою атаки є обчислення особистого довгострокового ключа d_i або сеансового ключа k_i, або їх обох і, таким чином їх компрометація.

Модель атаки вироблення ЕП з використанням залежних (зв'язаних) сеансових ключів для двох різних повідомлень M_i та M_j наведена в табл. 2.

Як результат такого ЕП для різних повідомлень M_i та M_j отримаємо з великою ймовірністю різні ЕП $S_1 \neq S_2$, оскільки має використовуватися колізійно-стійка геш-функція ГОСТ Р 34.11-2014. Тому $h_1 \neq h_2$ і, як наслідок, $S_1 \neq S_2$.

Криптоаналітику також відомий відкритий ключ сеансу r, а також попередньо обрані значення M_i та M_j повідомлень, для яких виконується підпис S_1, S_2.

Використовуючи алгоритм ГОСТ 34.10-2012 (табл. 2), обчислимо ЕП S_1, S_2 для повідомлень M_i та M_j та складемо з них систему рівнянь:

$$\begin{cases} S_1 = rd + k_1 h_1 \pmod q \\ S_2 = rd + (q - k_1) h_2 \pmod q. \end{cases} \quad (1)$$

Далі, нехай $k_2 = q - k_1$, тоді

$$\begin{cases} S_1 = rd + k_1 h_1 \pmod q \\ S_2 = rd + (0 - k_1) h_2 \pmod q. \end{cases} \quad (2)$$

Аналіз (2) показує, що маємо систему з двох рівнянь, але уже з двома невідомими d_i та k_i.

Додамо перше рівняння до другого, в результаті отримаємо

$$S_1 + S_2 = 2rd + k_1(h_1 - h_2) (\mathrm{mod}\ q). \quad (3)$$

Підставивши в (3) замість S_2 його значення

$$S_2 = rd - k_1 h_2$$

отримаємо

$$S_1 + rd - k_1 h_2 = 2rd + k_1 h_1 - k_1 h_2. \quad (4)$$

Використовуючи (4), знаходимо особистий довгостроковий ключ

$$d = \frac{S_1 - k_1 h_1}{r} \pmod q. \quad (5)$$

Аналіз (5) показує, що при відомому k_1 та h_1 можна визначити особистий довгостроковий ключ.

Далі підставляємо отримане значення d в друге рівняння системи (1) .

$$S_2 = r * \frac{S_1 - k_1 h_1}{r} - k_1 h_2 \pmod q$$

або

$$S_2 = S_1 - k_1 h_1 - k_1 h_2 \pmod q$$

$$S_1 - S_2 = -k_1 h_1 - k_1 h_2 \pmod q.$$

В результаті отримуємо значення особистого сеансового ключа k_1

$$k_1 = \frac{S_1 - S_2}{h_1 + h_2} \pmod q. \quad (6)$$

Із (6) можна зробити висновок, що значення сеансового ключа k_1 не залежить від довгострокового ключа d.

Тому при зв'язуванні сеансових ключів , знаючи s_1, s_2, h_1 та h_2, завжди можна, використовуючи (5), знайти особистий ключ d, *і* таким чином, компрометувати його.

Для підтвердження коректності атаки розглянемо її програмну модель.

```
Нехай довгостроковий ключ d має вигляд
548209732816469574618988630314004013637555
5211579238059451
Геш-значення повідомлення M1 h1:
340564665507150929337636486477153533243822
2712453430549188
Сеансовий ключ: k1
475156611601463455995777461222646608550515
7812410080806450
K2--
475156611601463455995777461222646608550515
7812410080806450
r--
451255193723105160893431075995032272071001
969660585174766
Підпис M1 s1:
494878545357269566324705163056665869979161
389405389808365
Для M2 маємо.
Довгостроковий ключ d
548209732816469574618988630314004013637555
5211579238059451
Геш значення M2 h2:
103211425288014122828156960969164592775194
5913758625913956
Сеансовий ключ k2(зв'язаний): k2
152553561937204620387801481094959292826203
6960772761477631
K1 =-152553561937204620387801481094959292
82620369607727614776 31
r--
451255193723105160893431075995032272071001
969660585174766
Підпис M2 s2:
238193505364381854027954311031363875966007
8634463134673911
Обчислюємо за формулою (6) особистий(таєм-
ний) ключ сеансу
K1=
475156611601463455995777461222646608550515
7812410080806450
Використовуючи (5) знаходимо особистий дов-
готроковий (таємний) ключ сеансу
475156611601463455995777461222646608550515
7812410080806450
Таким чином, особистий довгостроковий ключ d
обчислено вірно і він стає компрометований.
```

На останок потрібно підкреслити особливість реалізації атаки. Як випливає із наведеного вище порушник повинен зробити такі несанкціоновані дії.

Впровадити в засіб чи систему генерування забезпечення, яке дозволило б:

1. В певний момент здійснити зв'язування сеансових ключів та їх використати при підписуванні двох повідомлень.
2. Перехопити підписані таким чином повідомлення та виділити S – значення ЕП.
3. Обчислити для обох повідомлень їх геш-значення h_1 та h_2.
4. Обчислити згідно з (6) значення сеансового ключа k_1.
5. Обчислити згідно з (5) особистий довгостроковий ключ d.

2. МОДЕЛЬ АТАКИ НА АЛГОРИТМ ДСТУ 4145 2002

Алгоритми вироблення та перевіряння ЕП для ДСТУ 4145-2002 наведено в табл. 3 [4].

Умовні позначення, що використані в таблиці, такі: n – порядок базової точки; d – особистий довгостроковий ключ; f – функція обчислення передпідпису; π – функція виділення x – координати точки ЕК; (iH, M, Д) – підписане повідомлення, причому i – ідентифікатор функції гешування, M – повідомлення, що підписується, Д – ЕП у вигляді (r, s) складових; Q – відкритий ключ; M' – одержане повідомлення; Д = $\{r', s'\}$ – отриманий ЕП.

Проведемо аналіз захищеності алгоритму ЕП ДСТУ 4145-2002 від атаки на зв'язаних ключах, прийнявши, що k_1 та $k_2 = n - k_1$ є зв'язаними сеансовими ключами [1–2].

Алгоритм вироблення ЕП на зв'язаних ключах згідно з ДСТУ 4145-2002 наведено в табл. 4.

Нехай порушник, як для ГОСТ 34.10-2012, може впровадити в засіб чи систему генерування зв'язаних сеансових ключів і для ДСТУ 4145-2002.

Нехай, як і вище, він може скористатися лазівкою.

– Модель атаки на ЕП ДСТУ 4145-2002 зі зв'язуванням сеансових ключів для двох різних повідомлень M_i та M_j наведено в табл. 4.

Використовуючи алгоритм ДСТУ 4145-2002 (табл. 3), обчислимо ЕП S_1, S_2 для повідомлень M_i та M_j та складемо з них систему рівнянь

$$\begin{cases} S_1 = k_1 + dr_1 \pmod n \\ S_2 = n - k_1 + dr_2 \pmod n. \end{cases} \quad (6)$$

Після зведення за модулем n маємо

$$\begin{cases} S_1 = k_1 + dr_1 \pmod n \\ S_2 = -k_1 + dr_2 \pmod n. \end{cases} \quad (7)$$

Далі додамо праві та ліві частини системи (7), в результаті отримаємо, що

$$S_1 + S_2 = dr_2 + dr_1$$

або

$$S_1 + S_2 = d(r_2 + r_1). \quad (8)$$

Далі знайдемо із (8) довгостроковий особистий ключ, в результаті маємо, що:

$$d = \frac{S_1 + S_2}{r_1 + r_1} \pmod n. \quad (9)$$

Далі, підставляючи d в будь-яке рівняння системи (7), можемо отримати значення особистого сеансового ключа k_1,

$$k_1 = (S_1 - dr_1) \pmod n = S_1 - \frac{S_1 + S_2}{2r_1} r_1 \pmod n =$$

$$= S_1 - \frac{S_1 + S_2}{2} \pmod n. \quad (10)$$

На основі співвідношення (10) можна зробити висновок, що алгоритм ЕП згідно з ДСТУ 4145-2002 не захищений від атаки зі зв'язаними сеансовими ключами. При цьому, для обчислення особистого ключа необхідно перехопити підписані повідомлення та виділити значення S_1, S_2 та r_1.

Таким чином стандарт ЕП ДСТУ 4145-2002 уразливий до атаки на зв'язаних ключах, якщо у відповідній системі застосування ЕП існує можливість зв'язування сеансових ключів.

ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ

1. Для алгоритмів ЕП, що визначені в ГОСТ 34.10-2012 та ДСТУ 4145-2002, існує ефективна атака зі зв'язуванням сеансових ключів.

Таблиця 2

Модель атаки на ЕП ГОСТ 34.10 2012 зі зв'язуванням сеансових ключів

Для повідомлення M_i	Для повідомлення M_j
1. $h_1' = H(M_1)$. 2. $h_1 = h_1' (mod\, q)$ 3. $0 < k_{1<} q$ 4. $R_1 = (k_1 G) = (x_{R_1}, y_{R_1})$ 5. $r_1 = x_{R_1} (mod\, q)$	1. $h_2' = H(M_2)$. 2. $h_2 = h_2' (mod\, q)$ 3. $k_2 = q - k_1$ 4. $R_2 = (k_2 G) = ((q-k_1)G) = -(k_1 G) = (x_{R_1}, -y_{R_1})$ 5. $r_2 = x_{R_1} (mod\, q)$
Таким чином, незалежно від M_1 та M_2 завжди $r_1 = r_2 = x_{R_1} = (mod\, q)$	
6. $S_1 = (r_1 d + k_1 h_1)(mod\, q)$	6. $S_2 = (r_1 d + (q - k_1) h_2)(mod\, q)$

Таблиця 3

Алгоритми вироблення та перевіряння ЕП для ДСТУ 4145-2002

Вироблення підпису	Перевіряння підпису
1. Обчислити $k \in [1, n-1]$ 2. Обчислити $R = (kG) = (x_R, y_R)$ 3. Присвоїти $f_k = \pi(x_R, y_R) = x_R$ 4. Цифровий передпідпис (k, f_k) 5. Обчислити $h = H(M)$ та перетворити на елемент основного поля 6. Обчислити елемент основного поля $y = hf_k$ та перетворити на r 7. Обчислити $S = (k + rd)(mod\, q)$	1. $h = H(M')$ 2. Перевірити, що $0 < r' < n$, $0 < s' < n$ 3.Обчислити $R = (s'G + r'Q) = (x_R, y_R)$ 4. Обчислити $y = hx_R$ та перетворити на $\bar{r}$.
8. Результат (iH, M, Д = (r, S))	5. Якщо $r' = \bar{r}$, то підпис дійсний, інакше недійсний

Таблиця 4

Модель атаки на алгоритм електронного підпису ДСТУ 4145-2002

Для повідомлення M_i	Для повідомлення M_j
1. $k \in [1, n-1]$ 2. $f_{k_1} = \pi(k_1 G) = \pi\left(x_{R_1}, y_{R_1}\right) = x_{R_1}$ 3. Формується передпідпис $\left(k_1, f_{k_1}\right) = \left(k_1, x_{R_1}\right)$ 4. Обчислити $h_1 = H\left(M_i\right)$ 5. Обчислити елемент основного поля $y_1 = h_1 x_{R_1} = r_1$ 6.Обчислити $S_1 = (dr_1 + k_1)(mod\, n)$	1. $k_2 = (n - k_1) \in [1, n-1]$ 2. $f_{k_2} = \pi\left((n-k_1)G\right) = \pi(nG - k_1 G) = \pi\left(x_{R_1}, -y_{R_1}\right) = x_{R_1}$ 3. Формується передпідпис $\left(k_2, f_{k_2}\right) = \left(k_2, x_{R_1}\right)$ 4. Обчислити $h_2 = H\left(M_j\right)$ 5. Обчислити елемент основного поля $y_2 = h_2 x_{R_1} = r_2$ 6. Обчислити $S_2 = (dr_2 + k_2)(mod\, n)$

2. Для реалізації атаки зі зв'язуванням сеансових ключів порушник повинен знати загально-системні параметри та перехопити підписані повідомлення у форматах ГОСТ 34.10-2012 чи ДСТУ 4145-2002.

3. Значення сеансового ключа k_1 для ГОСТ34.10-2012 не залежить від довгострокового ключа *d*. Тому обчисливши згідно з (6) значення k_1, в подальшому можна обчислити особистий довгостроковий ключ *d*, використовуючи (5).

4. Для захисту від атак зі зв'язуванням сеансових ключів необхідно та достатньо зашифрувати повідомлення разом з ЕП-значеннями *s* та *r*.

5. Для протидії атакам зі зв'язаними сеансовими ключами цій атаці необхідно використовувати надійні засоби генерування ключів, виключивши таким чином можливість зв'язування сеансових ключів.

6. При удосконаленні ЕП ГОСТ 34.10-2012 та ДСТУ 4145-2002 для протидії атаці зі зв'язуванням сеансових ключів можна замінити функцію вирізання *x* кординати відповідних точок ЕК, наприклад, застосовувати як *x,* так і *y* координати в афінному базисі, а в проективному базисі можна використовувати і значення трьох координат *x*, та *z*.

Література

[1] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Теорія. Практика. Застосування. Монографія / І.Д. Горбенко. – Харків: Форт, 2012. – 878 с.

[2] *Горбенко Ю.І., Горбенко І.Д.* Інфраструктури відкритих ключів . Системи ЕЦП. Теорія та практика. Монографія. – Харків: Форт, 2010. – 593 с.

[3] ГОСТ Р 34.10-2012. Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи. – М.: Госстандарт России, 2012. – 33 с.

[4] ДСТУ 4145-2002 «Інформаційні технології. Криптографічний захист інформації. Цифровий підпис, що грунтується на еліптичних кривих. Формування та перевіряння».

Надійшла до редколегії 20.11.2015

Кулібаба Владислав Андрійович, студент факультету комп'ютерних наук кафедри безпеки інформаційних систем і технологій Харківського національного університету ім. В.Н. Каразіна. Наукові інтереси: дослідження стійкості асиметричних криптографічних систем, можливості застосування криптографічних алгоритмів на мобільних платформах.

Тоцький Олександр Сергійович, начальник відділу ПАТ «Інститут інформаційних технологій». Наукові інтереси: розробка та застосування систем і засобів КЗІ.

Горбенко Іван Дмитрович, фото та відомості про автора див. на с. 295.

УДК 681.3.06

Анализ защищенности алгоритмов электронных подписей в группе точек эллиптической кривой ГОСТ Р 34.10 2012 и ДСТУ 4145 2002 от атаки на связанных ключах / В.А. Кулибаба,Тоцкий А.С., И.Д. Горбенко // Прикладная радиоэлектроника: науч.-техн. журнал. – 2015. – Том 14. – № 4. – С. 296–299.

Предложена атака, которая позволяет определить секретные параметры электронной подписи путем связывания сеансовых ключей. Для двух стандартов приводятся упрощенные модели генерации и проверки подписи, а также модели атаки, которые предусматривают использование для двух разных сообщений так выбранных сеансовых ключей, чтобы подписи были зависимы между собой. Предлагаются два примера, в которых выполняются две электронных подписи с использованием связанных ключей и вычисляются тайные параметры электронной подписи в соответствии с предложенной методикой.

Ключевые слова: электронная подпись, атака на связанных ключах.

Табл.: 4. Библиогр.: 4 назв.

UDC 681.3.06

Analyzing the security of algorithms of electronic signatures in multipoint eleptic curves of State Standard R 34.10 2012 and DSTU 4145 2002 from related-key attacks / V.A. Kulibaba, A.C.Totskiy, I.D. Gorbenko // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 296–299.

An attack is proposed which allows to identify secret parameters by binding electronic signature session keys. The paper provides simplified models of generating and verifying the signatures as well as attack models for two standards which involve the use for two different messages of the session keys chosen in such a way so that signatures should be dependent on each other. Two examples are suggested in which two electronic signatures are performed by using related-keys and secret parameters of an electronic signature are calculated according to the suggested methodology.

Keywords: digital signature, key-related attack.

Tab.: 4. Ref.: 4 items.

УДК 004.056.55

МАТЕМАТИЧНА МОДЕЛЬ ПРОТОКОЛУ СЛІПОГО ЕЛЕКТРОННОГО ПІДПИСУ НА ЕЛІПТИЧНИХ КРИВИХ

М.В. ЄСІНА

У роботі розглядається математична модель протоколу сліпого електронного підпису на основі алгоритму ECDSA та алгоритмі, який описано у ДСТУ 4145-2002. Проводиться перевірка захищеності протоколу на основі цих алгоритмів за критерієм анонімності.

Ключові слова: анонімність, електронний підпис, сліпий підпис.

ВСТУП

Сліпий підпис знаходить широке застосування в протоколах електронного голосування та електронних грошей. В його основу, як правило, покладається електронний підпис (ЕП). В ході застосування сліпого підпису надається послуга анонімності (невідстежуваності), яка є обов'язковою в системах таємного електронного голосування та системах електронних грошей.

У типовій схемі сліпого підпису приймають участь, як правило, три сторони [5] – емітент документа, підписувач та перевірник (валідатор). Емітент створює документ, який підписувач має підписати анонімно. Тобто, підписувач не повинен знати вміст документа та вигляд остаточного підпису. Для цього емітент маскує документ за допомогою певного криптографічного перетворення. Підписувач підписує замаскований документ, а емітент на основі його підпису формує остаточний ЕП під документом у відкритому вигляді. Перевірник перевіряє правильність підпису за допомогою відкритого ключа емітента.

У [5, 6] запропоновані механізми (протоколи) сліпого ЕП, які ґрунтуються на алгоритмах ГОСТ 34.10-2001, Шнора та Ель Гамаля. Але сьогодні в Україні дозволеними чи такими, що рекомендуються до застосування, є алгоритми ЕП, що визначені в ДСТУ ISO/IEC 14888-3:2014 та ДСТУ 4145-2002. Тому важливою є задача розробки та детального дослідження вказаних алгоритмів ЕП з точки зору їх застосування в механізмах сліпого підпису.

Метою цієї статті є розробка пропозицій з побудови та дослідження механізму сліпого підпису на основі використання ЕП, що визначені в ДСТУ ISO/IEC 14888-3:2014 (ECDSA) та ДСТУ 4145-2002.

1. ЗАГАЛЬНИЙ ОПИС ПРОТОКОЛУ СЛІПОГО ЕЛЕКТРОННОГО ЦИФРОВОГО ПІДПИСУ НА ЕЛІПТИЧНИХ КРИВИХ

У схемі приймають участь дві сторони: А – підписувач, В – абонент (емітент документа / повідомлення m). Перевіряючим (валідатором) може виступати будь-хто з них, або довірена третя особа [5].

Загальні параметри:

- просте поле $GF(p)$;
- ЕК над цим полем;
- n – порядок базової точки;
- G – базова точка;
- функція гешування $H()$.

Протокол складається з трьох етапів:

1) генерація ключів,
2) постановка підпису,
3) перевірка підпису.

Під час генерації ключів секретний ключ d обирається випадково з діапазону $1 < d < (n-1)$. З нього формується відкритий ключ Q за відповідною формулою.

Етап постановки підпису починає підписувач А. Він обирає k (в деяких стандартах позначається як e) – випадкове число із діапазону $1 < k < (n-1)$ та обчислює точку E.

Підписувач А відправляє точку E абоненту В.

Абонент В формує геш-образ повідомлення h.

Після цього В обирає маскуючий параметр α, $1 < \alpha < n-1$.

Далі він обчислює точку C.

Абонент В обчислює величини r та r'.

Ці величини використовуються для засліплення геш-образу повідомлення h'.

Абонент В пересилає h' підписувачу А.

Підписувач А ставить під засліпленим геш-образом повідомлення h' засліплений підпис s' за допомогою свого власного секретного ключа d та пересилає отримане значення абоненту В.

Абонент В має можливість перевірити справжність засліпленого підпису s' за допомогою звичайної перевірки підпису, що описана у відповідному стандарті, використовуючи відкритий ключ Q підписувача А.

Якщо s' проходить перевірку, абонент В формує з нього остаточний підпис.

Сліпим підписом під документом m вважається пара значень $<r,s>$.

Перевіряючий (валідатор) під час перевірки підпису $\{m,<r,s>\}$ обчислює точку R, для чого використовує звичайну перевірку підпису, що описана у відповідному стандарті, використовуючи відкритий ключ Q підписувача А.

Підпис вважається справжнім, якщо виконується таке співвідношення: $r = x_R \bmod n$.

2. ПЕРЕВІРКА ЗАХИЩЕНОСТІ ПРОТОКОЛУ ЗА КРИТЕРІЄМ АНОНІМНОСТІ

Для схем сліпого підпису, на відміну від інших різновидів ЕЦП, актуальною є атака порушення анонімності. Спроба атаки може бути

здійснена підписувачем за умови, що він зберігатиме всі відомі йому параметри схеми сліпого підпису разом із ідентифікатором емітента для кожної сесії постановки підпису. Накопичена база даних може бути використана в атаці, яка полягає у спробі визначення автора відомого документа m із підписом $<r,s>$, що проходить перевірку за допомогою відкритого ключа підписувача Q [5].

У запропонованому протоколі атака порушення анонімності може бути здійснена наступним чином. Підписувач А для кожного рядка своєї бази даних повинен обчислити ймовірний засліплюючий параметр α' [6].

За допомогою обчислених параметрів підписувач А для кожного рядка бази даних обчислює точку R'.

Рядок бази даних, для якого виконується наступне співвідношення, має вказати на емітента повідомлення:

$$r = x_{R'} \bmod n .$$

Доведемо, що точка R' завжди збігається з перевірною точкою R і не залежить від параметрів h',r',s' і, отже, не дає можливості визначити автора документа m. Для доведення цього твердження використовується рівність R' стандартній перевірці цифрового підпису у відповідному стандарті.

Розглянуті протоколи вважаються захищеними за критерієм анонімності, тому що неможливо визначити автора документа m.

3. ПРОТОКОЛ СЛІПОГО ПІДПИСУ НА ОСНОВІ ДСТУ ISO/IEC 14888-3:2014 (ECDSA)

У таблицях 1 та 2 наведені параметри протоколу сліпого підпису на основі алгоритму цифрового підпису ECDSA [1].

Доведемо, що наведені формули відповідають дійсності: абонент В формує із сліпого ЕП остаточний підпис:

$$s = \frac{s'\cdot(r/r')}{\alpha} \bmod n . \qquad (1)$$

Перевіряючий (валідатор) під час перевірки підпису $\{m,<r,s>\}$ обчислює точку R, для чого використовує відкритий ключ Q підписувача А:

$$R = (\frac{h}{s}\cdot G + \frac{r}{s}\cdot Q) \bmod n = (x_R, y_R) . \qquad (2)$$

Покажемо, що математичний вираз остаточного підпису s проходить перевірку валідатора:

$$R = (\frac{h}{s}\cdot G + \frac{r}{s}\cdot Q) \bmod n = \frac{dr+h}{s}\cdot G \bmod n =$$
$$= \frac{k\alpha G(dr+h)}{(dr'+h')\cdot r/r'} \bmod n = \frac{k\alpha G(dr+h)}{(dr'+\frac{r'}{r}\cdot h)\cdot r/r'} \bmod n = \qquad (3)$$
$$= \frac{k\alpha G(dr+h)}{dr+h} \bmod n = k\alpha G \bmod n =$$
$$= \alpha E \bmod n = (x_R, y_R) = C = (x_C, y_C).$$

Таблиця 1

Вирази для сліпого та остаточного підпису, та їх перевірки

Параметри	ECDSA
Засліплений підпис	$s' = \frac{d\cdot r' + h'}{k} \bmod n$
Перевірка засліпленого підпису	$R' = (\frac{h'}{s'}\cdot G + \frac{r'}{s'}\cdot Q) \bmod n$
Остаточний підпис	$s = \frac{s'\cdot(r/r')}{\alpha} \bmod n$
Перевірка остаточного підпису	$R = (\frac{h}{s}\cdot G + \frac{r}{s}\cdot Q) \bmod n = (x_R, y_R)$, $r = x_R \bmod n$

Таблиця 2

Параметри протоколу сліпого ЕП та перевірки захищеності протоколу за критерієм анонімності

Параметри	ECDSA
Відкритий ключ	$Q = d\cdot G \bmod n$
Точка E	$E = k\cdot G \bmod n = (x_E, y_E)$
Геш-значення	$h = H(m)$
Точка C	$C = \alpha\cdot E \bmod n = (x_C, y_C)$
Величини r та r'	$r = x_C \bmod n$, $r' = x_E \bmod n$
Засліплений геш-образ	$h' = (\frac{r'}{r}\cdot h) \bmod n$
Параметр для перевірки на анонімність	$\alpha' = \frac{s'\cdot(r/r')}{s} \bmod n$
Перевірка на анонімність	$R' = \alpha'\cdot E \bmod n = (x_{R'}, y_{R'})$ => $R' = (\frac{r}{s}\cdot Q + \frac{h}{s}\cdot G) \bmod n$, $r = x_{R'} \bmod n$

У запропонованому протоколі атака порушення анонімності може бути здійснена наступним чином. Підписувач А для кожного рядка своєї бази даних повинен обчислити ймовірний засліплюючий параметр α':

$$\alpha' = \frac{s'\cdot(r/r')}{s} \bmod n . \qquad (4)$$

За допомогою обчислених параметрів підписувач А для кожного рядка бази даних обчислює точку R' за допомогою такої формули:

$$R' = \alpha'\cdot E \bmod n = (x_{R'}, y_{R'}) . \qquad (5)$$

Доведемо, що точка R' завжди збігається з перевірочною точкою R і не залежить від параметрів h',r',s' і, отже, не дає можливості визначити автора документа m. Для доведення цього твердження використовуватимемо рівність R' стандартній перевірці цифрового підпису ECDSA:

$$R' = \alpha'\cdot E \bmod n = \frac{s'\cdot(r/r')}{s}\cdot E \bmod n =$$
$$= \frac{\frac{dr'+h'}{k}\cdot\frac{r}{r'}}{s}\cdot E \bmod n = \frac{\frac{dr'+\frac{r'}{r}\cdot h}{k}\cdot\frac{r}{r'}}{s}\cdot E \bmod n = \qquad (6)$$

$$= \frac{\frac{dr+h}{k}}{s} \cdot E \bmod n = \frac{dr+h}{ks} \cdot kG \bmod n = \\ = (\frac{dr}{s} \cdot G + \frac{h}{s} \cdot G) \bmod n = (\frac{r}{s} \cdot Q + \frac{h}{s} \cdot G) \bmod n. \quad (6)$$

Отже, розглянутий протокол вважається захищеним за критерієм анонімності, тому що неможливо визначити автора документа m.

4. ПРОТОКОЛ СЛІПОГО ПІДПИСУ НА ОСНОВІ ДСТУ 4145-2002

У таблицях 3 та 4 наведено параметри протоколу сліпого підпису на основі алгоритму цифрового підпису ДСТУ [3].

Таблиця 3

Вирази для сліпого та остаточного підпису, та їх перевірки

Параметри	ДСТУ
Засліплений підпис	$s' = (e + dr') \bmod n$
Перевірка засліпленого підпису	$R' = (s' \cdot G + r' \cdot Q) \bmod n$
Остаточний підпис	$s = s' \cdot \alpha \bmod n$
Перевірка остаточного підпису	$R = (s \cdot G + r \cdot Q) \bmod n = (x_R, y_R)$, $r = x_R \bmod n$

Таблиця 4

Параметри протоколу сліпого ЕП та перевірки захищеності протоколу за критерієм анонімності

Параметри	ДСТУ
Відкритий ключ	$Q = -d \cdot G \bmod n$
Точка E	$E = k \cdot G \bmod n = (x_E, y_E)$
Геш-значення	$h = H(m)$
Точка C	$C = \alpha \cdot E \bmod n = (x_C, y_C)$
Величини r та r'	$r = h \cdot x_C \bmod n$, $r' = h' \cdot x_E \bmod n$
Засліплений геш-образ	$h' = \frac{x_C \cdot h}{x_E \cdot \alpha} \bmod n$
Параметр для перевірки на анонімність	$\alpha' = \frac{s}{s'} \bmod n$, $\alpha' = \frac{x_C \cdot h}{x_E \cdot h'} \bmod n = \frac{r}{r'} \bmod n$
Перевірка на анонімність	$R' = \alpha' \cdot E \bmod n = (x_{R'}, y_{R'}) \Rightarrow R' = (s \cdot G + r \cdot Q) \bmod n$, $r = x_{R'} \bmod n$

Доведемо, що наведені формули відповідають дійсності: абонент В формує із сліпого ЕП остаточний підпис:

$$s = s' \cdot \alpha \bmod n. \quad (7)$$

Перевіряючий (валідатор) під час перевірки підпису $\{m, <r, s>\}$ обчислює точку R, для чого використовує відкритий ключ Q підписувача А:

$$R = (s \cdot G + r \cdot Q) \bmod n = (x_R, y_R). \quad (8)$$

Покажемо, що математичний вираз остаточного підпису s проходить перевірку валідатора:

$$R = (s \cdot G + r \cdot Q) \bmod n = \\ = ((e + dr')\alpha G + rQ) \bmod n = \\ = (\alpha e G + \alpha r' dG + rQ) \bmod n = \\ = (\alpha E + \alpha h' x_E dG + rQ) \bmod n = \\ = (\alpha E + rQ + \alpha Q x_E \frac{x_C \cdot h}{x_E \cdot \alpha}) \bmod n = \\ = (\alpha E + rQ + x_C h Q) \bmod n = \\ = (\alpha E - rdG + rQ) \bmod n = \\ = (\alpha E - rdG + rdG) \bmod n = \\ = \alpha E \bmod n = (x_R, y_R) = C = (x_C, y_C). \quad (9)$$

У запропонованому протоколі атака порушення анонімності може бути здійснена наступним чином. Підписувач А для кожного рядка своєї бази даних повинен обчислити ймовірний засліплюючий параметр α':

– виходячи з формули остаточного підпису

$$\alpha' = \frac{s}{s'} \bmod n. \quad (10)$$

– виходячи з формули засліпленого геш-значення

$$\alpha' = \frac{x_C \cdot h}{x_E \cdot h'} \bmod n = \frac{r}{r'} \bmod n. \quad (11)$$

За допомогою обчислених параметрів підписувач А для кожного рядка бази даних обчислює точку R' за допомогою такої формули:

$$R' = \alpha' \cdot E \bmod n = (x_{R'}, y_{R'}). \quad (12)$$

Доведемо, що точка R' завжди збігається з перевірною точкою R і не залежить від параметрів h', r', s' і, отже, не дає можливості визначити автора документа m. Для доведення цього твердження використовуватимемо рівність R' стандартній перевірці цифрового підпису ДСТУ:

$$R' = \alpha' \cdot E \bmod n = eG \frac{r}{r'} \bmod n = \\ = G \frac{r}{r'} (s' - dr') \bmod n = \\ = (G \frac{r}{r'} s' - G \frac{r}{r'} s \cdot dr') \bmod n = \\ = (G \frac{r}{r'} s' - rdG) \bmod n = \\ = (\alpha s' G - rdG) \bmod n = (s \cdot G + r \cdot Q) \bmod n. \quad (13)$$

Отже, розглянутий протокол вважається захищеним за критерієм анонімності, тому що неможливо визначити автора документа m.

5. АТАКИ НА ПРОТОКОЛ СЛІПОГО ПІДПИСУ

На сліпий підпис існують різноманітні атаки. Ці атаки такі ж самі як і атаки на звичайні ЕП. Це досягається за допомогою того, що у нашому випадку протокол сліпого підпису побудовано на основі відповідних стандартів алгоритмів ЕП. Алгоритм формування сліпого підпису співпадає зі звичайним алгоритмом побудови ЕП відповідного стандарту, а формування остаточного

підпису у протоколі використовує раніше сформований сліпий підпис, який множиться або ділиться на випадкове число. Жодних інших операцій не використовується.

Зважаючи на все вищевказане, можна стверджувати, що сліпому та остаточному підпису у протоколі сліпого підпису загрожують аналогічні атаки.

5.1. Атака «повне розкриття» на основі підписаних даних

Стійкість двох розглянутих вище алгоритмів ЕП заснована на складності розв'язання дискретного логарифму в групі точок еліптичної кривої. Для знаходження секретного ключа необхідно розв'язати відносно d рівняння на основі обчислення відкритого ключа Q, що індивідуальні для кожного з розглянутих алгоритмів [4].

Як було вказано вище, маємо однакові атаки на стандартні алгоритми ЕП та на протоколи сліпого підпису, на основі цих алгоритмів. Отже, можна розглядати атаку «повне розкриття» для протоколів сліпого підпису таким же чином, як і для стандартних алгоритмів ЕП.

Розглянемо можливість знаходження d на основі атаки при відомих підписаних (перехоплених) повідомленнях. Нехай перехоплено і підписано повідомлення [3].

Сліпий підпис для ECDSA має такий вигляд

$$s' = \frac{d \cdot r' + h'}{k} \bmod n \qquad (14)$$

тобто, відносно d, отримаємо таке:

$$\begin{cases} d = \dfrac{k_1 s'_1 - h'_1}{r'_1} \bmod n \\ \ldots \\ d = \dfrac{k_i s'_i - h'_i}{r'_i} \bmod n \end{cases}. \qquad (15)$$

Сліпий підпис для ДСТУ 4145-2002 має такий вигляд

$$s' = (e + dr') \bmod n \qquad (16)$$

тобто, відносно d, отримаємо:

$$\begin{cases} d = \dfrac{s'_1 - e'_1}{r'_1} \bmod n \\ \ldots \\ d = \dfrac{s'_i - e'_i}{r'_i} \bmod n \end{cases}. \qquad (17)$$

Отже, у випадку сліпого підпису також маємо систему рівнянь порядку i рівнянь з $i+1$ невідомими. Тобто не маємо жодної різниці зі стандартним алгоритмом.

Тепер розглянемо ситуацію з остаточним підписом у протоколі сліпого підпису.

У ході формування остаточного сліпого підпису використовується раніше сформований засліплений підпис, який лише множиться або ділиться на визначене випадкове число. Жодних інших дій при формуванні остаточного підпису не проводиться.

Остаточний підпис у ході використання ECDSA має такий вигляд

$$s = \frac{s' \cdot (r / r')}{\alpha} \bmod n, \qquad (18)$$

де α – випадкове число із визначеного діапазону, $\frac{r}{r'}$ – відношення двох координат (також просто число).

Розпишемо s повністю та одержимо відносно d необхідний вираз

$$s = \frac{(e + dr') \cdot \frac{r}{r'}}{\alpha} \bmod n = \frac{e \cdot \frac{r}{r'} + dr}{\alpha} \bmod n. \qquad (19)$$

$$\begin{cases} d = \dfrac{s_1 \alpha - e_1 \frac{r_1}{r'_1}}{r_1} \bmod n \\ \ldots \\ d = \dfrac{s_i \alpha - e_i \frac{r_i}{r'_i}}{r_i} \bmod n \end{cases}. \qquad (20)$$

Остаточний підпис у ході використання ДСТУ 4145-2002 має такий вигляд

$$s = s' \cdot \alpha \bmod n, \qquad (21)$$

де α – випадкове число із визначеного діапазону.

Розпишемо s повністю та одержимо відносно d необхідний вираз

$$s = (e + dr')\alpha \bmod n = (e\alpha + dr'\alpha) d \bmod n \qquad (22)$$

$$\begin{cases} d = \dfrac{s_1 - e_1 \alpha}{r'_1 \alpha} \bmod n \\ \ldots \\ d = \dfrac{s_i - e_i \alpha}{r'_i \alpha} \bmod n \end{cases}. \qquad (23)$$

Як видно з наведених виразів для усіх розглянутих алгоритмів ЕП, сформований остаточний підпис у протоколі сліпого підпису має той самий вигляд, що і стандартний ЕП. Відмінність лише у наявності відповідних коефіцієнтів при основних значеннях. Але ці коефіцієнти – випадкові числа і жодним чином не впливають на стійкість до атаки.

Таким чином, для повного розкриття, тобто визначення секретного ключа d за i отриманим ЕП, необхідно розв'язувати систему i-го порядку з $i+1$ невідомими [4].

У разі, якщо повідомлення M є зашифрованим, невідомими є значення функцій гешування $h_1, h_2, \ldots, h_i$. Як результат одержимо систему рівнянь з $2i+1$ невідомими, тому шифрування підписаних повідомлень дозволяє істотно підвищити стійкість.

5.2. Аналіз захищеності ЕП від атак на реалізацію

Нехай розробник може закласти лазівку у програмну реалізацію вироблення підпису. Нижче наводяться теоретичні та експериментальні результати відносно захищеності сліпого підпису у протоколах сліпого ЕП на основі стан-

дартних алгоритмів ЕП від таких атак для двох розглянутих вище алгоритмів [4].

Для ECDSA порушнику відомо:

$$r'_1 = \pi(x_1, y_1) = x_1 \bmod n, \quad (24)$$

$$r'_2 = \pi(x_1, -y_1) = x_1 \bmod n. \quad (25)$$

Таким чином

$$r'_1 = r'_2 = x_1 \bmod n. \quad (26)$$

Для s'_1 та s'_2 маємо:

$$s'_1 = \frac{dr'_1 + h'_1}{k_1} \bmod n, \quad (27)$$

$$s'_2 = \frac{dr'_2 + h'_2}{k_1} \bmod n. \quad (28)$$

Оскільки $k_1 = k_2 = k$ та $r'_1 = r'_2 = x_1 \bmod n$, то

$$s'_1 = \frac{dr' + h'_1}{k_1} \bmod n, \quad (29)$$

$$s'_2 = \frac{dr' + h'_2}{k_1} \bmod n. \quad (30)$$

Розв'язавши відносно d та k маємо:

$$d = \frac{s'_1 h'_2 - s'_2 h'_2}{r'(s'_2 - s'_1)} \bmod n; \quad (31)$$

$$k = \frac{dr' + h'_1}{s'_1} \bmod n. \quad (32)$$

Для ДСТУ 4145-2002 пропонуємо:

$$k_1 = k_2 = k \in (1, n-1); \quad (33)$$

$$R_1 = R_2 = kG = (x_R, y_R) = R; \quad (34)$$

$$fk_1 = fk_2 = x_R = fk; \quad (35)$$

$$y_1 = h'_1 fk; \quad (36)$$

$$y_2 = h'_2 fk; \quad (37)$$

$$y_1 \Rightarrow r'_1; y_2 \Rightarrow r'_2; \quad (38)$$

Для s'_1 та s'_2 маємо:

$$s'_1 = (k + dr'_1) \bmod n; \quad (39)$$

$$s'_2 = (k + dr'_2) \bmod n. \quad (40)$$

Розв'язуючи (39) та (40) відносно (k,d) маємо:

$$d = \frac{s'_1 - s'_2}{r'_1 - r'_2} \bmod n, \quad (41)$$

$$k = (s'_1 - dr'_1) \bmod n. \quad (42)$$

Для протоколів сліпого підпису на основі алгоритмів ЕП ECDSA та ДСТУ 4145-2002 існують атаки на програмну реалізацію ЕП. Якщо зловмиснику вдасться змусити програму «вироблення підпису» двічі використати одне і те саме значення k для двох повідомлень, то він в реальному часі визначить особистий довгостроковий ключ d і зможе нав'язувати як хибні повідомлення, так і викривляти істинні.

Для захисту від такої атаки необхідно використовувати надійні засоби КЗІ типу ЕП у змісті наявності на них сертифікатів відповідності, експертних висновків та можливості безперервного контролю цілісності та справжності програми вироблення ЕП. Кращим способом забезпечення захисту від такої загрози є апаратна реалізація процедур вироблення та перевіряння ЕП [4].

5.3. Аналіз захищеності сліпого ЦП від атаки на анонімність

Як вказано вище, усі алгоритми проходять перевірку на анонімність і, навіть, якщо підписувач A зберігатиме усі параметри h', r', s', то в подальшому він не зможе встановити відповідність цих параметрів до емітента, для якого підпис було виконано. Але, якщо для алгоритму ECDSA це правильно повною мірою, то для інших є особливість – α' виражається двома виразами, які прийматимуть однакове значення лише для абонента B, що формував остаточний підпис за цими параметрами (ймовірність того, що буде ще один емітент, для якого два вирази α' матимуть однакове значення, дорівнює 2^{-n}). З цього випливає, що наведений механізм для EKGDSA, ECKCDSA та ДСТУ забезпечує сліпий підпис з відстежуваною анонімністю [2].

Змінити це можливо за допомогою апаратних чи апаратно-програмних засобів криптографічного захисту інформації (КЗІ). Використання таких засобів для сліпого підпису подібно використанню криптографічних модулів для генерації ключів користувачів у центрах сертифікації ключів (ЦСК). Користувач може згенерувати свій ключ на станції в самому центрі, але завдяки тому, що було використано сертифікований засіб КЗІ, користувач може бути впевнений в тому, що тільки він володіє ключем і у ЦСК не залишилися копії цього ключа.

Таким же чином можливо використання криптографічного модуля для сліпого підпису. Нехай буде мікромодуль D, в якому буде записана асиметрична пара ключів для виконання підпису та забезпечення конфіденційності при отриманні заслiпленого геш-образу. В такому випадку підписувач A – лише оператор криптомодуля D, який не має прямого доступу до ключів (або D може повністю заміщати A, тоді емітенту В надається доступ до роботи із засобом КЗІ). Виконуються такі операції:

1) абонент В направлено зашифровує h' на відкритому ключі криптомодуля D;

2) отримане $E_D(h')$ надсилається D (прямо чи за допомогою оператора A);

3) D розшифровує h' та створює s';

4) r' та s' відсилаються емітенту B, а h' вилучається з пам'яті D.

Через те, що h' обробляється лише у D і A немає можливості розшифрувати $E_D(h')$, підписувач не зможе здійснити атаку на анонімність, бо в нього не буде одного з параметрів.

Такий підхід може використовуватися при наданні послуг сліпого ЕП у хмарах. Також одним з прикладів використання сліпого підпису на основі засобів КЗІ є проведення виборів. Виборці заходять до кабінки, де стоїть автома-

тизована станція (АС), що оснащена криптографічним модулем; формують список голосів за кандидатів (повідомлення); виробляють підпис та надсилають його. За допомогою механізму сліпого підпису і використання запрограмованого на це засобу КЗІ, забезпечується анонімність голосування і підтверджується дійсність та цілісність кожного голосу.

ВИСНОВКИ

Розглянувши алгоритми ЕП, що базуються на стандартах ДСТУ ISO/IEC 14888-3:2014 (ECDSA) та ДСТУ 4145-2002 [1, 3], було побудовано на їх основі протокол сліпого електронного підпису. Дослідивши схеми даних сліпих електронних підписів, було визначено, що ці схеми стійкі за критерієм анонімності. Також дослідження цих схем показує, що співвідношення між маскуючими параметрами необхідно обирати таким чином, щоб за ними було неможливо вирахувати автора документу, що ставить підпис під документом [5, 6].

Перевагою запропонованого методу сліпого підпису над існуючими є те, що дії підписувача та валідатора такі самі, як описано у відповідних стандартах для звичайного підпису та перевірки в групі точок еліптичних кривих. Відмінність полягає лише в тому, що підписувач отримує геш-значення, а не знаходить його сам. Кроки, що відрізняють сліпий підпис від звичайного, виконуються емітентом. Така методика робить впровадження функціоналу сліпого підпису в існуючі інформаційно-телекомунікаційні системи таким, що майже не потребує додаткових зусиль. Необхідно лише реалізувати протокол для емітента, а підписувач та валідатор можуть використовувати вже існуючі засоби створення та перевірки ЕП.

Також даний метод має переваги під час розробки стандартів. Такий підхід дозволяє напряму посилатися на існуючі стандарти і не вступати з ними в протиріччя (перевірка підпису за одним стандартом, як для ЕП так і для сліпого підпису).

Через те, що алгоритми сліпого підпису у протоколі сліпого ЕП співпадають з алгоритмами ЕП відповідних стандартів, то алгоритми сліпого підпису є уразливими до тих самих атак, що і стандартні алгоритми ЕП. При формуванні остаточного підпису стандартний алгоритм ЕП також зберігається, тому що остаточний підпис формується зі сліпого, який множиться або ділиться на випадкове число, яке жодним чином не впливає на стійкість до атак.

В ході використання даного механізму ECDSA забезпечує повну анонімність, а ДСТУ 4145 – лише відстежувану. Для забезпечення повної анонімності необхідно використовувати засоби КЗІ. Іншим варіантом є переробка самого механізму, але це призведе до втрати усіх його переваг і може лише в подальшому слугувати альтернативою використанню апаратних криптографічних модулів.

Література

[1] Information technology – Security techniques – Digital signatures with appendix – Part 3: Discrete logarithm based mechanisms : ISO/IEC 14888-3 (Edition 2 (2006-11-15)) : 2006. – 68 p.

[2] Information technology – Security techniques – Blind digital signatures – Part 2: Discrete logarithm based mechanisms : ISO/IEC DIS 18370-2:2014(E):2015. – 70 p.

[3] Інформаційні технології – Криптографічний захист інформації – Цифровий підпис, що ґрунтується на еліптичних кривих – Формування та перевірка : ДСТУ 4145-2002. – К. : Держстандарт України, 2003. – 35 с. – (Національні стандарти України).

[4] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Теорія. Практика. Застосування: монографія. Харків: «Форт», 2012. – 870 с.

[5] *Нікуліщев Г. І.* Протокол сліпого електронного цифрового підпису на еліптичних кривих над скінченим векторним полем / Г. І. Нікуліщев // Радіоелектроніка, інформатика, управління. – 2013, № 2. – С. 71–76.

[6] *Нікуліщев Г. І.* Анонімність як критерій оцінки захищеності протоколів сліпого електронного цифрового підпису / Г. І. Нікуліщев, Г. Л. Козина // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. – 2012, № 2. – С. 59–65.

Надійшла до редколегії 24.11.2015

Єсіна Марина Віталіївна, аспірантка факультету комп'ютерних наук, кафедри безпеки інформаційних систем і технологій Харківського національного університету імені В.Н. Каразіна. Наукові інтереси: електронний підпис, методи багатофакторної автентифікації та їх застосування з метою захисту інформації.

УДК 004.056.55

Математическая модель протокола слепой электронной подписи на элиптических кривых / Есина М.В. // Прикладная радиоэлектроника: науч.-техн. журнал. – 2015. – Том 14. – № 4. – С. 300–305.

В работе рассматривается математическая модель протокола слепой электронной подписи на основе алгоритма ECDSA и алгоритма, который описан в ДСТУ 4145-2002. Проводится проверка защищенности протокола на основе этих алгоритмов по критерию анонимности.

Ключевые слова: анонимность, электронная подпись, слепая подпись.

Табл.: 4. Библиогр.: 6 назв.

UDC 004.056.55

Mathematical model of a protocol of blind electronic signature based on eliptic curves / M.V. Yesina // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 300–305.

The paper deals with a mathematical model of a protocol of electronic signature based on the algorithm ECDSA and algorithm that is described in DSTU 4145-2002. Testing the protocol security based on these algorithms by the criterion of anonymity has been performed.

Keywords: anonymity, digital signature, blind signature.

Tab.: 4. Ref.: 6 items.

УДК 681.3.06

АНАЛІЗ ЕЛЕКТРОННИХ ПІДПИСІВ З ВІДНОВЛЕННЯМ ПОВІДОМЛЕННЯ ЗГІДНО З ДСТУ ISO/IEC 9796-2

Н.В.КОВАЛЬОВА, Ю.І. ГОРБЕНКО, О.О. БАКЛИКОВ

Розглядаються сутності, вимоги, стан стандартизації, моделі та властивості електронних підписів з відновленням повідомленням згідно з ДСТУ ISO/IEC 9796-2, наводяться результати аналізу стійкості до існуючих атак. Формулюються пропозиції відносно застосування стандарту для забезпечення цілісності, справжності та та встановлення неспростовності відправника.

Ключові слова: електронний (цифровий) підпис з відновленням повідомлення, екзистенційна підробка, криптоатаки, механізми електронного підпису, методи факторизації, селективна підробка, повне розкриття.

ВСТУП

Сьогодні розроблено та застосовується значна кількість стандартів на електронні підписи (ЕП). Історично вони з'явились і ґрунтуються на криптографічних перетвореннях з використанням математичного апарату кілець, скінченних полів, еліптичних кривих тощо [1, 3]. Основними вимогами, які висуваються до ЕП, є їх криптографічна стійкість до існуючих та потенційних атак, а також часова та просторова складності криптографічних перетворень. З метою покращення властивостей алгоритмів ЕП згідно з ISO/IEC 9796-2 [3, 11] на міжнародному рівні зроблені корегування, після яких в Україні прийнято удосконалений стандарт ЕП з відновленням повідомлення ДСТУ ISO/IEC 9796-2:2014. Криптографічна стійкість трьох механізмів (алгоритмів) цього стандарту базується на складності факторизації [1–4] та стійкості геш-функцій. Попередній аналіз показав, що ЕП згідно з ISO/IEC 9796-2 [3] сьогодні застосовуються в Skype, а також для активної автентифікації у машино зчитувальних проїзних документах [6] тощо. Основною перевагою, що забезпечується в ході використання ЕП ISO/IEC 9796-2, є можливість відновлення повідомлення, а також те, що на відміну від ЕП з доповненням, наприклад, стандарт ДСТУ ISO/IEC 14888-3:2014, додаткове геш-значення в ISO/IEC 9796-2 не використовується, що дозволяє скоротити довжину геш-значення та і ЕП. Водночас у зв'язку з розвитком систем та засобів криптоаналізу, досягненням суттєвих результатів, особливо пов'язаних з можливою появою квантових комп'ютерів [1], на наш погляд необхідно провести додаткові дослідження як в частині криптографічної стійкості, так і відносно складності безпосередніх криптографічних перетворень.

Метою цієї статі є розгляд вимог, сутності, стану стандартизації та аналіз загальних властивостей, а також аналізу криптографічної стійкості механізмів ЕП згідно з ДСТУ ISO/IEC 9796-2:2014, до існуючих та потенційних атак, забезпечення цілісності, справжності та встановлення неспростовності відправника. Вказана мета досягається на основі розгляду третьої версії стандарту ISO/IEC 9796-2010-12-15.

1. ВИМОГИ

У стандарті ISO/IEC 9796 2010-12-15[3] наведено три механізми ЕП, але як основний, до використання рекомендований другий алгоритм електроннного підпису. При чому, якщо механізм не забезпечує генерацію підписувачем випадкових послідовностей, то рекомендується використати третій алгоритм.

Основними вимогами, які враховані відносно алгоритмів стандарту, є такі [3, 4]:

1. Повідомлення M, що підписується, має бути бінарним рядком будь-якої довжини, можливо, пустим.

2. При електронному підписі має використовуватись особистий ключ підписувача, а під час перевірки відповідний особистому відкритий ключ перевірки.

3. Кожен підписувач повинен використовувати та тримати особистий ключ у секреті, а також забезпечувати його цілісність, справжність та доступність до нього.

4. Кожен перевірник повинен мати доступ до відкритого ключа та мають забезпечуватися його цілісність, неспростовність та справжність, [1–5].

5. В ході використання кожного із механізмів ЕП ISO/IEC 9796-2 потрібно застосовувати стійку до колізії геш-функцію h, наприклад, що визначена у ISO/IEC 10118.

6. Обов'язково має бути зв'язок між алгоритмом ЕП та геш-функцією, що використовується. В ході використання слабкої геш-функції з'являється можливість підробки ЕП.

7. Перевірник ЕП повинен завжди мати незалежний захист засобів визначення, який з трьох алгоритмів підпису використовується при здійсненні ЕП. Так, якщо використовується алгоритм ЕП 2 чи 3, то він повинен мати засоби визначення, яка з двох функцій ЕП застосовується, наприклад, засобом зазначення ідентифікатора алгоритму ЕП.

8. Кожен з механізмів ЕП може мати свої параметри, діапазон можливих варіантів їх зміни, відносно яких перевірник повинен мати інформацію.

9. Для усіх трьох механізмів ЕП перевірник повинен знати який параметр додаткового поля використовується (1 чи 2).

10. Якщо використуваються механізми ЕП 2 чи 3, то перевірник повинен знати довжину випадкової компоненти (солі) S. Орієнтуючись на наведені вимоги, проведемо дослідження ISO/IEC 2010-12-15.

2. МОДЕЛІ МЕХАНІЗМІВ ЕП ТА ПЕРЕВІРКИ ПІДПИСУ

Аналіз показує, що якщо повідомлення M достатньо коротке, то відновлення повідомлення може бути повним, оскільки є можливість, що M буде повністю включено у підпис. Якщо повідомлення M довше, то відновлення повідомлення буде частковим. За такої умови повідомлення M має бути поділено на *відновлювану частину*, що є рядком бітів обмеженої довжини, що включається у підпис, та *не відновлювану частину*, що є рядком октетів будь-якої довжини, що зберігається і/або передається разом із підписом.

Механізм ЕП поділяється на три складові процедури – електронного підпису, перевірки підпису і додаткових перетворень, які необхідні для реалізації механізму ЕП [3, 11].

2.1. Перший механізм ЕП

Цей механізм забезпечує детермінований ЕП**.** Механізм може використовуватися тільки у таких середовищах, у яких операційні обмеження забезпечують неможливість отримання підпису порушником (криптоаналітиком) на великій кількості вибраних повідомлень. Механізм рекомендується застосовувати тільки в середовищах, де вимагається сумісність із системами, що реалізують перше видання цієї частини стандарту ISO/IEC 9796. Також механізм ЕП 1 може застосовуватися тільки у системах, які використовують геш-значення з довжиною не менше 160 бітів.

Параметри 1 *механізму ЕП.*

1. Особистий ключ підпису, що використовується, має модуль довжиною k бітів. Він визначає значення довжини ЕП c і довжину послідовності повідомлення F.

Довжина повідомлення ЕП c для 1 механізму визначається як $c = k - L_h - 8t - 4$.

2. Додаткове поле механізму, яке використовується як частина побудови послідовності повідомлення, може мати довжину 1 чи 2 октети, причому найбільший правий напів байт має завжди дорівнювати значенню ‘C’ у шістнадцятковій системі. Також допускаються два таких варіанти:

– при $t = 1$ додаткове поле складається з одного октету; який має дорівнювати значенню ‘BC’ у шістнадцятковій системі;

– при $t = 2$ додаткове поле має складатися з двох послідовних октетів; найбільш правий октет має дорівнювати значенню ‘CC’, а найбільш лівий має бути ідентифікатором геш-функції, що використовується.

Підпис повідомлення. Підпис повідомлення здійснюється за три кроки – розміщення повідомлення, створення відновлюваного рядка і вироблення ЕП.

Розміщення повідомлення. При розміщенні повідомлення M ділиться на дві частини – відновлювану частину M_1 і не відновлювану частину M_2, яка може бути пустою. Довжина відновлюваної частини обмежується зверху величиною c. При чому відновлювана частина відновлюється з підпису в процесі перевірки, а не відновлювана частина має бути доступною перевірнику іншими засобами, наприклад, вона може бути надіслана або збережена із підписом. Отже, якщо повідомлення достатньо коротке, повне повідомлення може бути розміщене у відновлювану частину, а невідновлювана частина буде пустою.

Вибір схеми підпису і ключа. Визначається довжиною c підпису, причому $c \geq 7$. Повідомлення M, що підписується, має бути розділено на дві частини M_1 і M_2.

Довжина відновлюваного повідомлення c^* має бути обрана так, що $c^* \leq c$, $c^* \leq |M|$, і $c^* \equiv |M| \pmod 8$. Причому для механізму ЕП 1 c^* має дорівнювати мінімальному значенню з $c-\Delta$ і $|M|$, де $\Delta = (c-|M|) \bmod 8$.

Якщо довжина $|M| = c^*$, то все повідомлення буде відновлюваним, тобто $M_1 = M$ і M_2 є пустим.

Якщо довжина $|M| > c^*$, то M_1 має дорівнювати c^* лівим бітам M, і M_2 має дорівнювати решті M, тобто M_2 містить $|M|-c^*$ бітів.

Зрозуміло, що завжди $M = M_1 \| M_2$.

Таким чином, при розміщенні повідомлення значення M_2 завжди є довжиною цілого числа октетів. Крім того, обрання c^* як мінімального значення з $c-\Delta$ і $|M|$, де $\Delta = (c-|M|) \bmod 8$, забезпечує, що M_1 є настільки довгим, наскільки можливо при цьому обмеженні. Також, якщо значення M є довжиною цілого числа октетів, тобто якщо $|M|$ є цілою складовою 8, то обидва значення M_1 і M_2 складатимуться з цілого числа октетів.

Створення послідовності повідомлення. На цьому кроці із відновлюваної й не відновлюваної частин повідомлення M_1 і M_2 формується форматований рядок, який називається послідовністю повідомлення F, що є вхідними даними до кроку вироблення підпису.

Безпосередньо створення послідовності повідомлення здійснюється за два роки:

– *Гешування повідомлення.* На цьому кроці повідомлення $M = M_1 \| M_2$ подається на вхід геш-функції h для обчислення $h = H(M)$ відповідної довжини (див. табл. 1).

Таблиця 1

Гешування повідомлення

Біти	0/1 ...	0/1	...	0/1
Довжина (бітів)	L_h			

Форматування повідомлення.

Рядок з k бітів має бути побудований таким чином (зліва направо):

– два біти із значенням ‘01’,

– один біт із значенням ‘0’ у випадку повного відновлення, тобто, коли ($M = M_1$) і ‘1’ у випадку часткового відновлення, тобто коли $|M_2| > 0$;

– ($k - L_h - |M_1| - 8t - 4$) бітів заповнення із значенням ‘0’, де k – бітова довжина особистого ключа підпису, L_h – бітова довжина геш -значення, t – число октет у полі показника;

– один біт із значенням ‘1’ (останній біт заповнення);

– довжина $|M_1|$ біт повідомлення M_1;

– L_h бітів геш-значення h;

– $8t$ бітів додаткового поля T (див. табл. 2).

Створення підпису. На цьому кроці для підпису на вхід подається послідовність повідомлення F й особистий ключ підпису d, внаслідок виробляється *підпис* Σ. При підписі використовуються відкриті ключі згідно з додатком В стандарту ISO/IEC 9796-2. Причому, послідовність повідомлення F отримується обробкою верхнього рядка зліва направо блоками з чотирьох послідовних біт, тобто напів байтів, такими кроками:

1) найбільш лівий напівбайт має залишитися незміненим;

2) якщо значення найбільш правого біта найбільш лівого напівбайта ‘0’, то:

– кожний наступний напівбайт зі значенням ‘0000’, якщо такі є, має бути замінений на напівбайт зі значенням ‘B’ у шістнадцятковій системі; він є частиною поля заповнення;

– із першим наступний напівбайтом зі значенням, яке не дорівнює ‘0000’ і значенням ‘B’ у шістнадцятковій системі (тобто‘1011’) виконується операція «виключне або»; це є напівбайтом, що містить останній біт заповнення.

3) усі наступні біти мають залишитися незмінними. Тобто, якщо значення найбільш правого біта найбільш лівого напів байту‘1’ (а отже нема бітів заповнення зі значенням ‘0’), то не проводяться ніякі зміни в рядку бітів (див. табл. 3).

4) перший біт отриманого рядка, значення якого завжди дорівнює ‘0’, має бути видаленим, що дає рядок F довжиною k-1 бітів.

В результаті рядок F має такий вигляд

$F = 01\|$Біти заповнення$\| M_1 \| h(M_1 \| M_2) \| T$.

У випадках, коли застосовується тільки часткове відновлення, значення $M2$ є настільки коротким, наскільки можливо, причому при обмеженні, що воно є цілим числом октетів.

Перевірка підпису. Підписане повідомлення складається або тільки з підпису Σ, у випадку повного відновлення, або з не відновлюваної частини повідомлення M_2* разом з підписом Σ, у випадку часткового відновлення.

При перевірянні підпису Σ і не відновлюваної частини повідомлення M_2*, необхідно виконати три кроки: розкриття підпису, відновлення повідомлення і складання повідомлення.

Розкриття підпису. При розкритті ЕП на вхід процедури подається ЕП Σ і відкритий ключ перевірки. Виходом є – при успішній перевірці послідовність відновленого повідомлення F*, а при неуспішній – помилка.

Відновлення повідомлення.

На цьому кроці на вхід подається послідовність відновленого повідомлення F* і не відновлювану частину повідомлення M_2*. Виходом є відновлювана частина повідомлення M_1* а при неуспішній – помилка, тобто перевірка не виконана.

Під час перевірки потрібно використати потрібну функцію гешування та довжину геш-значення Lh.

Якщо значення найбільш правого октету послідовності відновлюваного повідомлення F* дорівнює у шістнадцятковій системі ‘BC’, то додаткове поле складається з одного октету, а якщо із ‘CC’, то додаткове поле складається з двох найбільш правих октетів F*, де найбільш лівий октет є ідентифікатором геш-функції, яка використовується. Якщо ідентифікатор геш-функції не співпадає, то перевірка підпису не здійснюється.

Підпис Σ також має бути відхиленим, якщо значення найбільш лівого біту послідовності відновлюваного повідомлення ‘0’.

При успішній перевірці один біт зі значенням ‘0’ має бути приєднаним до лівого кінця рядку, так щоби він складався з k бітів. (див. табл. 4).

Далі рядок має бути обробленим блоками з чотирьох послідовних бітів зліва направо, тобто напівбайтами таким чином:

1. Найбільш лівий напівбайт має залишитися незміненим.

Таблиця 2

Форматування повідомлення

Біти	0	1	0/1	0	0	...	0	1	0/1	0/1	...	0/1	0/1	0/1	...	0/1	0/1	0/1	...	0/1
Блоки				Біти заповнення					M_1				h				T			
Довжина блоку (бітів)	1	1		$k - L_h - \|M_1\| - 8t - 4$				1	$\|M_1\|$				L_h				$8t$			
Загальна довжина (бітів)	k																			

Таблиця 3

Створення підпису

Напів байти	4 ... 7	B	B	...	B	0...F	0...F	...	0...F
Блоки		Біти заповнення				M_1, H, T			
Загальна довжина (бітів)	k								

Таблиця 4

Відновлення повідомлення

Напів-байти	4...7	B	B	...	B	0...*F*	0...*F*	...	0...*F*
Блоки		Біти заповнення				M_1, *H*, *T*			
Загальна довжина (бітів)	*k*								

2. Якщо значення найбільш лівого напівбайта ‘0’, то:

а) кожний наступний напівбайт зі значенням ‘B’ у шістнадцятковій системі, якщо такі є, є частиною поля заповнення;

б) для відновлення початкового значення цього напівбайта виконується операція «виключне або» першого наступного напівбайта зі значенням, яке не дорівнює ‘B’ і значенням ‘B’ у шістнадцятковій системі.

3. Всі наступні біти мають залишитися незмінними (див. табл. 5).

Після цього може бути визначено знаходження останнього найбільш правого біта заповнення і обчислена кількість бітів заповнення. Також для визначення того, чи забезпечує підпис часткове або повне відновлення, може бути оброблений третій біт першого напівбайта. Якщо присутні дев’ять або більше бітів заповнення, тобто вісім або більше бітів заповнення, ЕП має бути відхилено. Інакше, процес перевірки продовжується.

Всі біти до кінця поля заповнення зліва в модифікованої версії мають бути видалені, й одночи двохоктетне поле справа має бути видаленим. Рядок, що залишився, має бути поділений на дві частини. Відновлене геш-значення H^* має складатися з Lh найбільш правих бітів. Відновлена частина повідомлення $M1^*$ має складатися з бітів, що залишились зліва.

Також відновлена частина повідомлення M_1^* має бути приєднана до невідновлюваної частини повідомлення M_2^*. Якщо результат гешування співпадає з H^*, тобто якщо

$$H^* = h(M_1^* \| M_2^*),$$

то підпис правильний, а отже приймається M_1^*; інакше, підпис має бути відхиленим.

Складання повідомлення. На цьому кроці повідомлення M^* відтворюється з відновлюваної частини M_1^* і не відновлюваної частини M_2^*, тобто $M^* = M_1^* \| M_2^*$.

2.2. Другий механізм ЕП

Загальні положення. У другому механізмі використовується випадкове значення *S* [3]. Він заснований на механізмі PSS-R стандарту IEEE P1363a, а випадкова послідовність генерується згідно з методом, відомого як EMSR3 [8, 9]. Механізм ЕП 2 сумісний із механізмом IFSSR IEEE P1363a.

В механізмі особистий ключ ЕП, що використовується, має модуль довжиною *k* бітів. Його довжина визначає значення довжини ЕП *c* і довжину послідовності повідомлення *F*. Довжина випадкового значення (солі) $L_S > 0$.

Залежно від використання ідентифікатора геш-функції, додаткове поле T може приймати два варіанти. Варіант і довжина поля визначаються коефіцієнтом t.

У першому варіанті $t=1$, довжина 8 бітів або 2 напівбайти, ідентифікатор не використовується, і два напівбайти приймають значення, що дорівнюють ‘BC’:

У другому варіанті $t=2$, довжина 16 бітів або 4 напівбайти, перші два напівбайти можуть приймати значення від 0 до F, і вони визначають тип геш-функції, що використовуватиметься, а два інші напівбайти приймають значення, що дорівнюють ‘CC’.

Розмір *c* підпису для цієї схеми визначається як: $c = k - Lh - L_S - 8t - 2$.

Створення послідовності повідомлення. В механізмі 2 ЕП створення послідовності повідомлення здійснюється за 2 кроки – гешування та форматування.

Гешування повідомлення. В ході обчислення геш-значення h потрібно конвертувати довжину рядка $|M_1|$, у бітовий рядок *C* довжиною 64 біти. Далі необхідно згенерувати новий випадковий бітовий рядок *S* довжиною L_S бітів та обчислити геш-значення h довжиною *Lh* бітів, причому

$$h = H(C \| M_1 \| h(M_2) \| S) \text{ (див. табл. 1)}$$

Форматування повідомлення. Рядок повідомлення *F* має бути обчислений наступною або еквівалентною послідовністю кроків.

1. Нехай *P* є бітовим рядком, що складається з $(k + \delta - Lh - LS - |M_1| - 8t - 2)$ бітів зі значенням ‘0’, де $\delta = (1\text{-}k) \bmod 8$ (див. табл. 6).

2. Бітовий рядок *D* визначається як $D = P \| \text{‘1’} \| M_1 \| S$, де є один біт зі значенням ‘1’. Довжина *D* складає $(k + \delta - Lh - 8t - 1)$ бітів.

Якщо довжина геш-коду є цілим числом октетів, то бітовий рядок *D* також буде довжиною цілого числа октетів (див табл. 7).

3. Для створення бітового рядку *N* довжиною $(k + \delta - Lh - 8t - 1)$ бітів до геш-значення *h* застосовується функція генерації маски *g* (див. табл. 8).

Таблиця 5

Відновлення повідомлення

Біти	0	1	0/1	0	0	...	0	1	0/1	0/1	...	0/1	0/1	0/1	...	0/1	0/1	0/1	...	0/1
Блоки				Бітизаповнення					M_1				*H*				*T*			
Довжина блоку (бітів)	1	1		$k - L_h - \|M_1\| - 8t - 4$				1	$\|M_1\|$				L_h				$8t$			
Загальна довжина (бітів)	*k*																			

Таблиця 6

Форматування повідомлення

Біти	0	0	...	0
Довжина (бітів)	$k+\delta-L_h-L_S-\lvert M_1\rvert-8t-2$			

Таблиця 7

Форматування повідомлення

Біти	0	0	...	0	1	0/1	0/1	...	0/1	0/1	0/1	...	0/1
Блоки	P					M_1				S			
Довжина блоку (бітів)	$k+\delta-L_h-$ $-L_S-\lvert M_1\rvert-$ $-8t-2$				1	$\lvert M_1\rvert$				L_S			
Загальна довжина (бітів)	$k+\delta-L_h-8t-1$												

Таблиця 8

Форматування повідомлення

Біти	0/1	0/1	...	0/1
Довжина (бітів)	$k+\delta-L_h-8t-1$			

4. Довжина значення $D\oplus N$ складає ($k+\delta-$ $-Lh-8t-1$) бітів. Нехай D' є рядком з ($k-Lh-$ $-8t-1$) бітів, отриманих шляхом видалення δ найбільш лівих бітів $D\oplus N$ (див. табл. 9).

Таблиця 9

Форматування повідомлення

Біти	0/1	0/1	...	0/1	0/1	0/1	...	0/1
Блоки					D'			
Довжина блоку (бітів)	δ				$k-L_h-8t-1$			

Таблиця 10

Форматування повідомлення

Біти	0/1	0/1	...	0/1	0/1	0/1	...	0/1	0/1	0/1	...	0/1	1	1	0	0
Блоки	D'				H				T							
Довжина блоку (бітів)	$k-L_h-8t-1$				L_h				$8t$							
Загальна довжина (бітів)	$k-1$															

5. Нехай $F=D'\|H\|T$, де T є додатковим полем з $8t$ бітів, а F є рядком з (k-1).

Нехай також

$$D=P\|1\|M_1\|S$$

$$D'=(P\|1\|M_1\|S)\oplus g(H)$$

$$F=((P\|1\|M_1\|S)$$

$$\oplus g(H))\|h(C\|M_1\|h(M_2)\|S)\|T$$

Вказана послідовність подається на функцію створення підпису.

Відновлення повідомлення. Якщо значення найбільш правого октету послідовності відновлюваного повідомлення F^*, рядка з k-1 бітів, дорівнює‘BC’ у шістнадцятковій системі, то додаткове поле складається з цього одного октету; а якщо – ‘CC’, то додаткове складається з двох найбільш правих октетів F^*, де найбільш лівий октет є ідентифікатором геш-функції.

Під час перевірки потрібно використати потрібну функцію гешування та довжинук геш-значення Lh.

Підпис Σ відхилюється, якщо або додаткове поле або ідентифікатор геш-функції не можуть бути інтерпретовані, інакше, процес перевірки продовжується.

Далі частина відновлюваного повідомлення $M1$ відновлюється з послідовності відновленого повідомлення F^* і невідновлюваної частини M_2 таким чином:

1. Приєднуються δ бітів зі значенням ‘0’ до лівого кінця F^*.

2. Нехай D'^* є ($k+\delta-Lh-8t-1$) найбільш лівими бітами отриманого рядка, а h^* – наступними Lh бітів (див. табл. 11).

Таблиця 11

Відновлення повідомлення

Біти	0	0	...	0	0/1	0/1	...	0/1	0/1	0/1	...	0/1
Блоки	D'*								H*			
Довжина блоку (бітів)	δ				$k-L_h-8t-1$				L_h			
Загальна довжина (бітів)	$k-1+\delta$											

3. Для створення бітового рядка N^* довжиною ($k+\delta-Lh-8t-1$) бітів до геш-значення h^* застосується функція генерації маски g. $N^*=g(H^*)$ (див. табл. 12).

Таблиця 12

Відновлення повідомлення

Біти	0/1	0/1	...	0/1
Довжина (бітів)	$k+\delta-L_h-8t-1$			

4. Нехай $D^*=D'^*\oplus N^*$.

5. Встановлюється значення ‘0’ на δ найбільш лівих бітах D^*.

6. Починаючи з лівого кінця D^* знаходять перший біт зі значенням ‘1’. Цей біт і всі нулі зліва від нього видаляються. Н нехай S^* є L_S найбільш правими бітами D^*, і $M_1{}^*$ є бітами D^*, що залишились. Якщо нема першого біта зі значенням ‘1’, повертається значення «помилка» і процес зупиняється (див. табл. 13).

Таблиця 13

Відновлення повідомлення

Біти	0	0	...	0	1	0/1	0/1	...	0/1	0/1	0/1	...	0/1
Блоки	P					M_1				S			
Довжина блоку (бітів)	$k+\delta-L_h-$ $-L_S-\lvert M_1\rvert-$ $-8t-2$				1	$\lvert M_1\rvert$				L_S			
Загальна довжина (бітів)	$k+\delta-L_h-8t-1$												

7. Бітовий рядок $M_1{}^*$ конвертується у бітовий рядок C довжиною 64 біти .

8. Якщо $H^*=h(C\|M_1{}^*\|h(M2^*)\|S^*)$, то на вихід подається частина відновленого повідомлення $M_1{}^*$, *інакше «помилка»*.

2.3. Третій механізм ЕП

Третій механізм ЕП ідентичний другому, за виключенням того, що S є фіксованим значенням, але довжина $L_S \geq 0$ (на відміну від обмеження $L_S > 0$, яке застосовується вище у другому механізмі). Отже ця схема є детерміністською.

Фіксоване значення солі S може бути обрана підписувачем. По-іншому, вона може бути зазначеною як частина параметрів домену.

Вважається, що третій механізм ЕП 3 переважає перший, хоча вони і подібні, але відносно механізму 3 є математичні докази безпеки.

3. УЗАГАЛЬНЕННЯ МЕХАНІЗМІВ ЕП НА ОСНОВІ ВІДКРИТИХ КЛЮЧІВ ТА ФУНКЦІЇ ГЕНЕРАЦІЇ МАСКИ

Узагальнені дані відносно механізмів ЕП, що викладені в стандарті ДСТУ 9796-2:2014, подано у табл. 14 [3].

Також у подальшому використовуються такі позначення: f – ціле число, яке має F як бінарне подання; f^* – ціле число, що обчислене під час розкриття підпису; J – ціле число, що обчислене під час вироблення підпису; J^* – ціле число, що обчислене під час розкриття підпису; n – коефі-

Таблиця 14

Узагальнення механізмів ЕП

Таємні параметри безпеки того, хто підписує	p, q, s
Розмір підпису	$c \geq 7$
Довжина відновлюваного повідомлення	$c^* \leq c$, $c^* \leq \lvert M \rvert\, i$ $c^* \equiv \lvert M \rvert \pmod 8$; *Для схеми підпису* 1*:* $c^* = min\{c - \Delta\, i\, \lvert M \rvert\}$, де $\Delta = (c - \lvert M \rvert) \bmod 8$.
Вимоги до модуля n	$n = p^*q$ Кожен підписувач повинен особисто та випадково вибрати два простих числа p та q, що відрізняються, згідно з такими умовами: – якщо v непарне, тоді $p-1$ та $q-1$ мають бути взаємно простими відносно показника перевірки v; – якщо v парне, тоді $(p-1)/2$ та $(q-1)/2$ мають бути взаємно простими до v. Більш того, p та q не мають співпадати один з одним за модулем 8
Вимоги до довжини модуля k	$2^{k-1} < n \leq 2^k - 1$
Вимоги до показника підпису s	** якщо v непарне* $sv - 1 \equiv lcm(p-1, q-1)$; ** якщо v парне* $sv - 1 \equiv lcm(p-1, q-1)/2$
Створення підпису	<u>функцією створення підпису</u>: *якщо v є непарним, тоді $J=f$, *якщо v парне та $(f \mid n) = +1$, тоді $J=f$, та *якщо v парне та $(f \mid n) = -1$, тоді $J=f/2$; $\Sigma = k$-1 біт, що відповідає $\Sigma \equiv \min\{J^s \bmod n,\ n - (J^s \bmod n)\}$.
	<u>альтернативною функцією створення підпису</u> (v – непарне): $J = f$; $\Sigma = k$ біт,що відповідає $\Sigma \equiv J^s \bmod n$
Розкриття підпису	<u>функцією розкриття підпису</u>: $J^* = \Sigma^{\,v} \bmod n$. Якщо$v$ непарне та – якщо $J^* mod\ 16 = 12$, тоді $f^* = J^*$, – якщо $J^* mod\ 16 = n - 12\ mod\ 16$, тоді $f^* = n - J^*$. Якщоv парне та – якщо $J^* mod\ 8 = 1$, тоді $f^* = n - J^*$, – якщо $J^* mod\ 8 = 4$, тоді $f^* = J^*$, – якщо $J^* mod\ 8 = 6$, тоді $f^* = 2J^*$, – якщо $J^* mod\ 8 = 7$, тоді $f^* = 2(n - J^*)$ *Відхилено підпис тоді, коли:* $f^* mod\ 16 \neq 12$ та коли не виконується умова $f^* \leq 2^{k-1} - 1$.
	<u>альтернативною функцією розкриття підпису</u>: $J^* = \Sigma^{\,v} mod\ n$ $f^* = J^*$ *Відхилено підпис тоді, коли:* $f^* mod\ 16 \neq 12$ та коли не виконується умова $f^* \leq 2^{k-1} - 1$.
Складання підпису	$M^* = M_1^* \| M_2^*$

цієнт (частина особистого ключа підпису та відкритого ключа перевірки); p, q – прості множники модуля (коефіцієнта); s – показник підпису; v – показник перевірки; lcm (a, b) – найменше спільне кратне чисел a та b; min $\{a,b\}$ – найменше з двох значень a та b; $(a|\ n)$ – символ Якобі a відносно n.

Генерація маски (бітовий рядок N) має здійснюватися наступною або еквівалентною послідовністю кроків.

1. Якщо значення L_Z перевищує обмеження довжини (2^{64}–33 для виділених функцій гешування 1 і 3 зі стандарту ISO/IEC 10118-3), або якщо $L_N > L_h \times 2^{32}$, то «помилка» і зупинитися.

2. Нехай N є пустим рядком.

3. Нехай i=0.

3.1 Перетворити i у бітовий рядок C довжиною 32 біти з використанням [3].

3.2 Нехай $N := N \| h(Z \| C)$.

3.3 Нехай $i := i+1$.

3.4 Якщо $i < \lceil L_N / L_h \rceil$, повернутися до кроку 3.1.

4. Інакше подати на вихід L_N крайніх лівих бітів N.

4. АНАЛІЗ СТІЙКОСТІ ЕП З ВІДНОВЛЕННЯМ ПОВІДОМЛЕННЯ

Розглянемо в історичному плані питання стану та удосконалення ЕП за критерієм криптографічної стійкості. Як основні під час дослідженнь криптографічної стійкості виберемо атаки типу екзистенційна підробка, селективна підробка та повне розкриття. Також вважатимемо, що геш-функція, яка застосовується, є колізійно стійкою.

4.1. Атака « екзистенційна підробка»

Загрозливість атаки «екзистенційна погроза» полягає в створенні криптоаналітиком для будь-якого, можливо безглуздого повідомлення m', що відрізняється від перехопленого, відповідного правильного ЕП. Це можливо, якщо геш-функція, що використовується, має ряд слабкостей, тому геш функція має бути колізійно стійкою [3, 7-9, 13].

μ(x, 0) = 6a05b0daa253785bc8ab962c6047d6171eab4429160aa5defc7ff64bcc0a1de4
6b48a5c79a4e3394a66da695bb972de0be0bd94e40e8bad820b2b942aa8e71db
bca41cb3c2e7f2a68c88f4aa47a70a491199ef13b6a75b0650b22b4b2547f72e
f607c019702fdaa4f406a00cee511fcc5dd67dc2cd52c519976bb74971974b40
cf0e5459b5d18b7e15f338db671eb02f5ea32db8bc4a0bcf5cc896894fe1e738
6395f2f6f5c76ff7b6056ab5a5dfa84c95a117316615586bffb701d8716fd0bd
2712ada91a94d2d516246b3910d717cc9de96c48d2bc825994af77d0c283b93f
67f3c8256aeb45bab7037c000000000000000000000000000000000000000000000000bc16

Таким чином, цілі числа $t_i = bN - a\mu(x, h_i)$, більші за 170 біт.

*Добуток перших простих чисел. Було обґрунтовано о*птимальний вибір для 1 170 біт – це 2^{21}. При цьому, оскільки комп'ютери Amazon обмежені за пам'яттю (менше 1 Гб оперативної пам'яті на ядро), то було було використане $l = 220$. Вибір цього значення мав також перевагу при завершальному кроці – розв'язанні системи рівнянь, який виконувався одним офлайн – ПК. Добуток простих чисел визначається за лічені секунди за допомогою бібліотеки MPIR.

Гешування. На цьому кроці потрібно визначити геш-значення повідомлень для m_i варіантів, оскільки частина атаки при знаходженні гладкості вимагає t_i варіантів (було обрано 2^{19} цілих чисел). Повідомлення були 2048-бітні. Вони гешувалися за допомогою SHA-1 OpenSSL.

Пошук гладких та напівгладких чисел. Пошук здійснювався за допомогою алгоритму Бернштейну.

Розклад та пошук колізій. Вихідними даними для цього кроку був набір текстових файлів, що містять гладкі та напівгладкі t_i з відповідними номерами повідомлень. Була побудована матриця 1048576 × 1050667, яка мала 14,215,602 ненульових.

Основні ненульові елементи кінцевої матриці над GF(2) були знайдені за допомогою блокового алгоритму Cop-persmith's block Wiedemann algorithm з параметрами $m = n = 4$ та $k = 2$). На це було необхідно 16 годин на одному ПК (2.7 GHz), причому перша частина обчислень – з використання двох ядер, а заключна – чотирьох ядер. Кількість підписів, необхідних для отримання 124 відповідних підробок змінюється в інтервалі від 432,903 до 435,859.

Аналіз складності підробки. В табл. 15 наведено результати оцінки складності здійснення атаки типу екзистенцій на підробку [4, 7–10].

Таблиця 15

Алгоритм Бернштейна + велике просте число (ПК 2,4 ГГц)

$a = \log_2 t_i$	$\log_2 l$	Оцінка часу	$\log_2 \tau$	Коштовність Amazon *ec*2
64	11	15 секунд	20	незначна
128	19	4 дні	33	10
160	21	6 місяців	38	470
170	22	1.8 років	40	1,620
176	23	3.8 років	41	3,300
204	25	95 років	45	84,000
232	27	19 століть	49	1,700,000
256	30	320 століть	52	20,000,000

Аналіз даних табл. 15 дозволяє зробити висновок, що атака можлива до $\simeq$ 200 біт, але стає неможливою для більших значень a.

Ця атака показує недолік практичних уразливостей у чинному на сьогодні стандарті ISO / IEC 9796-2.

Рекомендується одноранговий режим перетворення і заміна його функцією перетворення PSS.

4.2. Атака «селективна підробка»

Атака «селективна підробка» зводиться до створення для заздалегідь обраного повідомлення m правильного ЕП, для якого результати перевірки на цілісність і справжність будуть позитивними.

Використаємо згідно з табл. 16 альтернативні функції.

Таблиця 16

«Селективна підробка»

$J = J^*$	
Підпис	Перевірка
$\Sigma = J^S \bmod n \Rightarrow J \equiv \Sigma^{\frac{1}{S}} \bmod n$	$J^* \equiv \Sigma^{\nu} \bmod n$
$\Sigma^{\frac{1}{S}} \bmod n = \Sigma^{\nu} \bmod n$	

В ході використання альтернативних функцій, загроза атаки «селективна підробка» можлива тоді, коли показник перевірки буде протилежно пропорційним показнику підпису $\frac{1}{s} = v$ та коли виконуватиметься умова $J^s < n$. Оскільки $n = p*q$, то для захисту даних схем від «селективної підробки» прості множники p та q повинні бути досить великими.

Оскільки при створенні підпису підпис Σ приймає мінімальне значення із

$$J^s \bmod n \text{ та } n - (J^s \bmod n),$$

тобто, $\Sigma \equiv \min\{J^s \bmod n,\ n - (J^s \bmod n)\}$, а при мінімальному $J^s \bmod n$ результат досліду буде такий же, як і під час використання альтернативних функцій (що описаний вище), тоді потрібно дослідити лише за умови мінімального $n - (J^s \bmod n)$ (див. табл. 17)

Таблиця 17

«Селективна підробка»

$J = J^*$	
Підпис	Перевірка
$\Sigma \equiv \min\{J^S \bmod n, n - J^S \bmod n\} \sim$ $\sim \Sigma \equiv n - J^S \bmod n$ $\Rightarrow J \equiv (n - \Sigma)^{\frac{1}{S}} \bmod n$	$J^* \equiv \sum^{\nu} \bmod n$
$(n - \Sigma)^{\frac{1}{S}} \bmod n = \sum^{\nu} \bmod n$	

Оскільки розв'язок останнього рівняння потребує дуже великих ресурсів, то при даному значенні $n - (J^s \bmod n)$, механізм ЕП можна вважати захищеними, до того ж, для більшого захисту нанізмів потрібно дотримуватися умова $n \gg \Sigma$.

Результати атаки «селективна підробка». В ході використання не альтернативних функцій та приймаючи $n - (J^s \bmod n)$ як мінімальне значення, захищеність від атаки «селективна підробка» буде вищою. Використовуючи альтернативні функції та приймаючи $(J^s \bmod n)$ як мінімальне значення, загроза атаки «селективна підробка» можлива лише, коли показник перевірки буде протилежно пропорційним показнику підпису $\frac{1}{s} = v$ та коли виконуватиметься умова $J^s < n$.

Усе це свідчить про те, що для кращого захисту механізмів стандарту прості множники p та q мають бути досить великими.

Атака «екзистенційна підробка»

Погроза цієї атаки полягає в створенні криптоаналітиком для будь-якого, можливо безглуздого повідомлення m', що відрізняється від перехопленого, реального ЕЦП. Це можливо, якщо використовувана геш-функція має ряд слабкостей, тому складність створення колізій має носити експоненційний характер.

5. АНАЛІЗ СКЛАДНІСТЬ МЕХАНІЗМІВ

В табл. 18 наведені результати оцінки складності вироблення та перевірки ЕП для механізмів стандарту.

Таблиця 18

Складність механізмів ЕП ДСТУ ISO/IEC9796-2

	Операція	Схема підпису 1	Схеми підпису 2, 3
Підпис повідомлення	$h()$	1	2
	$a + b \bmod n$	1	1
	$a \oplus b$	1	1
	$g()$		1
	$k^a \bmod n$	1	
	$(a\|b)$	1	
Перевірка підпису	$h()$	1	2
	$a + b \bmod n$		1
	$a \oplus b$	1	1
	$g()$		1
	$k^a \bmod n$	1	

ВИСНОВКИ

1. Сьогодні досягнуто реальні показники збільшення потужності криптоаналітичних систем. Але реальні загрози існуючим криптографічним системам можуть принести квантові комп'ютери. Тому однією із найбільших проблем у криптографії є захист відповідних криптосистем та стандарті, що на основі них розробляються, від існуючих та потенційних атак.

2. Сьогодні знайшли застосування 2 основних типи ЕП – з додатком та з відновленням повідомлення.

3. В ЕП з відновленням повідомлення частина або повне повідомлення можуть бути відновленими з ЕП. Основним його призначенням є підпис коротких повідомлень.

4. Особливістю механізму ЕП з відновленням повідомлення є те, що в ньому висувають правила використання функції формування доповнення. Для повної перевірки перевірник ЕП повинен мати повну та неушкоджену збитковість повідомлення. Тому саме такий механізм ЕП з відновленням повідомлення доцільно ви-

користовувати в інформаційних системах і протоколах з чітко визначеними повідомленнями. Вказане є принциповою особливістю з точки зору застосування ДСТУ 9796 – 2:-2014.

5. Схема ЕП з відновленням повідомлення дає перевагу в ході застосування з повідомленнями невеликого розміру. ЕП з таким механізмом може ефективно використовуватися в інфраструктурах з відкритими ключами, у протоколах з малим розміром повідомлення, наприклад, електронних магазинах, тобто для захисту товарів і послуг тощо.

6. Підпис з відновленням повідомлення, порівняно з підписом з доповненням, надає додаткову послугу безпеки – конфіденційність. Також для невеликих обсягів повідомлення можливо зробити таємною всю інформацію, що передається, у самому підписі.

7. Крім того, механізми ЕП 1, 2 та 3 досліджені щодо стійкості відносно атак «повне розкриття», «екзистенційна підробка» та «селективна підробка».

В табл. 19 наведено узагальнені результати оцінки стійкості криптографічних перетворень ДСТУ 9796 – 2:-2014 відносно основних атак.

Таблиця 19

Захищеність ДСТУ 9796 – 2:-2014 від основних атак

Атаки / Алгор.	Атака «екзистенційна підробка»	Атака «селективна підробка»	Атака «повне розкриття»
Схема підпису 1	Можливі вразливості при використанні «альтернативних» функцій	Захищені	Субекспоненційна складність
Схема підпису 2			
Схема підпису 3			

Література

[1] *Горбенко Ю.І.* Методи побудування та аналізу, стандартизація та застосування криптографічних систем. Монографія. Харків: Форт, 2015. – 959 с.

[2] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Монографія. Видання 2-ге. Харків: Форт, 2012. – 878 с.

[3] ISO/IEC 9796-2:2010 Information technology – Security techniques – Digital signature schemes giving message recovery – Part 2: Integer factorization based mechanisms.

[4] *Jean-Srebastien Coron, David Naccache, Mehdi Tibouchi, and Ralf-Philipp Weinmann.* Practical Cryptanalysis of ISO/IEC 9796-2 and EMV Signatures – 2009.

[5] *Tom Berson.* Skype Cryptosystem Overview – Anagram Laboratories, December 2005.

[6] ICAO: Машино зчитувальні проїзні документи. Частина 1. Машинозчитувальні паспорта. Том 2.Специфікація на електронні паспорта із засобами біометричної автентифікації. Видання 6. 2006 р.

[7] *J.-S. Coron*, 'On the exact security of full domain hashing'. In: M. Bellare (editor), Advances in Cryptology – Crypto 2000, Lecture Notes in Computer Science 1880 (2000), Springer-Verlag. – P. 229–235.

[8] *J.-S. Coron, D. Naccache, and J.P. Stern,* 'On the security of RSA padding'. In: M.J. Wiener (editor), Advances in Cryptology – Crypto '99, Lecture Notes in Computer Science 1666 (1999), Springer-Verlag. – P. 1–18.

[9] IEEE Std 1363-2000, Standard specifications for public key cryptography

[10] IEEE Std 1363a-2004, Standard specifications for public key cryptography – Amendment 1: Additional techniques.

[11] ISO/IEC 9797-2:2002, Information technology – Security techniques – Message Authentication Codes (MACs) – Part 2: Mechanisms using a dedicated hash-function.

[12] *J. Jonsson,* 'Security proofs for the RSA-PSS signature scheme and its variants'. Proceedings of the 2nd NESSIE Workshop, Royal Holloway, University of London, September 2001. Full version available in IACR cryptology archive 2001/053.

[13] *B. Kaliski,* 'On hash function firewalls in signature schemes'. In: B. Preneel (editor), Cryptographers' Track RSA Conference 2002, Lecture Notes in Computer Science 2271 (2002), Springer-Verlag. – P. 1–16.

Надійшла до редколегії 27.11.2015

Ковальова Наталія Вікторівна, студентка кафедри БІКС ХНУ ім. В.Н. Каразіна. Наукові інтереси: дослідження міжнародних стандартів, аналіз стійкості цифрових підписів.

Горбенко Юрій Іванович, фото та відомості про автора див. на с. 290.

Бакликов Олександр Олександрович, інженер систем захисту інформації ПАТ «Інститут інформаційних технологій». Наукові інтереси: розробка та застосування систем криптографічного захисту інформації.

УДК 681.3.06

Анализ ЦП с восстановлением сообщения на основе стандарта ДСТУ ISO/IEC 9796-2 / Н.В. Ковалёва, Ю.И. Горбенко, А.А. Баклыков // Прикладная радиоэлектроника: науч.-техн. журнал. – 2015. – Том 14. – № 4. – С. 306–314.

Рассмотрены суть, требования, модели цифровых подписей с восстановлением сообщения на основе стандарта ДСТУ ISO/IEC 9796-2. «Информационные технологии. Методы защиты. Цифровые подписи, обеспечивающие обновление сообщения. Часть 2. Основные методы факторизации целых чисел» и анализ их стойкости к атакам.

Ключевые слова: цифровая подпись с восстановлением сообщения, атаки, анализ.

Табл.: 19. Библиогр.: 13 назв.

UDC 681.3.06

Analysis of digital signatures giving message recovery based on DSTU ISO/IEC 9796-2 / N.V. Kovaleva, Yu.I. Gorbenko, O.O. Buklykov // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 306–314.

Consideration of the nature, requirements, models of digital signatures with message recovery based on ISO / IEC 9796-2 is provided. «Information Technologies. Security techniques. Digital signature schemes giving message recovery. Part 2: Integer factorizatiuon based mechanisms» and the analysis of their resistance to attacks.

Keywords: digital signature giving message recovery, attacks, analysis.

Tab. 19. Ref.: 13 items.

МЕТОДЫ И СРЕДСТВА СИММЕТРИЧНЫХ КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

UDC 681.3.06

IMPROVEMENT OF THE METHOD FOR OPTIMAL S-BOXES GENERATION

M.Yu. RODINKO, R.V. OLIYNYKOV, T.O. HRINENKO

The known method of high nonlinear S-boxes generation based on the gradient descent requires a consecutive application of several criteria for each formed substitution. This paper presents improvement of the considered method by the appropriate selection of the criteria application order which decreases the required computation power for S-box generation. The proposed modification allows generation of a byte substitution with nonlinearity 104, algebraic immunity 3 and 8-uniformity within approximately 30 minutes of a single PC running time.

Keywords: S-box, nonlinearity, algebraic immunity, vectorial Boolean function.

INTRODUCTION

Block ciphers are among the most spread cryptographic primitives. Such algorithms are used to provide data confidentiality and integrity, as well as they are used as a core element of other cryptographic transformations like pseudorandom sequences generators, hash functions, etc. [1, 2].

Each block cipher contains a nonlinear function in a quotient ring for providing nonlinear dependence between plaintext, key and ciphertext [3]. Often such a function is implemented using a substitution table (S-box).

S-box properties have serious impact to the cipher strength (and its margin) against various methods of cryptanalysis [4, 5]. Appropriate selection of S-boxes allows reducing the number of rounds of iterative symmetric transformation (increasing performance) keeping its cryptographic strength.

S-boxes are called optimal if they are satisfied a set of essential criteria reaching extreme values for differential, linear and algebraic characteristics [6].

Most known methods of S-box generation are insufficiently effective for obtaining substitutions with optimal cryptographic characteristics on a single PC.

This paper presents improvement of the known high nonlinearity S-boxes generation method [7] allowing several times decrease of required computation power.

1. THE S-BOXES SELECTION CRITERIA

Basic S-boxes selection criteria can be divided into two groups. The first one includes the criteria taking into account the transformation strength against cryptanalysis methods. Currently as the main characteristics are considered the following: differential [8], linear [9] and algebraic [10].

The second group includes criteria based on the S-box Boolean functions cryptographic properties evaluation [11]. These include nonlinearity, the autocorrelation maximum, distribution criterion and others. However, as shown in [6], many of this group criteria are unessential or redundant for applications in block transformations. Thus, the following criteria considered to be essential [6].

1. The maximum value of difference distribution table (DDT) is defined as

$$\delta = \max_{\alpha \in F_2^n, \alpha \neq 0, \beta \in F_2^n} \#\{x \mid S(x) \oplus S(x \oplus \alpha) = \beta\}.$$

This value influences the cipher strength against differential cryptanalysis, which is one of the most universal and effective attacks on block ciphers.

Equivalent to a maximum value of differential table is the notion of δ-uniformity [12].

Definition 1. *Let G_1 and G_2 be finite Abelian groups. A mapping $F: G_1 \to G_2$ is called differentially δ-uniform if for all $\alpha \in G_1$, $\alpha \neq 0$ and $\beta \in G_2$*

$$|\{z \in G_1 | F(z + \alpha) - F(z) = \beta\}| \leq \delta.$$

According to this definition, the optimal characteristics of resistance of transformation F against differential attacks are associated with low values of δ-uniformity. Obviously that the requirement of low values of δ-uniformity is equivalent to the requirement of low values of the maximum value of non-trivial difference transformation. Therefore to achieve a high strength of cryptographic transformation it is necessary to obtain low values of δ-uniformity.

2. The maximum absolute value of linear approximation table (LAT) is defined as

$$\lambda = \max_{\alpha,\beta \neq 0} \left| LAT(\alpha,\beta) - 2^{n-1} \right|,$$

where

$$LAT(\alpha,\beta) = \#\left\{ x \,\middle|\, x \in Z_2^n, \bigoplus_{s=0}^{N} (x[s] \cdot \alpha[s]) = \bigoplus_{t=0}^{N} (S(x)[t] \cdot \beta[t]) \right\},$$

and $\mu[s]$ – bit s of value μ.

The property influences the cipher strength against linear cryptanalysis. In [13] it was shown that the complete set of linear characteristics, called linear hull, should be taken into consideration for precise evaluation of the cipher strength against linear attacks.

A large part of the known methods of evaluating of block cipher strength to differential and linear crypta-

nalysis based on the differential and linear properties of S-boxes used in their construction. In [14] it was shown that the SPN structure with maximal diffusion layer provides a provable security against differential (linear) cryptanalysis: the probability of each differential (linear hull) is bounded by p^n (q^n), where p (q) is a maximal non-trivial differential (linear) probability of n active S-boxes.

3. The minimum degree of S-box Boolean function (BF). Each S-box can be represented as a set of Boolean functions. Let $S = (f_0, f_1, \ldots, f_{m-1})$ – substitution $n \times m$, where f_i – Boolean function from n variables. The minimum degree of S-box [15] is defined as

$$deg(S) = \min_{0<j<2^m} \left(deg\left(g_j\right) \right),$$

where g_j – a set of all linear combinations of f_i; $deg\left(g_j\right)$ – the maximum degree of Boolean function represented in ANF.

4. Algebraic immunity characterizes the cipher strength against algebraic attack, i.e., a minimum degree of an overdefined system of equations which can be used to describe the S-box. At such description of the S-box a lower terms degree can be received than when presenting in the form of a set of Boolean functions.

In general form for S-box $n \times m$ the required number of equations of system of degree d is [6]

$$r = N_c - Rank(A),$$

where

$$N_c = \sum_{i=0}^{d} C_{n+m}^i,$$

where *Rank(A)* – rank of the binary matrix A containing all possible multiplications of input and output bits of S-box.

Dimensionality of such matrix is

$$|A| = \left(2^n\right) \times N_c.$$

5. The absence of fixed points. According to this criterion substitution S must not have such transitions that $S(x) = x$. In most ciphers criterion is used for protection against statistical attacks.

The nonlinearity of S-box is also assumed to be a one of the main criterion. In terms of Boolean functions [15], the nonlinearity of substitution S is

$$NL(S) = \min_{0<j<2^n} \left(NL\left(g_j\right) \right),$$

where $NL\left(g_j\right)$ – minimal Hamming distance between the function g_j and all affine functions over the field $GF(2^n)$.

However, the value of nonlinearity is uniquely determined from the maximum of linear approximation table [15] and for substitution S of degree 2^n is

$$NL(S) = 2^{n-1} - \frac{1}{2} \max_{\alpha,\beta \in GF\left(2^n\right)} \left| LAT(\alpha,\beta) \right|.$$

2. OPTIMAL S-BOXES GENERATION

A large part of all existing methods of S-boxes generation can be divided into two types: algebraic [16, 17] or random. The latter are simpler for implementation, but with computational power for practical implementation limited to the single PC it is possible to obtain byte substitutions with nonlinearity up to 98.

Table 1 shows the properties research results of randomly generated substitutions of degree $n = 2^8$. The sample in the given experiment obtained for 10 million substitutions. During the experiment no substitution with nonlinearity 100 has been found. Herewith, all generated S-boxes satisfied the criterion of algebraic immunity. In [6] it was obtained random substitutions with nonlinearity 100, but on the cluster of 4096 computers.

Contrary to random generation methods, the algebraic ones suggest the S-boxes on the basis of balanced Boolean functions with nonlinearity 112.

Table 1

Cryptographic properties of randomly generated S-boxes

Criterion	Value	% of S-boxes satisfying criterion
The maximum of DDT	8	0,004
The maximum of LAT (nonlinearity)	32 (96)	11
	30 (98)	0,15
	28 (100)	0
The minimum degree of BF	7	30
Algebraic immunity	3	100

Among the algebraic methods of S-boxes generation it is widely used the power operations in the finite field [16]. Such substitutions were considered the most optimal for a long time, and Rijndael/AES [18] also uses this type of S-box. However, these byte substitutions have the unwanted property: the value of their algebraic immunity is only two that creates the potential cipher vulnerability to algebraic attacks.

The considered method of high nonlinear S-boxes generation providing both algebraic immunity and strength to the differential and linear cryptanalysis has the following steps [7].

1. Pseudorandom substitution generation:

1.1. generation permutation S based on vectorial Boolean function that implements power transformation in the finite field;

1.2. random swap of N value pairs of permutation S and forming permutation S'.

2. Compliance test of generated permutation S' to the S-box criteria set.

The given algorithm combines the advantages of algebraic and random methods of S-boxes generation and allows obtaining the substitution with an algebraic immunity 3 and nonlinearity up to 104. The problem of the existence of permutations with a higher nonlinearity while maintaining high values of algebraic immunity remains open.

Another method that allows obtaining the S-boxes with nonlinearity 104 has been proposed in [19]. The method combines the special genetic algorithm with total tree searching. However, the author does not give any information about the values of other indicators of obtained substitutions.

We note that the block symmetric cipher Kalyna [20] and the hash function Kupyna [21] presented in the corresponding new Ukrainian standards use S-boxes with the best currently known cryptographic characteristics (given in the Table 2).

Thus, the modified method of gradient descent is currently assumed to be the most effective method of the optimal S-boxes generation. However, the method can be further optimized in terms of performance for using on a single PC.

Table 2

Cryptographic characteristics of substitutions from the Kalyna and Kupyna

Characteristic	Value
The maximum of DDT	8
The maximum of LAT	24
The minimum degree of BF	7
Nonlinearity	104
Algebraic immunity	3
The absence of fixed points	Yes

3. OPTIMIZATION OF S-BOXES GENERATION METHOD

As input parameters the method of generating S-boxes accepts the following [7]:

– vectorial Boolean function $F(x)$ (with nonlinearity 112 and the maximum of difference distribution table equal to 4);

– the number of random pairs of values N to be swapped.

As a vectorial Boolean function is proposed to use $F(x) = x^d$. To obtain possible values of degree d it is used the following formula [16]:

$$d = (2^n - 1) - 2^i,\ i = 0,...,7.$$

Table 3 shows the vectorial Boolean functions permitted for use in the *S*-boxes generation algorithm with $n = 2^8$.

Table 3

List of the vectorial Boolean functions permitted for use

i	$F(x)$
0	x^{127}
1	x^{191}
2	x^{223}
3	x^{239}
4	x^{247}
5	x^{251}
6	x^{253}
7	x^{254}

A value of $N = 22$, at which all necessary properties of the S-box are reached, has been obtained in [6].

The following PEA-equivalent transformation is applied to the final substitution not only for removing fixed points, but also for destruction cyclic structure:

$$F(x) = M_1 \cdot G(M_2 \cdot x \oplus V_2) \oplus V_1.$$

The main part of computational resources is spent on the second stage of the search S-boxes – checking substitutions for compliance to the selection criteria set. Optimization of this stage significantly decreases the *S*-boxes generation time.

Selection criteria of substitutions are partially interdependent. Changing the order of applying the criteria can substantially reduce the search time of S-box. Let's consider the principle of finding the most optimal order of applying the criteria.

Let there be given k selection criteria substitutions $\xi_0, ..., \xi_{k-1}$. Then the number of possible combinations of k criteria specifying the order of their use is $k!$.

Let F_σ, where $\sigma \in [0; k!)$, is a combination of the criteria of the following form:

$$F_\sigma = \xi_{\theta_{k-1}(\sigma)} \circ \xi_{\theta_{k-2}(\sigma)} \circ \ldots \circ \xi_{\theta_i(\sigma)} \circ \ldots \circ \xi_{\theta_0(\sigma)},$$

where $\theta_i(\sigma) \in [0; k)$ – function that sets criterion for the i-th position in combination F_σ.

Let $T(F_\sigma)$ be a function returning time of checking a single substitution using a criteria sequence F_σ. Then the problem of minimizing time of checking substitution for compliance of k criteria is to find t_{min}:

$$t_{\min} = \min_{0 \le \sigma < k!} \left(T(F_\sigma) \right).$$

The combination of the criteria F_σ corresponding to the value t_{min} is the most optimal.

Now define an analytic expression for finding the values of the function $T(F_\sigma)$. It is introduced the following factors influencing time of substitution check: p_i – probability that the substitution satisfies the i-th criterion; v_i – time of checking of one substitution to compliance to the i-th criterion.

Here the index i denotes the ordinal number of criterion in the particular combination F_σ. Application of criteria is performed from right to left.

The values of the factors are found experimentally because there no analytical methods for their obtaining at the moment.

Using these factors, the following expression for $T(F_\sigma)$ was got:

$$T(F_\sigma) = \sum_{i=0}^{k-1} (\varphi_i \cdot v_i),$$

where $\varphi_0 = 1$, $\varphi_i = \varphi_{i-1} \cdot p_{i-1}$, $i = 1 \ldots k-1$.

Minimizing the function $T(F_\sigma)$ allows to get the optimal criteria sequence application for the S-boxes generation.

4. PRACTICAL RESULTS

4.1. Comparison of theoretical and empirical results

The proposed optimization was used for byte S-boxes generation with application of the following four criteria ($k = 4$):

– the maximum of DDT $a = 8$;

– the maximum of LAT $b = 26$;

– the minimum degree of BF $c = 7$;

– algebraic immunity $d = 3$.

Substitutions generation is performed on the basis of a vectorial Boolean function $F(x) = x^{254}$.

Table 5 shows the experimentally obtained values of the factors p and v.

According to the formula the values of function $T(F_\sigma)$ for combinations of criteria $F_\sigma \in [0; 24)$ were calculated. The calculated values are shown in Table 6.

Table 5

The values of factors for four criteria

Criterion	Factor p	Factor v, sec
a	0.66	0.0003
b	0.1	0.0017
c	0.3	0.0018
d	0.6	0.0067

According to the Table 6 $t_{\min} = 0.0016735$ value of the function is obtained for the combination of criteria $d \circ c \circ b \circ a$.

Values given in the Table 6 is time taken to check a single substitution. Experiments have shown that for one *S*-box generation satisfying four criteria 102 substitutions must be checked on the average. Thus, the time of *S*-box generation can be calculated as $T_{theor}(F_\sigma) = T(F_\sigma) \cdot 102$.

Experimental values of *S*-box generation time for all combinations of criteria were obtained. Fig. 1 shows graphs of functions $T_{theor}(F_\sigma)$ (continuous curve) and $T_{exp}(F_\sigma)$ (dotted curve).

4.2. High nonlinear S-boxes generation

The values of the function were calculated and it was chosen the best order of the criteria application for optimal S-boxes generation for the following set:

- the maximum of DDT $a = 8$;
- the maximum of LAT $b = 24$ (compared with the previous case the nonlinearity increased to 104);
- the minimum degree of BF $c = 7$;
- algebraic immunity $d = 3$;
- the absence of short cycles (up to 3).

Experiments have shown that the probability that the substitution has nonlinearity 104 is equal to 0,0000007.

The absence of short cycles is reached by applying PEA-equivalence to the given *S*-box, so it is not necessary to include this criterion to the list of criteria when the minimum of time is been calculating.

The minimum value $t_{\min} = 0{,}001422$ when combinations of criteria $d \circ c \circ b \circ a$ and $c \circ d \circ b \circ a$.

To generate an optimal *S*-box it is needed to check 1,100,000 substitutions on average. Thus, the average generation time of the one optimal *S*-box equals to $t_{\min} \cdot 1{,}100{,}000 = 0.001422 \cdot 1{,}100{,}000 = 1564.2$ sec. ≈ 26 minutes. The experimental results of optimal substitution generation time confirms analytically obtained value.

CONCLUSIONS

The paper presents the optimization of the known *S*-box generation method with high nonlinearity, based on the time minimization of *S*-box checking for compliance with the set of criteria. The presented approach allows the order determination of the selection criteria application in which the checking time of *S*-box will be minimal.

Table 6

The calculated values of function $T(F_\sigma)$

A number of criteria combination	The order of application of the criteria	The value of function $T(F_\sigma)$	A number of criteria combination	The order of application of the criteria	The value of function $T(F_\sigma)$
0	$a \circ b \circ c \circ d$	0.0080914	12	$c \circ a \circ b \circ d$	0.0078093
1	$a \circ b \circ d \circ c$	0.0041214	13	$c \circ a \circ d \circ b$	0.0024593
2	$a \circ c \circ b \circ d$	0.0078334	14	$c \circ b \circ a \circ d$	0.0076245
3	$a \circ c \circ d \circ b$	0.0024834	15	$c \circ b \circ d \circ a$	0.0054665
4	$a \circ d \circ b \circ c$	0.0025164	16	$c \circ d \circ a \circ b$	0.0022435
5	$a \circ d \circ c \circ b$	0.0020864	17	$c \circ d \circ b \circ a$	0.0019355
6	$b \circ a \circ c \circ d$	0.0080360	18	$d \circ a \circ b \circ c$	0.0024517
7	$b \circ a \circ d \circ c$	0.0040660	19	$d \circ a \circ c \circ b$	0.0020217
8	$b \circ c \circ a \circ d$	0.0077948	20	$d \circ b \circ a \circ c$	0.0023593
9	$b \circ c \circ d \circ a$	0.0056368	21	$d \circ b \circ c \circ a$	0.0019573
10	$b \circ d \circ a \circ c$	0.0034186	22	$d \circ c \circ a \circ b$	0.0019815
11	$b \circ d \circ c \circ a$	0.0030166	23	$d \circ c \circ b \circ a$	0.0016735

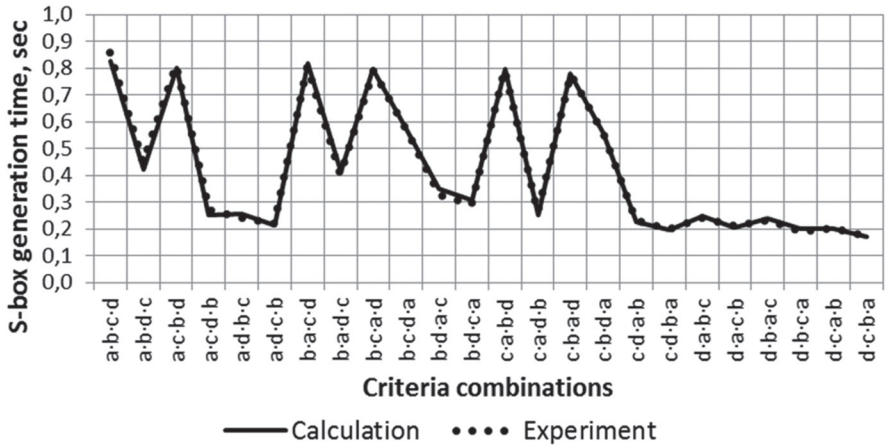


Fig. 1. Graphs of functions $T_{theor}(F_\sigma)$ and $T_{exp}(F_\sigma)$

Two variants of the optimal order of the criteria application on the S-boxes generation were proposed. Software implementation on a single PC allows to reach average 30 minutes generation time for a permutation of 2^8 degree with nonlinearity 104.

References

[1] *Menezes Alfred J.* Handbook of Applied Cryptography [Text] / Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone.

[2] *Gorbenko Ivan Dmytrovych.* Applied cryptology. Theory. Practice. Application [Text]: monograph / Gorbenko I.D., Gorbenko Yu.I.; Kharkiv National University of Radioelectronics, JSC "Institute of Information Technologies". – Kharkiv: Fort, 2012. 868 pp. (in Ukrainian).

[3] *Shannon C.E.* Communication Theory of Secrecy Systems [Text] / C.E Shannon // Bell System Technical Journal. - 1949. - Vol. 28. – pp. 656-715.

[4] *Soroka L.S.* The research of differential properties of block symmetric / L.S. Soroka, O.O. Kuznetsov, I.V. Moskovchenko, S.A. Isayev // Information processing systems. – 2010. - № 6(87). – 286–294 pp. (in Russian).

[5] *Oliynykov R.* An Impact Of S-box Boolean Function Properties To Strength Of Modern Symmetric Block Ciphers / R. Oliynykov, O. Kazymyrov // Радиотехника. – 2011. – Вып. 166. – С. 11-17.

[6] *Kazymyrov O.V.* Methods and Techniques of Generation of Nonlinear Substitutions for Symmetric Encryption Algorithms / O.V. Kazymyrov // The thesis for the scholarly degree of candidate of technical sciences, speciality 05.13.21 – Information security systems. – Kharkiv National University of Radioelectronics, Kharkiv, 2014 (in Russian).

[7] *Kazymyrov O.* A Method For Generation Of High-Nonlinear S-boxes Based On Gradient Descent / O. Kazymyrov, V. Kazymyrova, R. Oliynykov // IACR Cryptology ePrint Archive, 2013. – p. 578.

[8] *Biham E.* Differential Cryptanalysis of DES-like Cryptosystem [Text] / E. Biham, A. Shamir // Journal of Cryptology. – 1991. – Vol. 4. – pp. 3-72.

[9] *Matsui, M.* Linear Cryptoanalysis Method for DES Cipher [Text] / M. Mitsuru // EUROCRYPT'93. - May 1993. – pp. W112–123.

[10] *Courtois N. T.* Cryptanalysis of block ciphers with overdefined systems of equations [Text] / N. T. Courtois, J. Pieprzyk // Advances in Cryptology – ASIACRYPT 2002 : proceedings of the 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1–5, 2002. – Berlin ; Heidelberg : Springer, 2002. – P. 267–287. – (Lecture Notes in Computer Science; vol. 2501).

[11] *Y. Crama and P.L. Hammer.* Boolean Models and Methods in Mathematics, Computer Science and Engineering / Encyclopedia of Mathematics and its Applications. V. 2, Cambridge University Press, 2010.

[12] *K. Nyberg*, Differentially uniform mapping for cryptography, Copyright (c) 1998, Springer-Verlag.

[13] *K. Nyberg*, Linear approximation of block ciphers, Advances in Cryptology – Eurocrypt'94, Lecture Notes in Computer Science, vol. 950, Springer-Verlag, 1994.

[14] *S. Hong, S. Lee, J. Lim, J. Sung, D. Cheon and I. Cho*, Provable Security against Differential and Linear cryptanalysis for SPN Structure. B. Schneier (Ed.): FSE 2000, LNCS 1978, pp. 273-283, 2001.

[15] *Carlet C.* Vectorial Boolean Functions for Cryptography [Text] / C. Carlet // Boolean Models and Methods in Mathematics, Computer Science, and Engineering / ed. Y. Crama, P. Hammer. – Cambridge : Cambridge University Press, 2010. – P. 398–469.

[16] *Nyberg K.* Perfect nonlinear S-boxes [Text] / K. Nyberg // Advances in Cryptology – EUROCRYPT '91 : proceedings of the Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, April 8–11, 1991. Berlin ; Heidelberg : Springer, 1991. – P. 378–386. – (Lecture Notes in Computer Science; vol. 547).

[17] *Kazymyrov O.V.* Vectorial Boolean functions application in substitutions generation for symmetric cryptographic transformation [Text] / O.V. Kazymyrov, R.V. Oliynykov // Information processing systems. – 2012. – № 6 (104). – 97–102 pp. (in Russian).

[18] *Daemen J., Rijmen V.,* AES submission, Document on Rijndael, Version 2, September 1999, pp1-45.

[19] *TESAR P.* A New Method for Generating High Nonlinearity SBoxes [Text] / Petr TESAR // Radioengineering. – 2010. – Vol. 19, № 1. – P. 23–26.

[20] *Roman Oliynykov, Ivan Gorbenko, Oleksandr Kazymyrov, Victor Ruzhentsev, Oleksandr Kuznetsov, Yurii Gorbenko, Oleksandr Dyrda, Viktor Dolgov, Andrii Pushkaryov, Ruslan Mordvinov, Dmytro Kaidalov.* DSTU 7624:2014. National Standard of Ukraine. Information technologies. Cryptographic Data Security. Symmetric block transformation algorithm. Ministry of Economical Development and Trade of Ukraine, 2015 (in Ukrainian).

[21] *Roman Oliynykov, Ivan Gorbenko, Oleksandr Kazymyrov, Victor Ruzhentsev, Artem Boyko, Oleksandr Kuznetsov, Yurii Gorbenko, Viktor Dolgov, Oleksandr Dyrda, Andrii Pushkaryov.* DSTU 7564:2014. National Standard of Ukraine. Information technologies. Cryptographic Data Security. Hash function. Ministry of Economical Development and Trade of Ukraine, 2015 (in Ukrainian).

APPENDIX A

EXAMPLES OF THE OPTIMAL S-BOXES

An example of the optimal S-box based on the Vectorial Boolean function x^{191} (hexadecimal notation)

5c	06	e1	54	39	4c	9b	08	f4	32	c1	22	7a	0b	81	47
79	e2	a5	10	76	e4	86	c0	2a	75	1c	77	f0	1e	3d	a4
91	19	34	95	7d	85	b8	c7	a7	3b	e8	cd	4d	b4	fc	bb
7c	17	42	98	31	ec	bc	f5	5d	fb	02	4f	4e	78	e6	94
e7	30	2c	0d	e0	f3	bf	fa	db	ba	15	1d	40	18	ca	b1
f9	03	d0	d8	ad	44	3a	72	a2	73	df	66	01	fe	be	fd
ef	e3	a9	cb	28	b2	d5	2b	23	2e	99	5e	2d	5b	c8	48
6e	8f	f6	c5	d7	cc	82	65	14	67	c3	1f	26	e9	8c	97
a1	71	8d	ae	1b	ee	c6	68	84	b9	60	87	5f	9c	49	6b
b6	b0	6f	ff	d9	b7	38	cf	a0	eb	8b	4a	f7	3f	3e	da
80	b5	59	0c	6a	1a	96	d2	89	8e	9e	d4	24	25	16	ab
a6	9d	33	70	05	74	63	7b	5a	36	6d	4b	ea	dd	f8	ac
21	2f	69	53	51	f2	7f	92	9a	6c	43	00	d6	50	a3	46
c9	29	90	37	c2	41	7e	09	55	58	20	aa	27	e5	88	64
61	f1	d3	af	d1	11	9f	0a	0e	13	12	3c	dc	35	ed	45
93	b3	c4	bd	57	62	52	8a	a8	0f	04	ce	de	07	83	56

An example of the optimal S-box based on the Vectorial Boolean function x^{254} (hexadecimal notation)

1a	a8	96	a1	a6	97	80	26	c1	f2	32	7f	8b	c9	f0	c3
64	79	27	10	43	4c	6c	9b	c4	ac	d8	ea	b2	9e	d5	8e
7d	02	c7	0e	17	83	cb	07	61	e0	84	fa	3e	03	7a	24
be	8c	19	6f	1d	f7	b8	68	b3	e6	db	78	d1	cd	0a	a7
a3	b4	f1	fc	3f	5d	57	4f	42	8d	ca	71	5f	ab	66	d9
a0	72	16	ad	9c	2c	49	30	bb	99	31	ce	34	3c	fe	d3
18	d0	ef	cf	82	36	cc	6d	d6	b7	c6	5c	58	86	20	e4
75	7e	87	41	8a	53	1f	21	63	67	74	37	0c	2d	91	48
54	df	38	73	44	b1	ae	40	2a	62	fb	c5	f5	1c	4d	af
45	70	dc	95	04	ec	0f	bc	fd	6b	0d	a2	2e	93	3a	eb
59	aa	c0	55	06	ed	e1	50	4b	d7	5a	65	4a	e3	25	a9
c8	b5	5b	76	47	05	14	22	2f	81	9a	0b	c2	77	09	35
90	1e	e9	3d	7b	f4	51	92	29	33	b0	9d	23	d2	12	6a
89	2b	d4	28	dd	f6	f8	8f	08	69	39	00	a5	e5	e2	88
52	1b	f9	da	bf	b9	f3	60	13	ff	56	7c	de	6e	5e	85
3b	9f	e8	11	4e	bd	94	a4	46	ba	ee	15	98	01	b6	e7

Manuscript received October, 6, 2015

Rodinko Mariia Yuriivna, Master Student of Information Technologies Security Department at KNURE. Scientific interests: symmetric cryptography and cryptanalysis.

Oliynykov Roman Vasylovych, Doctor of Technical Sciences, Professor at Information Technologies Security Department at KNURE. Scientific interests: symmetric cryptography and cryptanalysis, network security.

Hrinenko Tetiiana Oleksiivna, Candidate of Technical Sciences (PhD), Associate Professor at Information Technologies Security Department at KNURE. Scientific interests: information technologies security.

УДК 681.3.06

Усовершенствование метода генерации оптимальных S-блоков / М.Ю. Родинко, Р.В. Олейников, Т.А. Гриненко // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 315–320.

Известный метод генерации S-блоков с высокой нелинейностью, основанный на градиентном спуске, предполагает последовательное применение нескольких критериев к каждой сформированной подстановке. В данной работе представлено усовершенствование рассматриваемого метода путем выбора порядка применения критериев, для которого требуемая вычислительная мощность для генерации подстановок будет наименьшей. Предложенная модификация позволяет сгенерировать байтовую подстановку с нелинейностью 104, алгебраическим иммунитетом 3 и 8-равномерностью в пределах приблизительно 30 минут на персональном компьютере.

Ключевые слова: S-блок, нелинейность, алгебраический иммунитет, векторная булева функция.

Табл.: 6. Ил.: 1. Библиогр.: 21 наим.

УДК 681.3.06

Удосконалення методу генерації оптимальних S-блоків / М.Ю. Родінко, Р.В. Олійников, Т.О. Гріненко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 315–320.

Відомий метод генерації S-блоків з високою нелінійністю, заснований на градієнтному спуску, передбачає послідовне застосування декількох критеріїв до кожної сформованої підстановки. У даній роботі наведено удосконалення методу, що розглядається, шляхом вибору порядку застосування критеріїв, для якого потрібна обчислювальна потужність для генерації підстановок буде найменшою. Запропонована модифікація дозволяє згенерувати байтову підстановку з нелінійністю 104, алгебраїчним імунітетом 3 та 8-рівномірністю в межах приблизно 30 хвилин на персональному комп'ютері.

Ключові слова: S-блок, нелінійність, алгебраїчний імунітет, векторна булева функція.

Табл.: 6. Іл.: 1. Бібліогр.: 21 найм.

УДК 004.056.55

ДОСЛІДЖЕННЯ ПОТОКОВИХ АЛГОРИТМІВ ШИФРУВАННЯ – ПЕРЕМОЖЦІВ МІЖНАРОДНОГО ПРОЕКТУ ESTREAM

О.О. КУЗНЕЦОВ, Д.В. ІВАНЕНКО, Р.І. МОРДВІНОВ

Аналізуються потокові алгоритми шифрування, які визначено переможцями міжнародного проекту eSTREAM. Досліджується базова структура алгоритмів-переможців, застосовані функціональні зав'язки, шари лінійних та нелінійних перетворень тощо. Проводяться експериментальні дослідження статистичної безпеки за методикою тестування NIST Special Publication 800-22. Встановлено, що вихідні послідовності досліджуваних шифрів відповідають сучасним вимогам, розглянуті криптоперетворення можуть бути застосовані під час розробки різних варіантів побудови національного стандарту потокового шифрування України.

Ключові слова: криптографічне перетворення, потоковий шифр, статистичні тести.

ВСТУП

Міжнародний проект eSTREAM проводився у 2004 – 2008 роках Європейським Союзом та був призначений для виявлення нового потокового шифру, придатного для широкого застосування [1–7]. Його було започатковано через невдачу всіх шістьох потокових шифрів, поданих свого часу для проекту NESSIE [8, 9].

Проект eSTREAM складався з декількох фаз і його метою було знаходження алгоритмів, придатних для різних варіантів застосування. Для цього шифри в проекті eSTREAM згруповані за такими профілями:

– профіль 1 – шифри для програмного забезпечення з високими вимогами до пропускної здатності;

– профіль 2 – шифри для апаратного забезпечення з обмеженими ресурсами (пам'яті, кількості транзисторів або інших пристроїв) або споживання енергії.

На сьогоднішній день відібрані 7 шифрів, які відповідають встановленим вимогам (див. табл. 1) [1–7]. Наведені шифри не мають обмежень щодо застосування, в тому числі патентних чи будь-яких інших [1–7].

Таблиця 1

Профіль 1	Профіль 2
HC-128	Grain
Rabbit	MICKEY
Salsa20/12	Trivium
SOSEMANUK	

Метою роботи є аналіз та дослідження структури та основних перетворень потокових алгоритмів шифрування з міжнародного проекту eSTREAM, дослідження показників їхньої статистичної безпеки та обґрунтування можливості застосування певних технічних рішень під час розробки різних варіантів побудови національного стандарту потокового шифрування України.

1. ДОСЛІДЖЕННЯ ПРОГРАМНО-ОРІЄНТОВАНИХ ПОТОКОВИХ ШИФРІВ

До першого профілю міжнародного проекту eSTREAM відносяться криптоалгоритми, які спрямовано на програмну реалізацію з високими вимогами до швидкості формування псевдовипадкових послідовностей (ПВП). До цього профілю віднесено: HC-128; Rabbit; Salsa 20/12; SOSEMANUK.

Розглянемо більш докладно кожен алгоритм, проаналізуємо особливості побудови та можливості застосування.

1.1. Потоковий шифр HC-128

Потоковий шифр HC-128 є спрощеною версією HC-256 [2] для 128-бітного рівня безпеки. Це простий, безпечний, програмно-орієнтований шифр з ефективною реалізацією і може вільно використовуватися. Він складається з двох секретних таблиць, кожна з яких містить 512 32-бітових елементів. На кожному кроці виконується оновлення одного елементу таблиці з нелінійною функцією зворотного зв'язку. Всі елементи двох таблиць оновлюється кожні 1024 кроків. На кожному кроці, один 32-бітовий вихід генерується з нелінійної функції вихідної фільтрації.

Шифр призначений для формування ключового потоку довжиною до 2^{64} бітів, який виробляється з 128-бітного ключа і 128-бітного вектора ініціалізації. В ході реалізації HC-128 використовуються операції з табл. 2.

Дві таблиці, які використовуються в HC-128, позначаються як P і Q. Ключ і вектор ініціалізації позначаються як K і IV. Ключовий потік, який генерується шифром, позначається як s.

Шифр HC-128 використовує шість функцій (так само, як і шифр HC-256). Функції $f_1(x)$ і $f_2(x)$ збігаються з функціями $\sigma_0^{\{256\}}(x)$ і $\sigma_1^{\{256\}}(x)$, які використовуються в алгоритмі ґешування SHA-256. Для функції $h_1(x)$ як S-блок використовується таблиця Q. Для функції $h_2(x)$ як S-блок використовується таблиця P. Функції мають вигляд:

$$f_1(x) = (x >>> 7) \oplus (x >>> 18) \oplus (x >> 3);$$

$$f_2(x) = (x >>> 17) \oplus (x >>> 19) \oplus (x >> 10);$$

$$g_1(x, y, z) = ((x >>> 10) \oplus (z >>> 23)) + (y >>> 8);$$

$$g_2(x, y, z) = ((x <<< 10) \oplus (z <<< 23)) + (y <<< 8);$$

$$h_1(x) = Q[x_0] + Q[256 + x_2];$$

$$h_2(x) = P[x_0] + P[256 + x_2],$$

де $x = x_3 \| x_2 \| x_1 \| x_0$, x – це 32-бітне слово, x_0, x_1, x_2, x_3 – чотири байти.

Байти x_3 і x_0 позначаються як найбільш значущий і молодший байт x, відповідно.

Таблиця 2

	Зміст позначення
+	додавання за 2^{32}
-	віднімання за модулем 512
$\oplus$	побітне виключне АБО (XOR)
$\|\|$	конкатенація
>>	зсув праворуч на вказане число бітів
<<	зсув ліворуч на вказане число бітів
>>>	циклічний зсув праворуч на вказане число бітів
<<<	циклічний зсув ліворуч на вказане число бітів
P	Таблиця з 512 32-бітними елементами. Кожен елемент позначається як $P[i]$, $0 \le i \le 511$
Q	Таблиця з 512 32-бітними елементами. Кожен елемент позначається як $Q[i]$, $0 \le i \le 511$
K	128-бітний ключ потокового шифру HC-128
IV	128-бітний вектор ініціалізації
s	Ключовий потік. На кожному кроці формується 32-бітний вихід s_i, $s = s_0 \| s_1 \| s_2 \| ...$

Алгоритм ініціалізації. Процес ініціалізації складається з розширення ключа і вектора ініціалізації в P і Q (за аналогією з установкою повідомлень в SHA-256) і запуску шифрування 1024 разів без генерації вихідного результату (виходи використовуються для оновлення P і Q):

1. Нехай $K = K_0 \,||\, K_1 \,||\, K_2 \,||\, K_3$ і $IV = IV_0 \,||\, IV_1 \,||\, IV_2 \,||\, IV_3$, де кожен K_i і IV_i позначає 32-розрядне число. Нехай $K_{i+4} = K_i$, і $IV_{i+4} = IV_i$ для $0 \le i < 4$. Ключ і вектор ініціалізації розширюються в масив W_i $(0 \le i \le 1279)$ таким чином:

$$W_i = \begin{cases} K_i, & 0 \le i \le 7; \\ IV_{i-8}, & 8 \le i \le 15; \\ f_2(W_{i-2}) + W_{i-7} + f_1(W_{i-15}) + W_{i-16} + i, & \\ \qquad 16 \le i \le 1279. & \end{cases}$$

2. Оновлення таблиці P і Q з масиву W.

$P[i] = W_{i+256}$ для $0 \le i \le 511$;

$Q[i] = W_{i+768}$ для $0 \le i \le 511$.

3. Виконання 1024 кроків шифрування і використання виходів для заміни елементів таблиці таким чином:

Для всіх від i = 0 до 511:

$P[i] = (P[i] + g1(P[i-3],$

$P[i-10], P[i-511])) \oplus h1(P[i-12]);$

Для всіх від $i = 0$ до 511:

$Q[i] = (Q[i] + g2(Q[i-3],$

$Q[i-10], Q[i-511])) \oplus h2(Q[i-12]).$

Після виконання наведених кроків процес ініціалізації завершується і шифр готовий генерувати ключовий потік.

Алгоритм генерації ключового потоку. На кожному кроці оновлюється один з елементів таблиці і генерується один 32-бітовий вихід. Кожен S-блок використовується для генерації тільки 512 виходів, то він оновлюється протягом наступних 512 кроків.

Головною особливістю в структурі потокового шифру HC-128 (та його більш потужної версії HC-256) є застосування логічних функцій $f_1(x)$ і $f_2(x)$ та процесу ініціалізації, які запозичені з алгоритму гешування SHA-256. Це дає можливість застосувати вже досліджені та добре перевірені часом обчислювально ефективні елементи криптоперетворення та досягти певних показників компактності реалізації та можливості розпаралелювання. Як результат, потоковий шифр HC-128 є дуже зручним для реалізації на сучасних мікропроцесорах, причому залежність між окремими операціями зведена до мінімуму: три послідовні кроки алгоритму можуть бути обчислені паралельно. Отримана можливість розпаралелювання призводить до високих показників ефективності на сучасних процесорах [2].

1.2. Потоковий шифр Rabbit

Алгоритм Rabbit – високошвидкісний потоковий шифр, який вперше був представлений в лютому 2003 року на 10-му симпозіумі FSE. У травні 2005, цей шифр був відправлений на конкурс eStream [4]. Розробниками алгоритму Rabbit є Martin Boesgaard, Mette Vesterager, Thomas Pedersen, Jesper Christiansen і Ove Scavenius.

Основним компонентом шифру є генератор бітового потоку, який формує 128 бітів за ітерацію. Перевага шифру Rabbit полягає в ретельному перемішуванні його внутрішніх станів між двома послідовними ітераціями. Функція перемішування повністю заснована на арифметичних операціях, доступних на сучасних процесорах, тобто S-блоки підстановок і пошукові таблиці не потрібні для реалізації шифру.

Алгоритм Rabbit можна стисло описати наступним чином. Як вхідні дані алгоритм приймає 128-бітний секретний ключ і 64-бітний вектор ініціалізації IV (за бажанням) і формує для кожної ітерації вихідний блок 128 псевдовипадкових бітів з комбінації бітів внутрішнього стану. Шифрування / розшифрування виконується операцією XOR псевдовипадкових даних з відкритим / зашифрованим текстом. При викладенні специфікації потокового шифру Rabbit використовують позначення з табл. 4 [4].

Таблиця 4

	Зміст позначення
$\oplus$	Операція логічного виключного АБО (XOR)
&	Операція логічного І (AND)
>>	Оператор зсуву праворуч на вказане число бітів
<<	Оператор зсуву ліворуч на вказане число бітів
>>>	Оператор циклічного зсуву праворуч на вказане число бітів
<<<	Оператор циклічного зсуву ліворуч на вказане число бітів
◊	Конкатенація двох бітових послідовностей
$A^{[g..h]}$	Позначення послідовності бітів з g по h змінної A

Внутрішній стан потокового шифру складається з 513 бітів. З них 512 бітів розділені між вісьмома 32-бітовими змінними стану $x_{j,i}$ і вісьмома 32-бітовими змінними лічильника $c_{j,i}$. В цих позначеннях $x_{j,i}$ є змінною стану j-ї підсистеми на i-й ітерації, а $c_{j,i}$ є змінною лічильника j-ї підсистеми на i-й ітерації, відповідно. Застосовується також один лічильник бітів перенесення, що по-

значається як $\phi_{7,i}$ і який має зберігатися між ітераціями. Цей лічильник бітів перенесення встановлюється на нуль. Вісім змінних стану і вісім змінних лічильників встановлюються з ключа при ініціалізації.

Схема установки ключа. Алгоритм шифрування Rabbit ініціюється розширенням 128-бітного ключа в обидві складові: вісім змінних стану і вісім змінних лічильників. Існує взаємно однозначна відповідність між ключем і початковим станом цих змінних.

Установку ключа можна розділити на три етапи: розширення ключа, система ітерацій, модифікація лічильника.

Етап розширення ключа гарантує взаємно однозначну відповідність між ключем стану і ключем лічильника, який запобігає надлишку ключів. Він також розподіляє біти ключа оптимальним чином для всіх ітерацій.

На цьому етапі ключ поточного шифрування, який позначається як $K^{[127...0]}$, ділиться на вісім частин, т.з. субключів:

$$k_0 = K^{[15..0]},\ k_1 = K^{[31..16]},\ \ldots,\ k_7 = K^{[127..112]}.$$

Система ітерацій гарантує, що після однієї ітерації функції наступного стану кожен біт ключа вплине на всі вісім змінних стану. Це також гарантує, що після другої ітерації функції наступного стану всі біти ключа вплинуть на всі біти стану з імовірністю 0,5.

Змінні стану і лічильників зініціюються з цих субключів таким чином:

$$x_{j,0} = \begin{cases} k_{(j+1 \bmod 8)} \lozenge k_j, & \text{для всіх парних } j; \\ k_{(j+5 \bmod 8)} \lozenge k_{(j+4 \bmod 8)}, & \text{для всіх непарних } j; \end{cases}$$

$$A_{j,0} = \begin{cases} k_{(j+4 \bmod 8)} \lozenge k_{(j+5 \bmod 8)}, & \text{для всіх парних } j; \\ k_j \lozenge k_{(j+1 \bmod 8)}, & \text{для всіх непарних } j. \end{cases}$$

Схема установки ключа повторюється чотири рази, відповідно до функції наступного стану (визначеної далі). Це спрямовано на зменшення кореляції між бітами ключа і бітами змінних внутрішнього стану.

Модифікація лічильника. Навіть, якщо лічильники будуть відомі зловмисникові, модифікація лічильника значно ускладнює відновлення ключа шляхом інвертування лічильника системи, оскільки в цьому випадку будуть потрібні відомості щодо змінних стану. Крім того, модифікація порушує взаємно однозначне співвідношення між ключем і лічильником.

Змінні лічильника модифікуються відповідно до такого правила:

$$c_{j,4} = c_{j,4} \oplus x_{(j+4 \bmod 8),4},$$

яке виконується для всіх j.

Схема установки вектора ініціалізації. Позначимо внутрішній стан після схеми установки ключа як майстер стан і нехай копія цього майстер стану буте модифікована згідно зі схемою вектора ініціалізації *IV*. Схема установки *IV* функціонує шляхом зміни стану лічильника як функція від *IV*. Це робиться шляхом операції XOR над 64-бітним вектором ініціалізації *IV* на всіх 256 бітах стану лічильника. Ці 64 біти вектора ініціалізації *IV* позначаються як $IV^{[63..0]}$. Лічильники модифікуються таким чином:

$$\begin{aligned} c_{0,4} &= c_{0,4} \oplus IV[31..0], \\ c_{1,4} &= c_{1,4} \oplus (IV[63..48] \lozenge IV[31..16]), \\ c_{2,4} &= c_{2,4} \oplus IV[63..32], \\ c_{3,4} &= c_{3,4} \oplus (IV[47..32] \lozenge IV[15..0]), \\ c_{4,4} &= c_{4,4} \oplus IV[31..0], \\ c_{5,4} &= c_{5,4} \oplus (IV[63..48] \lozenge IV[31..16]), \\ c_{6,4} &= c_{6,4} \oplus IV[63..32], \\ c_{7,4} &= c_{7,4} \oplus (IV[47..32] \lozenge IV[15..0]). \end{aligned}$$

Наведена схема модифікації повторюється чотири рази, щоб усі біти стану стали нелінійно залежними від усіх бітів вектора ініціалізації *IV*. Модифікація змінних лічильника за вектором ініціалізації *IV* гарантує, що застосування всіх 2^{64} різних векторів ініціалізації *IV* призведе до формування унікальних ключових потоків.

Функція наступного стану. Ядром алгоритму є ітеративна схема, яка визначена за допомогою таких рівнянь:

$$\begin{aligned} x_{0,i+1} &= g_{0,i} + (g_{7,}{}_i <<< 16) + (g_{6,i} <<< 16); \\ x_{1,i+1} &= g_{1,i} + (g_{0,i} <<< 8) + g_{7,i}; \\ x_{2,i+1} &= g_{2,i} + (g_{1,\,i} <<< 16) + (g_{0,i} <<< 16); \\ x_{3,i+1} &= g_{3,i} + (g_{2,\,i} <<< 8) + g_{1,i}; \\ x_{4,i+1} &= g_{4,i} + (g_{3,\,i} <<< 16) + (g_{2,\,i} <<< 16) \\ x_{5,i+1} &= g_{5,i} + (g_{4,i} <<< 8) + g_{3,i}; \\ x_{6,i+1} &= g_{6,i} + (g_{5,\,i} <<< 16) + (g_{4,i} <<< 16); \\ x_{7,i+1} &= g_{7,i} + (g_{6,\,i} <<< 8) + g_{5,i}, \end{aligned} \qquad (5)$$

де

$$g_{j,i} = ((x_{j,i} + c_{j,i+1})^2 \oplus ((x_{j,i} + c_{j,i} + 1)^2 >> 32)) \bmod 2^{32}$$

і всі додавання виконуються за модулем 2^{32}.

Наведені аналітичні вирази в поєднанні між собою можна в графічному вигляді подати на рис. 1. Перед ітерацією лічильники збільшуються, як це описано далі.

Система лічильників. Динаміка зміни лічильників визначається таким чином:

$$\begin{aligned} c_{0,i+1} &= c_{0,i} + a_0 + \phi_{7,i} \bmod 2^{32}; \\ c_{1,i+1} &= c_{1,i} + a_1 + \phi_{0,i+1} \bmod 2^{32}; \\ c_{2,i+1} &= c_{2,i} + a_2 + \phi_{1,i+1} \bmod 2^{32}; \\ c_{3,i+1} &= c_{3,i} + a_3 + \phi_{2,i+1} \bmod 2^{32}; \\ c_{4,i+1} &= c_{4,i} + a_4 + \phi_{3,i+1} \bmod 2^{32}; \\ c_{5,i+1} &= c_{5,i} + a_5 + \phi_{4,i+1} \bmod 2^{32}; \\ c_{6,i+1} &= c_{6,i} + a_6 + \phi_{5,i+1} \bmod 2^{32}; \\ c_{7,i+1} &= c_{7,i} + a_7 + \phi_{6,i+1} \bmod 2^{32}, \end{aligned}$$

де лічильник біту перенесення $\phi_{j,i+1}$ задається як

$$\phi_{j,i+1} = \begin{cases} 1, & c_{0,i} + a_0 + \phi_{7,i} \ge 2^{32} \wedge j = 0; \\ 1, & c_{j,i} + a_j + \phi_{j-1,i+1} \ge 2^{32} \wedge j > 0; \\ 0, & \textit{otherwise}. \end{cases}$$

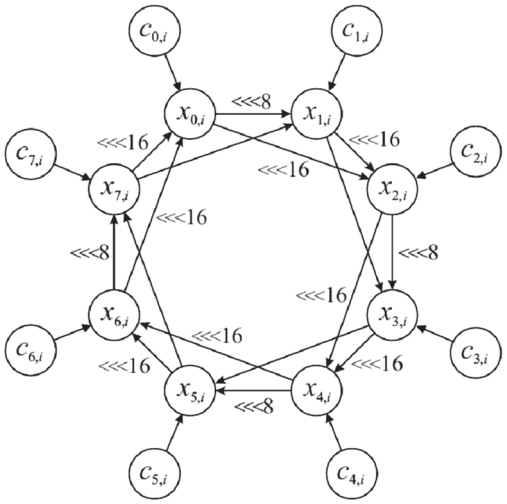


Рис. 1

Постійні a_j визначаються як:

$a_0 = 0x4D34D34D$; $a_1 = 0xD34D34D3$;

$a_2 = 0x34D34D34$; $a_3 = 0x4D34D34D$;

$a_4 = 0xD34D34D3$; $a_5 = 0x34D34D34$;

$a_6 = 0x4D34D34D$; $a_7 = 0xD34D34D3$.

Схема виходу. Після кожної ітерації вихідний ключовий потік формується таким чином:

$$s_i^{[15..0]} = x_{0,i}^{[15..0]} \oplus x_{5,i}^{[31..16]};\ s_i^{[31..16]} = x_{0,i}^{[31..16]} \oplus x_{3,i}^{[15..0]};$$

$$s_i^{[47..32]} = x_{2,i}^{[15..0]} \oplus x_{7,i}^{[31..16]};\ s_i^{[63..48]} = x_{2,i}^{[31..16]} \oplus x_{5,i}^{[15..0]};$$

$$s_i^{[78..64]} = x_{4,i}^{[15..0]} \oplus x_{1,i}^{[31..16]};\ s_i^{[95..80]} = x_{4,i}^{[31..16]} \oplus x_{7,i}^{[15..0]};$$

$$s_i^{[111..96]} = x_{6,i}^{[15..0]} \oplus x_{3,i}^{[31..16]};\ s_i^{[127..112]} = x_{6,i}^{[31..16]} \oplus x_{1,i}^{[15..0]},$$

де s_i є 128-бітний блок ключового потоку на i-й ітерації.

Схема шифрування/розшифрування. Сформовані біти ключового потоку додаються (операцією XOR) до відкритого/зашифрованого тексту для шифрування/розшифрування:

$$c_i = p_i \oplus s_i,\ p_i = c_i \oplus s_i,$$

де c_i і p_i позначають i-й 128-бітний блок зашифрованого тексту і блок відкритого тексту, відповідно.

Головною особливістю в структурі потокового шифру Rabbit є застосування логічних функцій (5), які використовують модульне додавання та піднесення до квадрата (за модулем 2^{32}). Це дозволяє значно спростити схему обчислення значень вихідного ключового потоку, акцентувати приріст швидкості генерації гами.

Таким чином, алгоритм Rabbit був розроблений для забезпечення високої швидкодії, зокрема для того, щоб бути швидшим, ніж зазвичай використовувані шифри із розміром ключа 128 біт для шифрування до 2^{64} блоків тексту. Для зловмисника, який не знає ключ, можливість розрізнити 2^{64} блоків шифру від виходу генератора справжніх випадкових біт має потребувати не менше дій, ніж потрібно було б для перебору 2^{128} ключів [4].

1.3. Потоковий шифр Salsa 20

Salsa 20 є потоковим шифром, який був розроблений Daniel J. Bernstein [5]. Він має дві модифікації Salsa 20/8 і Salsa 20/12, 8-м і 12-ть раундів за шифрування, відповідно, замість 20 оригінальних в Salsa 20. Вихідний алгоритм Salsa 20 був зроблений з великим запасом стійкості, тоді як Salsa 20/8 показує гарні результати у швидкодії, випереджаючи у більшості випадків багато інших шифрів, у тому числі AES і RC4 [5]. Алгоритм використовує геш-функцію з 20 циклами. Основні її перетворення нагадують алгоритм AES. При викладенні специфікації потокового шифру Salsa 20 використовують позначення, що наведено в табл. 5 [5].

Таблиця 5

	Зміст позначення
$\oplus$	Операція побітного додавання за модулем 2 (XOR)
$+$	Операція додавання 32-х бітних чисел за модулем 2^{32}
$<<<$	Оператор циклічного зсуву ліворуч на вказане число бітів

Функція ***quarterround***. Основним функціональним блоком системи шифрування є перетворення *quarterround*(y) над 4-ма словами. З цього перетворення будуються більш загальні та складні перетворення.

Сутність перетворення *quarterround*(y) полягає в тому, що для кожного слова складають два попередніх, отриману суму циклічно зсувають на певну кількість біт і побітово підсумовують результат з обраним словом. Наступні операції проводяться з новими значеннями слів.

Якщо y – це послідовність 4-х слів, тобто $y = (y_0, y_1, y_2, y_3)$, тоді обчислення функції *quarterround*(y) $= (z_0, z_1, z_2, z_3)$ полягає в тому, що:

$$z_1 = y_1 \oplus ((y_0 + y_3) <<< 7),$$
$$z_2 = y_2 \oplus ((z_1 + y_0) <<< 9),$$
$$z_3 = y_3 \oplus ((z_2 + z_1) <<< 13),$$
$$z_0 = y_0 \oplus ((z_3 + z_2) <<< 18).$$

Схема обчислення функції *quarterround*(y) $= (z_0, z_1, z_2, z_3)$ наведена на рис. 2. Можна розглядати цю функцію як перетворення слів y_1, y_2, y_3 і y_4. Кожне з перетворень обернене, як і функція в цілому.

Функція ***rowround***. На вхід перетворення поступає 16 слів у вигляді матриці 4×4. Кожен рядок цієї «матриці» обробляється функцією *quarterround*(y). Слова з рядка беруться по порядку, починаючи з i-го для i-го рядка, де $i = \{0, 1, 2, 3\}$.

Якщо $y = (y_0, y_1, y_2, ..., y_{15})$ – послідовність з 16 слів, тоді обчислення функції *rowround*(y) $= (z_0, z_1, z_2, ..., z_{15})$ полягає в наступному:

$$(z_0, z_1, z_2, z_3) = quarterround(y_0, y_1, y_2, y_3),$$
$$(z_5, z_6, z_7, z_4) = quarterround(y_5, y_6, y_7, y_4),$$
$$(z_{10}, z_{11}, z_8, z_9) = quarterround(y_{10}, y_{11}, y_8, y_9),$$
$$(z_{15}, z_{12}, z_{13}, z_{14}) = quarterround(y_{15}, y_{12}, y_{13}, y_{14}).$$

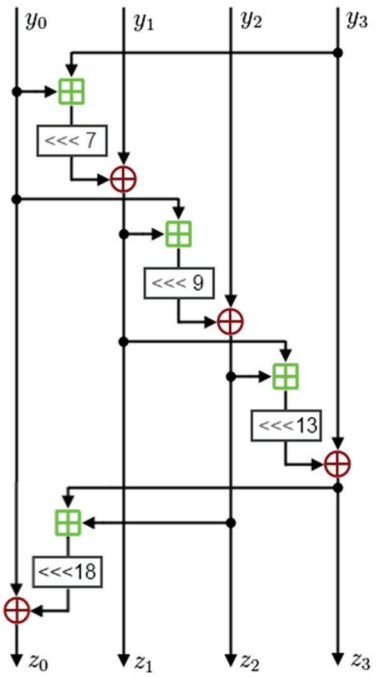


Рис. 2

Функція ***columnround***. Перетворення аналогічне функції *quarterround(y)*, аргументом якої є значення стовпців (колонок) матриці, починаючи з *j*-го для *j*-го стовпця, де $j = \{0,1,2,3\}$.

Функція $columnround(y) = (z)$ також оперує послідовністю 16-и слів так, що

$$(z_0, z_4, z_8, z_{12}) = quarterround(y_0, y_4, y_8, y_{12}),$$
$$(z_5, z_9, z_{13}, z_1) = quarterround(y_5, y_9, y_{13}, y_1),$$
$$(z_{10}, z_{14}, z_2, z_6) = quarterround(y_{10}, y_{14}, y_2, y_6),$$
$$(z_{15}, z_3, z_7, z_{11}) = quarterround(y_{15}, y_3, y_7, y_{11}).$$

Еквівалентна формула:

$$(z_0, z_4, z_8, z_{12}, z_1, z_5, z_9, z_{13}, z_2, z_6, z_{10}, z_{14}, z_3, z_7, z_{11}, z_{15}) = rowround(y_0, y_4, y_8, y_{12}, y_1, y_5, y_9, y_{13}, y_2, y_6, y_{10}, y_{14}, y_3, y_7, y_{11}, y_{15}).$$

Функція ***doubleround***. Функція *doubleround(y)* є послідовним застосуванням функцій *columnround* а потім *rowround*, тобто:

$$doubleround(y) = rowround(columnround(y)).$$

Функція ***littleendian***. Salsa 20 використовує запис слова, яке починається з молодшого байта. Для цього введено окреме перетворення *littleendian(b)*.

Нехай $b = (b_0, b_1, b_2, b_3)$ – 4-байтова послідовність, тоді функція $littleendian(b)$ повертає слово, таке що $littleendian(b) = b_0 + 2^8 b_1 + 2^{16} b_2 + 2^{24} b_3$. Слід відмітити, що ця функція є оберненою.

Геш-функція ***Salsa20(x)***. На вхід функції подається 64-байтова послідовність і повертає 64-бійтову послідовність. Обчислення функції $Salsa20(x)$ починається з вектора

$$x = (x[0], x[1], ..., x[63]),$$

де

$$x_0 = littleendian(x[0], x[1], x[2], x[3]),$$
$$x_1 = littleendian(x[4], x[5], x[6], x[7]),$$
$$x_2 = littleendian(x[8], x[9], x[10], x[11]),$$
$$...$$
$$x_{15} = littleendian(x[60], x[61], x[62], x[63]).$$

Тут застосовано такі позначення: $x[i]$ – байти вхідної послідовності x, x_j – слова, що використовуються в описаних вище функціях.

Визначимо $(z_0, z_1, ..., z_{15}) = doubleround^{10}(x_0, x_1, ..., x_{15})$, тоді значення функції $Salsa20(x)$ обчислюється як:

$$littleendian^{-1}(z_0 + x_0);$$
$$littleendian^{-1}(z_1 + x_1);$$
$$littleendian^{-1}(z_2 + x_2);$$
$$...$$
$$littleendian^{-1}(z_{15} + x_{15}).$$

Функція поширення ***Salsa20$_k$(n)***. На вхід функції $Salsa20_k(n)$ поступає 32-байтова або 16-байтова послідовність k і 16-байтова послідовність n та повертає 64-байтову послідовність. Для цього використовуються константи:

$$\sigma_0 = (101, 120, 112, 97),$$
$$\sigma_1 = (110, 100, 32, 51),$$
$$\sigma_2 = (50, 45, 98, 121),$$
$$\sigma_3 = (116, 101, 32, 107).$$

Якщо k_0, k_1, n є 16-байтовими послідовностями, тоді

$$Salsa20_{k0,k1}(n) = Salsa20(\sigma_0, k_0, \sigma_1, n, \sigma_2, k_1, \sigma_3).$$

Позначимо

$$\tau_0 = (101, 120, 112, 97),$$
$$\tau_1 = (110, 100, 32, 49),$$
$$\tau_2 = (54, 45, 98, 121),$$
$$\tau_3 = (116, 101, 32, 107).$$

Якщо k, n є 16-байтовими послідовностями, тоді

$$Salsa20_k(n) = Salsa20(\tau_0, k, \tau_1, n, \tau_2, k).$$

Функція шифрування ***Salsa20***. Шифртекстом l-байтової послідовності m для $l \in \{0,1, ... 2^{70}\}$ буде

$$Salsa20_k(v) \oplus m,$$

де v – унікальний 8-байтовий номер (*nonce*).

Шифртекст має розмір l байт, так само як і вихідний текст.

$Salsa20_k(v)$ – ключовий потік 2^{70} байт:

$$Salsa20_k(v, \underline{0}), Salsa20_k(v, \underline{1}), Salsa20_k(v, \underline{2}), ... , Salsa20_k(v, \underline{2^{64} - 1}),$$

де $\underline{i}$ – унікальна послідовність з 8 байт $(i_0, i_1, ..., i_7)$ така, що

$$i = i_0 + 2^8 i_1 + ... + 2^{56} i_7.$$

Відповідно

$$Salsa20_k(v) \oplus (m[0], m[1], .., m[l\text{-}1]) = (c[0], c[1], ..., c[l\text{-}1]),$$

де $c[i] = m[i] \oplus Salsa20_k(v, \lfloor i/64 \rfloor)[i \bmod 64]$.

Таким чином, алгоритм *Salsa*20 практично не має накладних обчислень. Багато шифросистем розраховують попередні обчислення, результати яких мають зберігатися в кеші першого рівня (L1) процесора. Оскільки вони залежать від ключа у випадку його оновлення всі обчислення потрібно здійснити заново. У *Salsa*20 ж досить просто завантажити ключ у пам'ять.

1.4. Потоковий шифр SOSEMANUK

Sosemanuk – це новий синхронний програмно-орієнтований потоковий шифр, який відповідає першому профілю міжнародного конкурсу eECRYPT [6]. Його довжина ключа може бути обрана між 128 і 256 бітами. Шифр працює з 128 бітовим початковим значенням, при цьому, як стверджується розробниками алгоритму [6], будь-яка довжина ключа досягає 128-бітного захисту.

Алгоритм Sosemanuk використовує деякі основні принципи потокового шифру Snow 2.0 [10] і деякі перетворення, отримані з блокового симетричного шифру (БСШ) SERPENT [11]. Sosemanuk спрямований на поліпшення Snow 2.0 як в сенсі безпеки, так і в сенсі ефективності реалізації. Зокрема, Sosemanuk використовує швидку процедуру установки вектора ініціалізації IV. Він також вимагає зменшеної кількості статистичних даних, що дає більш високу продуктивність на декількох архітектурах.

SERPENT і його похідні. SERPENT [11] – є БСШ, який відповідає умовам конкурсу AES на визначення стандарту БСШ США. Алгоритм SERPENT працює над блоками з 128 біт, які розділені на чотири 32-бітові блоки кожен. Пронумеруємо вхідні і вихідні четвірки алгоритму SERPENT від 0 до 3 і запишемо їх в порядку: (Y_3, Y_2, Y_1, Y_0). Y_0 є мінімальним значущим словом і містить мінімальні значущі біти з 32-х 4-бітових слів, що входять в S-блок шифру SERPENT. У той час як вихід алгоритму SERPENT записується в 16 байт, значення Y_i записуються від молодшого до старшого (перший – найменш значимий байт), Y_0 виводиться першим, далі Y_1, і т. д.

З алгоритму SERPENT визначаються два примітива, це Serpent1 і Serpent24 [11].

Serpent1. Раунди алгоритму БСШ SERPENT складаються в такому порядку:

– побітова операція XOR;

– застосування S-блоку (що виражається як набір бітових перестановок між чотирма використовуваними 32-бітними словами);

– лінійна бієктивна трансформація (яка складається з декількох операцій XOR, зсувів і поворотів).

Таким чином, примітив Serpent1 – це один раунд алгоритму БСШ SERPENT без додавання ключа і лінійної трансформації. SERPENT використовує 8 унікальних S-блоків, пронумерованих, відповідно, з S_0 до S_7 на 4-бітових словах. Serpent1 визначається як застосування S_2, в режимі «bitslice», він приймає чотири 32-бітних слова на вхід і надає чотири 32-бітних слова на вихід.

Serpent24. Serpent24 – це алгоритм SERPENT, зменшений до 24 раундів замість 32 раундів повної версії SERPENT. Serpent24 еквівалентний першим 24 раундів SERPENT, де останній, 24-й раунд є повним і включає повний раунд з лінійною трансформації і операцію XOR з 25 підключами. Іншими словами, 24-й раунд Serpent24 є еквівалентним тридцять другому раунду алгоритму SERPENT, за винятком того, що він містить лінійну трансформацію і того, що тут використовуються 24-й та 25-й підключі (32-й і 33-й підключі в алгоритмі SERPENT). Таким чином, рівнянням останнього раунду є:

$$R23\,(X) = L\,(S23\,(X \oplus K23)) \oplus K24.$$

Примітив Serpent24 використовує тільки 25 128-бітних підключів, які є першими 25 підключами алгоритму SERPENT. Serpent24 в Sosemanuk використовується на етапі ініціалізації, тільки в режимі зашифрування. Розшифрування не використовується.

Регістр зсуву з лінійним зворотним зв'язком (РЗЛЗЗ). Для великої частини потокових шифрів внутрішній стан зберігається та оновлюється в РЗЛЗЗ. В шифрі Sosemanuk застосовується РЗЛЗЗ, який містить 10 елементів кінцевого поля $GF(2^{32})$.

Базове кінцеве поле. Елементи поля $GF(2^{32})$ представлені так само, як і в специфікації шифру SNOW 2.0. Наведемо коротко тут ці позначення.

Нехай $GF(2)$ позначає кінцеве поле з двома елементами. Нехай β є коренем примітивного полінома:

$$Q(X) = X^8 + X^7 + X^5 + X^3 + 1$$

в кільці поліномів $GF(2)[X]$. Визначимо поле $GF(2^8)$ як кільце $GF(2)[X] / Q(X)$. Кожен елемент в полі $GF(2^8)$ можемо визначити, використовуючи базис (β^7, β^6, ... β, 1). Оскільки обраний поліном примітивний, тоді β є мультиплікативним породжувальним елементом поля $GF(2^8)$: кожен ненульовий елемент з $GF(2^8)$ дорівнює β^k для деякого невід'ємного цілого k ($0 \le k \le 254$). Будь-який елемент з $GF(2^8)$ ідентифікується з 8-бітовим числом наступною бієктивною функцією:

$$\phi : GF(2^8) \to \{0,1,...,255\},\ x = \sum_{i=0}^{7} x_i \beta^i \mapsto \sum_{i=0}^{7} x_i 2^i\ ,$$

де кожен x_i є або 0 або 1. Наприклад, β^{23} представляється цілим числом $\phi\,(\beta^{23}) = 0xE1$ (в шістнадцятковій системі). Отже, додавання двох елементів в $GF(2^8)$ відповідає бітовій операції XOR між відповідними цілочисельними уявленнями. Множення на β рівнозначно лівому зсуву на один біт цілочисельного уявлення. Він виконується за операцією XOR з фіксованою маскою, якщо найстарший біт, який був видалений за допомогою зсуву, дорівнює 1.

Нехай α є коренем примітивного полінома:

$$P(X) = X^4 + \beta^{23} X^3 + \beta^{245} X^2 + \beta^{48} X + \beta^{239}$$

з $GF(2^8)[X]$. Поле $GF(2^{32})$ далі визначається як кільце $GF(2^8)[X]\ /\ P(X)$, тобто, його елементи представляються в базисі (α^3, α^2, α, 1). Будь-який елемент з $GF(2^{32})$ ідентифікується з 32-бітовим цілим числом такою бієктивною функцією:

$$\psi : GF(2^{32}) \to \{0,1,...,2^{32}-1\},$$

$$x = \sum_{i=0}^{3} y_i \alpha^i \mapsto \sum_{i=0}^{3} \phi(y_i) 2^{8i}.$$

Таким чином, додавання двох елементів в $GF(2^{32})$ відповідає побітовій операції XOR між їх цілочисельними уявленнями. Надалі ця операція буде позначена $\oplus$. Sosemanuk також використовує операції множення і ділення елементів в $GF(2^{32})$ на α. Множення $z \in GF(2^{32})$ на α відповідає лівому зсуву на 8 біт $\psi(z)$, яка слідує за операцією XOR з 32-бітною маскою, яка залежить тільки від найбільш значущих байт $\psi(z)$.

Визначення РЗЛЗЗ. РЗЛЗЗ працює з елементами $GF(2^{32})$. Початковий стан в момент часу $t = 0$ відповідає десяти 32-бітовим значенням $s_1 - s_{10}$. На кожному етапі, нове значення обчислюється за допомогою такої рекурсії:

$$s_{t+10} = s_{t+9} \oplus \alpha^{-1} s_{t+3} \oplus \alpha_{st}, \ \forall\, t \geq 1$$

і регістр зсувається (див. рис. 3 для ілюстрації роботи РЗЛЗЗ).

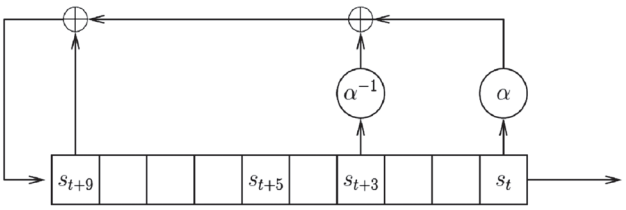


Рис. 3

Для РЗЛЗЗ, схема якого наведена на рис. 3, застосовується такий поліномом зворотного зв'язку:

$$\varpi(X) = \alpha X^{10} + \alpha^{-1} X^7 + X + 1 \in GF(2^{32})[X].$$

Оскільки РЗЛЗЗ є несингулярним, і оскільки ϖ є простим поліномом, то послідовність 32-бітових слів (s_t) $t \geq 1$ є періодичною і має максимальний період, що дорівнює 2^{320}-1.

Finite State Machine (кінцевий автомат). Кінцевий автомат (FSM) є компонентом з 64-ма бітами пам'яті, який відповідає двом 32-бітовим регістрам: $R1$ і $R2$. На кожному етапі FSM приймає на вхід кілька слів зі стану РЗЛЗЗ; він оновлює біти пам'яті і виробляє 32-біта на вихід. FSM оперує над станами РЗЛЗЗ в момент часу $t \geq 1$, як показано нижче:

$$\text{FSM}t\text{: } (R1_{t-1}, R2_{t-1}, s_{t+1}, s_{t+8}, s_{t+9})\ 1 \rightarrow (R1_t, R2_t, f_t),$$

де

$$R1_t = (R2_{t-1} + \text{mux}\,(\text{lsb}\,(R1_{t-1}),$$
$$s_{t+1}, s_{t+1} \oplus s_{t+8})) \bmod 2^{32},$$
$$R2_t = \text{Trans}\,(R1_{t-1}),$$
$$f_t = (s_{t+9} + R1_t \bmod 2^{32}) \oplus R2_t,$$

де $\text{lsb}(x)$ – найменший значущий біт x, $\text{mux}\,(c, x, y)$ дорівнює x, якщо $c = 0$, або y, якщо $c = 1$.

Внутрішня перехідна функція *Trans* на $GF(2^{32})$ визначається

$$Trans\,(z) = (M \times z \bmod 2^{32}) <<< 7,$$

де M є постійним значенням 0x54655307 і $<<<$ означає бітовий зсув 32-бітного значення (в даному випадку, на 7 біт).

Вихідне перетворення. Виходи FSM згруповані по чотири і Serpent1 застосовується до кожної групи. Далі результат комбінується операцією XOR з відповідними значеннями стану РЗЛЗЗ, для отримання на виході значень z_t:

$$(z_{t+3}, z_{t+2}, z_{t+1}, zt) =$$
$$= \text{Serpent1}\,(f_{t+3}, f_{t+2}, f_{t+1}, f_t) \oplus (s_{t+3}, s_{t+2}, s_{t+1}, s_t).$$

Чотири послідовних раунди Sosemanuk схематично зображені на рис. 4.

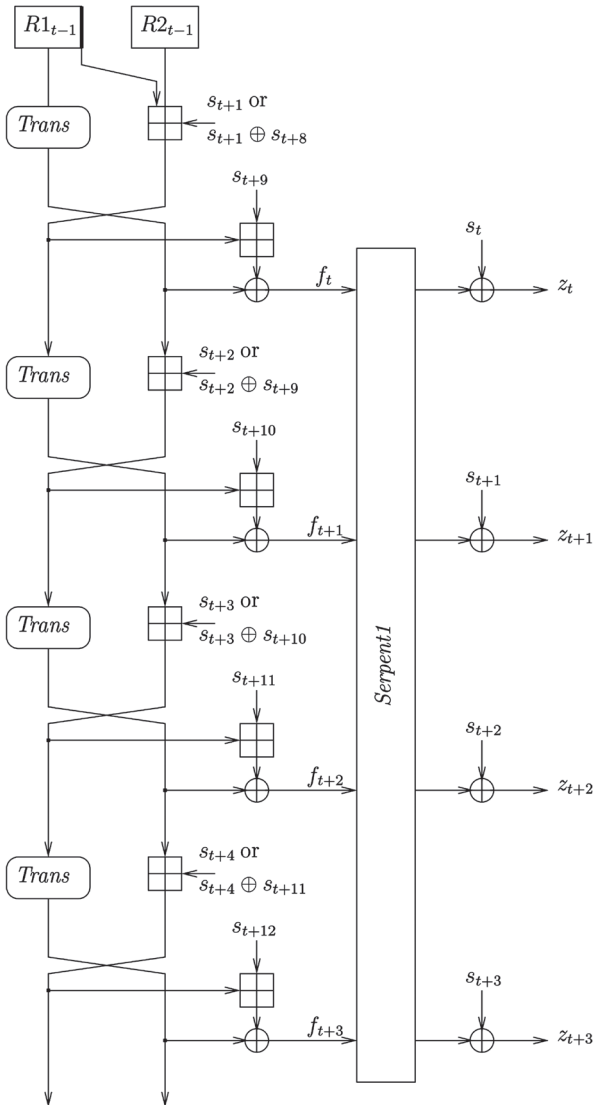


Рис. 4

Робочий процес Sosemanuk. Шифр Sosemanuk комбінує в собі FSM і РЗЛЗЗ для отримання вихідних значень z_t. Момент часу $t = 0$ позначає внутрішній стан після ініціалізації; перше вихідне значення – z_1. На рис. 5 наведено графічне позначення шифру Sosemanuk.

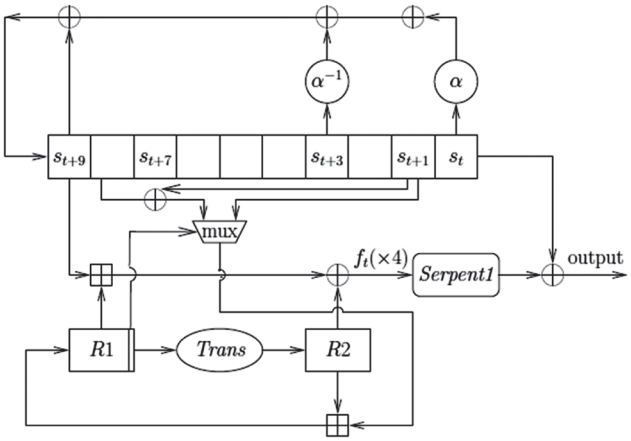


Рис. 5

У момент часу $t \geq 1$ виконуються такі операції:

– оновлюється FSM: $R1_t$, $R2_t$ і проміжне значення f_t обчислюється з $R1t-1$, $R2_{t-1}$, s_{t+1}, s_{t+8} та s_{t+9};

– оновлюється РЗЛЗЗ: s_{t+10} обчислюється виходячи з s_t, s_{t+3} та s_{t+9}. Значення s_t відправляється у внутрішній буфер і РЗЛЗЗ зсувається.

Раз на чотири кроки чотири вихідних значення z_t, z_{t+1}, z_{t+2}, z_{t+3} виходять з накопичених значень $f_t, f_{t+1}, f_{t+2}, f_{t+3}$ та s_t, s_{t+1}, s_{t+2}, s_{t+3}.

Таким чином, алгоритм Sosemanuk виробляє 32-бітові значення. Рекомендується зашифровувати їх у групи по чотири байти, використовуючи порядок байтів від молодшого до старшого, оскільки останній спосіб має більшу швидкість на більш широко використовуваному програмному забезпеченні – платформах високого класу (x86-сумісних). Крім того, цей метод використовує алгоритм SERPENT.

Отже, перші чотири ітерації Sosemanuk є такими:

– початковий стан РЗЛЗЗ містить значення $s_1 - s_{10}$; s_0 не містить ніякого значення. Початковий стан FSM містить $R1_0$ і $R2_0$;

– протягом першого етапу $R1_1$, $R2_1$ і f_1 обчислюються виходячи з $R1_0$, $R2_0$, s_2, s_9 і s_{10};

– з першого етапу отримують буферизовані проміжні значення s_1 і f_1;

– протягом першого етапу зворотне слово s_{11} обчислюється з s_{10}, s_4 і s_1, потім оновлюється внутрішній стан РЗЛЗЗ, що призводить до нового стану, який складається з $s_2 - s_{11}$;

– першими чотирма вихідними значеннями є z_1, z_2, z_3 і z_4, і обчислюються, використовуючи Serpent1 над (f_4, f_3, f_2, f_1), де вихід комбінується операціями XOR з (s_4, s_3, s_2, s_1).

Ключ ініціалізації і ін'єкція IV. Процес ініціалізації алгоритму Sosemanuk розділений на два етапи:

– генерація ключів, яка обробляє секретний ключ, але є незалежною від IV;

– ін'єкція IV, яка використовує результат попередньої операції і IV. Це ініціює внутрішній стан потокового шифру.

Генерація ключів. Генерація ключів відповідає генерації ключів Serpent24, який виробляє 25 128-бітових підключів, як 100 32-бітних слова. Ці 25 128-бітних підключа ідентичні першим 25 128-бітовим підключам, які виробляє генерація ключів алгоритму SERPENT.

Алгоритм SERPENT приймає будь-яку довжину ключа від 1 до 256 біт, отже Sosemanuk може працювати точно з такими ж ключами. Однак, оскільки Sosemanuk спрямований на 128-бітове шифрування, його довжини ключів мають бути принаймні 128 біт. Таким чином, 128 біт є стандартною довжиною ключа. Підтримується будь-яка довжина ключа від 128 біт до 256 біт. Але рівень безпеки як і раніше є 128-бітним. Іншими словами, використання більш довгого секретного ключа не гарантує забезпечення рівня захищеності, такого ж як і довжина ключа.

Ін'єкція IV. Вектор ініціалізації IV – це 128-бітове значення. Воно використовується як вхідні дані блокового шифру Serpent24, який ініційовано попередньою операцією (генерацією ключів). Алгоритм Serpent24 складається з 24 раундів, причому використовуються виходи 12, 18 і 24 раундів. Позначимо ці виходи таким чином:

– $(Y_3^{12}, Y_2^{12}, Y_1^{12}, Y_0^{12})$: вихід 12-го раунду;

– $(Y_3^{18}, Y_2^{18}, Y_1^{18}, Y_0^{18})$: вихід 18-го раунду;

– $(Y_3^{24}, Y_2^{24}, Y_1^{24}, Y_0^{24})$: вихід 24-го раунду.

Вихід кожного раунду складається з чотирьох 32-бітових слів відразу після лінійного перетворення, за винятком 24-го раунду, для якого вихід виходить тільки після складання з двадцять п'ятим підключем. Ці значення використовуються для ініціалізації внутрішнього стану Sosemanuk, з такими значеннями:

$$(s_7, s_8, s_9, s_{10}) = (Y_3^{12}, Y_2^{12}, Y_1^{12}, Y_0^{12});\ (s_5, s_6) = (Y_1^{18}, Y_3^{18});$$

$$(s_1, s_2, s_3, s_4) = (Y_3^{24}, Y_2^{24}, Y_1^{24}, Y_0^{24});\ R1_0 = Y_0^{18};$$

$$R2_0 = Y_2^{18}.$$

Перша покращена властивість процесу ініціалізації Sosemanuk полягає в тому, що він розділений на дві окремі стадії: генерація ключів, що не залежать від IV, і ін'єкція IV, яка генерує початковий стан генератора з вектора ініціалізації IV і виходу схеми генерації ключів. Потім настройка IV для фіксованого ключа є менш затратною, ніж повна установка ключа. Це покращує конструкцію, оскільки зміна вектора ініціалізації IV відбувається частіше, ніж зміна секретного ключа. Другою покращеною характеристикою є те, процес установки вектора ініціалізації IV одержано із застосуванням блокового шифру над IV. Якщо ми розглянемо функцію, яка відображає F_K n-бітний вектор ініціалізації IV у перші n бітів вихідного потоку (що генерується від ключа K і IV), тоді F_K має бути обчислювально невідрізнюваною від випадкової функції.

2. ДОСЛІДЖЕННЯ АПАРАТНО-ОРІЄНТОВАНИХ ПОТОКОВИХ ШИФРІВ

Переможцями у категорії апаратно-орієнтованих шифрів є такі алгоритми: Grain; MICKEY; Trivium. Розглянемо їх структуру, проаналізуємо основні перетворення.

2.1. Потоковий шифр Grain

Потоковий шифр Grain було запропоновано в [1]. Шифр націлений на апаратні середовища з дуже обмеженою кількістю вентилів (gate), споживаної потужності і пам'яті.

Алгоритм шифрування Grain заснований на двох регістрах зсуву і нелінійній функції виводу. Шифр має додаткову функцію, в результаті чого швидкість може бути збільшена за рахунок додаткового обладнання. Складність апаратної реалізації і швидкодії вигідно відрізняє цей алго-

ритм від інших апаратно орієнтованих потокових шифрів, наприклад, таких як E0 і A5 / 1.

Алгоритм Grain є синхронним потоковим шифром, у якому ключовий потік генерується незалежно від тексту. Конструкція заснована на двох регістрах зсуву, один з лінійним зворотним зв'язком (РЗЛЗЗ) і один регістр зсуву з нелінійним зворотним зв'язком (РЗНЛЗЗ). Застосування РЗЛЗЗ гарантує мінімальний період для гами, а також забезпечує збалансованість на виході. Застосування РЗНЛЗЗ разом з нелінійною функцією виходу вводить нелінійність до шифру. Вхід РЗНЛЗЗ маскується з виходу РЗЛЗЗ таким чином, що стан РЗНЛЗЗ стає збалансованим. Отже, в алгоритмі Grain використовується позначення РЗНЛЗЗ хоча це насправді фільтр. Те, що відомо про циклічні структури регістрів зсуву з нелінійним зворотним зв'язком не може бути негайно застосовано для аналізу шифру. Обидва регістра зсуву мають розмір 80 біт. Розмір ключа складає 80 біт і розмір вектора ініціалізації IV складає 64 біта. Структурна схема криптоперетворення схематично зображена на рис. 6.

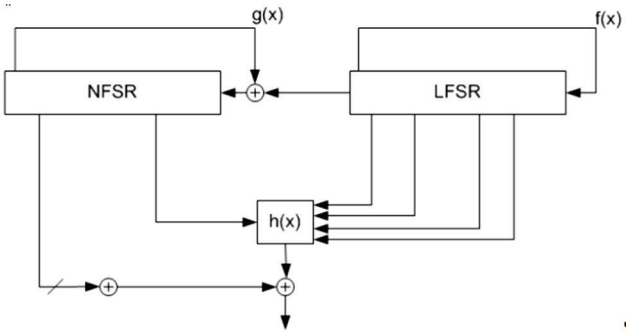


Рис. 6

Шифр складається з трьох основних блоків, а саме РЗЛЗЗ, в РЗНЛЗЗ і вихідної функції. Вміст РЗЛЗЗ позначається як

$$s_i, s_{i+1}, \ldots, s_{i+79},$$

вміст РЗНЛЗЗ позначається як

$$b_i, b_{i+1}, \ldots, b_{i+79}.$$

Поліномом зворотного зв'язку РЗЛЗЗ $f(x)$ є примітивний поліном степеня 80. Він має такий вигляд:

$$f(x) = 1 + x^{18} + x^{29} + x^{42} + x^{57} + x^{67} + x^{80}.$$

Для виключення будь-якої можливої двозначності в специфікації алгоритму шифрування визначено функцію оновлення в РЗЛЗЗ таким чином:

$$s_{i+80} = s_{i+62} + s_{i+51} + s_{i+38} + s_{i+23} + s_{i+13} + s_i.$$

Поліномом зворотного зв'язку РЗНЛЗЗ $g(x)$ визначається як

$$g(x) = 1 + x^{18} + x^{20} + x^{28} + x^{35} + x^{43} + x^{47} + x^{52} + x^{59} + x^{66} + x^{71} + x^{80} + x^{17}x^{20} + x^{43}x^{47} + x^{65}x^{71} + x^{20}x^{28}x^{35} + x^{47}x^{52}x^{59} + x^{17}x^{35}x^{52}x^{71} + x^{20}x^{28}x^{43}x^{47} + x^{17}x^{20}x^{59}x^{65} + x^{17}x^{20}x^{28}x^{35}x^{43} + x^{47}x^{52}x^{59}x^{65}x^{71} + x^{28}x^{35}x^{43}x^{47}x^{52}x^{59}.$$

Функція оновлення РЗНЛЗЗ має вигляд:

$$b_{i+80} = s_i + b_{i+62} + b_{i+60} + b_{i+52} + b_{i+45} + b_{i+37} + b_{i+33} + b_{i+28} + b_{i+21} + b_{i+14} + b_{i+9} + b_i + b_{i+63}b_{i+60} + b_{i+37}b_{i+33} + b_{i+15}b_{i+9} + b_{i+60}b_{i+52}b_{i+45} + b_{i+33}b_{i+28}b_{i+21} + b_{i+63}b_{i+45}b_{i+28}b_{i+9} + b_{i+60}b_{i+52}b_{i+37}b_{i+33} + b_{i+63}b_{i+60}b_{i+21}b_{i+15} + b_{i+63}b_{i+60}b_{i+52}b_{i+45}b_{i+37} + b_{i+33}b_{i+28}b_{i+21}b_{i+15}b_{i+9} + b_{i+52}b_{i+45}b_{i+37}b_{i+33}b_{i+28}b_{i+21}.$$

Слід звернути увагу на те, що біт s_i маскується з входом функції поновлення.

Вміст двох зсувних регістрів РЗЛЗЗ та РЗНЛЗЗ представляє стан шифру. П'ять змінних з цього стану взяті як вхід булевої функції, $h(x)$. Ця функція є нелінійним фільтром, вона обирається так, щоб бути збалансованою, забезпечувати кореляційний імунітет першого порядку і мати алгебраїчну степінь 3. Максимально можливою нелінійністю такої функції може бути 12. Вхід приймається як з РЗЛЗЗ, так і з РЗНЛЗЗ. Булева функція визначається як

$$h(x) = x_1 + x_4 + x_0x_3 + x_2x_3 + x_3x_4 + x_0x_1x_2 + x_0x_2x_3 + x_0x_2x_4 + x_1x_2x_4 + x_2x_3x_4,$$

де змінні x_0, x_1, x_2, x_3 і x_4 відповідають позиціям s_{i+3}, s_{i+25}, s_{i+46}, s_{i+64} і b_{i+63}, відповідно.

Вихідна функція має вигляд:

$$z_i = \sum_{k \in A} b_i + k + h(s_{i+3}, s_{i+25}, s_{i+46}, s_{i+64}, b_{i+63}),$$

де $A = \{1, 2, 4, 10, 31, 43, 56\}$.

Ініціалізація ключа. Перед генерацією ключового потоку шифр має бути ініційований ключем і вектором ініціалізації. Нехай біти ключа k будуть позначені як k_i, причому $0 \le i \le 79$, а біти вектора ініціалізації будуть позначені як IV_i, $0 \le i \le 63$. Ініціалізація ключа робиться таким чином.

По-перше, РЗНЛЗЗ завантажується ключовими бітами:

$$b_i = k_i,\ 0 \le i \le 79,$$

а потім перші 64 біти РЗЛЗЗ завантажуються бітами вектора ініціалізації:

$$s_i = IV_i,\ 0 \le i \le 63.$$

Решта бітів РЗЛЗЗ заповнюються одиницями:

$$s_i = 1,\ 64 \le i \le 79.$$

Через це РЗЛЗЗ не може бути ініційованим нульовим станом (одними нульовими значеннями). Далі шифр функціонує 160 тактів без виводу ключового потоку. Замість цього функція виходу подається назад і виконується операція XOR з входом як для РЗЛЗЗ, так і для РЗНЛЗЗ. Схематично це наведено на рис. 7.

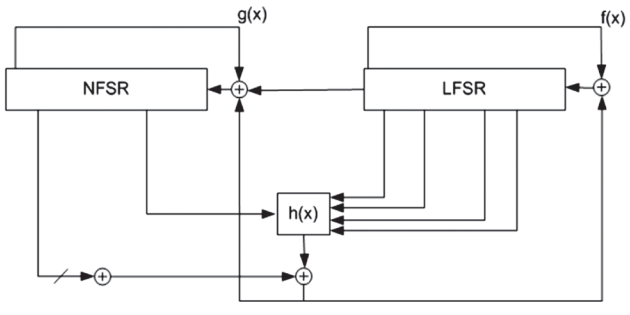


Рис. 7

Можна збільшити швидкість шифру за рахунок більш потужних апаратних засобів. Це може бути зроблено через збільшення кількості реалізацій функцій зворотного зв'язку $f(x)$ і $g(x)$. Внаслідок цього швидкість виведення також буде збільшена в декілька разів. Для того, щоб спростити цю реалізацію останні 15 біт регістрів зсуву s_i, $65 \le i \le 79$ і b_i, $65 \le i \le 79$ не використовуються у функціях зворотного зв'язку або на вході функції фільтра. Це дозволяє легко примножити швидкість до 16 разів, звісно, якщо доступна достатня кількість апаратних ресурсів. Приклад реалізації шифру, коли швидкість подвоюється, можна побачити на рис. 8.

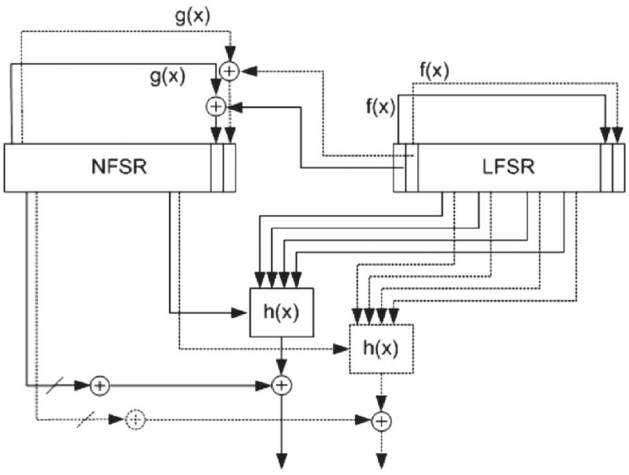


Рис. 8

Природно, що зсувні регістри також мають бути реалізовані так, щоб кожен біт зсувався t кроків замість одного, коли швидкість збільшено з коефіцієнтом t. Зі збільшенням швидкості на коефіцієнт 16, шифр виводить 16 біт за один зсув. Оскільки в ключовий ініціалізації шифру виконується 160 зсувів, можливості для збільшення швидкості обмежено показником $t \le 16$, причому t має бути дільником 160. Число зсувів, що використовуються при ініціалізації ключа дорівнюватиме $160 / t$. Оскільки функції фільтра і зворотного зв'язку досить малі, цілком можливо збільшити швидкість виведення таким чином [1].

2.2. Потоковий шифр MICKEY

В роботі [3] наведено вдосконалену версію 2.0 потокового шифру MICKEY (розшифровується як Mutual Irregular Clocking KEYstream generator – генератор ключового потоку із взаємно нерівномірним рухом), який призначений для апаратних платформ з обмеженими ресурсами.

Вхідні і вихідні параметри. Потоковий шифр MICKEY приймає два вхідних параметри:

– ключ K довжиною 80 біт, його біти позначаються k_0, k_1 ..., k_{79};

– вектор ініціалізації IV довжиною від 0 до 80 біт, його біти позначаються iv_0, ..., $iv_{LENGTH\,IV-1}$.

Послідовність бітів ключового потоку, що формує шифр MICKEY, позначається як $z_0, z_1, \ldots$. Шифртекст виробляється з незашифрованого (відкритого) тексту за допомогою побітової операції XOR з ключовим бітовим потоком, як і в більшості потокових шифрів.

Обмеження для застосування. Максимальна довжина ключового потоку, отриманого за допомогою однієї пари (K, IV) становить 2^{40} біт. Це є прийнятним для генерації подібних послідовностей з одним ключем K, але з різними значеннями IV. Не допускається використання двох векторів ініціалізації різної довжини з одним і тим же ключем K. Звичайно не є прийнятним повторне використання одного і того ж значення IV з фіксованим K.

Складові частини генератора послідовності ключів. Генератор складається з двох регістрів R і S. Довжина кожного регістра дорівнює 100 розрядів, кожен розряд містить 1 біт.

Біти цих регістрів позначаються r_0, r_1, ..., r_{99} та s_0, s_1, ..., s_{99} відповідно. Взагалі, в специфікації алгоритму вказано, що R – це «лінійний регістр», а S – «нелінійний регістр».

Зсув регістра R. Набір входів зворотного зв'язку регістра R:

$RTAPS$ = {0, 1, 3, 4, 5, 6, 9, 12, 13, 16, 19, 20, 21, 22, 25, 28, 37, 38, 41, 42, 45, 46, 50, 52, 54, 56, 58, 60, 61, 63, 64, 65, 66, 67, 71, 72, 79, 80, 81, 82, 87, 88, 89, 90, 91, 92, 94, 95, 96, 97}

Операція зсуву

CLOCK_R (R, $INPUT_BIT_R$, $CONTROL_BIT_R$)

визначається такою послідовністю дій:

– нехай r_0, r_1, ..., r_{99} – стан регістра R до зсуву, а r'0, r'1, ..., r'99 – після зсуву;

– FEEDBACK_BIT = r99 ⊕ INPUT_BIT_R;

– для всіх $1 \le i \le 99$, r'i = ri-1; r'0 = 0;

– для всіх $0 \le i \le 99$, якщо $i \in$ RTAPS, r'i = r'i ⊕⊕ FEEDBACK_BIT;

– якщо CONTROL_BIT_R = 1, то

– для всіх $0 \le i \le 99$, r'i = r'i ⊕ ri.

Зсув регістра S. Визначимо чотири послідовності

COMP01 ... COMP098, COMP11 ... COMP198,
FB00 ... FB099, FB10 ... FB199

як це наведено у табл. 6.

Функція зсуву регістра

CLOCK_S (S, $INPUT_BIT_S$, $CONTROL_BIT_S$)

визначається такою послідовністю дій:

– нехай s_0, s_1, ..., s_{99} – стан регістра S до зсуву, а s'_0, s'_1, ..., s'_{99} – після зсуву. Для простоти позначатимемо проміжний стан регістра як $\hat{s}_0$, $\hat{s}_1$, ..., $\hat{s}_{99}$;

– $FEEDBACK_BIT = s99 \oplus INPUT_BIT_S$;

– для всіх $1 \le i \le 98$, $\hat{s}_i = s_{i-1} \oplus ((s_i \oplus COMP0_i) \cdot (s_{i+1} \oplus COMP1_i))$; $\hat{s}_0 = 0$; $\hat{s}_{99} = s_{98}$;

– якщо $CONTROL_BIT_S = 0$, тоді для всіх $0 \le i \le 99$, $s'_i = \hat{s}_i \oplus (FB0_i \cdot FEEDBACK_BIT)$;

– якщо $CONTROL_BIT_S = 1$, то для $0 \le i \le 99$, $s'_i = \hat{s}_i \oplus (FB1_i \cdot FEEDBACK_BIT)$.

Управління рухом всього генератора визначено функцією

Таблиця 6

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$COMP0_i$		0	0	0	1	1	0	0	0	1	0	1	1	1	1	0	1	0	0	1	0	1	0	1	0
$COMP1_i$		1	0	1	1	0	0	1	0	1	1	1	1	0	0	1	0	1	0	0	0	1	1	0	1
$FB0_i$	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	0	0	1	0	1	1	1	1	1	1
$FB1_i$	1	1	1	0	1	1	1	0	0	0	0	1	1	1	0	1	0	0	1	1	0	0	0	1	0
i	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49
$COMP0_i$	1	0	1	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	1	0	1	0	1	0	1
$COMP1_i$	0	1	1	1	0	1	1	1	1	0	0	0	1	1	0	1	0	1	1	1	0	0	0	0	1
$FB0_i$	1	1	1	1	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	1	0	0	1	0	1
$FB1_i$	0	1	1	0	0	1	0	1	1	0	0	0	1	1	0	0	0	0	0	1	1	0	1	1	0
i	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74
$COMP0_i$	0	0	0	0	1	0	1	0	0	1	1	1	1	0	0	1	0	1	0	1	1	1	1	1	1
$COMP1_i$	0	0	0	1	0	1	1	1	0	0	0	1	1	1	1	1	1	0	1	0	1	1	1	0	1
$FB0_i$	0	1	0	0	1	0	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0
$FB1_i$	0	0	1	0	0	0	1	0	0	1	0	0	1	0	1	1	0	1	0	1	0	0	1	0	1
i	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99
$COMP0_i$	1	1	1	0	1	0	1	1	1	1	1	1	0	1	0	1	0	0	0	0	0	0	1	1	
$COMP1_i$	1	1	1	0	0	0	1	0	0	0	0	1	1	1	0	0	0	1	0	0	1	1	0	0	
$FB0_i$	1	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	1	1	0	0	1	1	0	0	0
$FB1_i$	0	0	0	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	1	0	0	0	0	1

CLOCK_KG (*R*, *S*, *MIXING*, *INPUT_BIT*) таким чином:

– $CONTROL_BIT_R = s_{34} \oplus r_{67}$;

– $CONTROL_BIT_S = s_{67} \oplus r_{33}$;

– якщо *MIXING* = *TRUE*, тоді $INPUT_BIT_R = INPUT_BIT \oplus s_{50}$, а якщо *MIXING* = *FALSE*, тоді $INPUT_BIT_R = INPUT_BIT$;

– $INPUT_BIT_S = INPUT_BIT$;

– *CLOCK_R*(*R*, *INPUT_BIT_R*, *CONTROL_BIT_R*);

– *CLOCK_S*(*S*, *INPUT_BIT_S*, *CONTROL_BIT_S*).

Завантаження ключа та ініціалізація. Ініціалізація регістрів з вхідних параметрів відбувається таким чином:

– заповнення регістрів *R* і *S* нулями;

– (завантаження в *IV*). Для всіх $0 \le i \le IV_{LENGTH} - 1$,

o CLOCK_KG (*R*, *S*, *MIXING* = *TRUE*, $INPUT_BIT = iv_i$);

– (завантаження в *K*). Для всіх $0 \le i \le 79$,

o CLOCK_KG (*R*, *S*, *MIXING* = *TRUE*, $INPUT_BIT = k_i$);

– (перед зсувом). Для всіх $0 \le i \le 99$,

o CLOCK_KG (*R*, *S*, *MIXING* =*TRUE*, $INPUT_BIT = 0$).

Генерація ключового потоку. Після завантаження та ініціалізації можна генерувати ключовий потік $z_0, \ldots, z_{L-1}$ таким чином:

– Для всіх $0 \le i \le L - 1$,

$$z_i = r_0 \oplus s_0;$$

CLOCK_KG (*R*, *S*, *MIXING* = *FALSE*, $INPUT_BIT = 0$).

Шифр MICKEY використовує дуже просту функцію виходу ($r_0 \oplus s_0$) для обчислення бітів ключового потоку від станів регістру. Алгоритм шифрування MICKEY має просту апаратну реалізацію і при цьому забезпечуючи високий рівень безпеки. У ньому використовується нерегулярний рух регістрів зсуву, а також нові методи, що забезпечують досить великий період ПВП і стійкість до певних криптоаналітичних атак.

2.3. Потоковий шифр Trivium

Алгоритм Trivium – це апаратно-орієнтований потоковий шифр [7]. Він був сконструйований як приклад спрощення потокового шифру без зменшення рівня безпеки, швидкості і гнучкості. У [7] наведено повний опис потокового шифру Trivium, досліджено особливості реалізації шифру та безпеки алгоритму.

За специфікацією [7] алгоритм Trivium – це потоковий шифр, призначений для генерації 2^{64} біт ключового потоку з 80 біт секретного ключа і 80 біт вектора ініціалізації (табл. 7).

Як і у більшості інших потокових шифрів, процес шифрування складається з двох фаз: спочатку внутрішній стан шифру зініціюється за допомогою ключа та вектора ініціалізації, далі стан кілька разів оновлюється і використовується для генерації бітів ключового потоку. Розглянемо спочатку другу фазу.

Таблиця 7

Параметри Trivium	
Розмір ключа	80 біт
Розмір вектора ініціалізації	80 біт
Внутрішній стан	288 біт

Генерація ключового потоку. Схема шифрування містить 288 біт внутрішнього стану і позначається (s_1, ..., s_{288}). Процес генерації ключового потоку є ітеративним, застосовує значення 15-ти спеціальних станів бітів і використовує їх всі для оновлення трьох бітів стану та підрахунку одного біта ключового потоку z_i.

Далі біти стану, перевертаються і процес повторюється знову, доки не буде згенеровано $N \le 2^{64}$ бітів ключового потоку. Повний опис подається у вигляді простого псевдокоду:

for i=1 to N do

$t_1 \leftarrow S_{66} + S_{93}$

$t_2 \leftarrow S_{162} + S_{177}$

$t_3 \leftarrow S_{243} + S_{288}$

$z_i \leftarrow t_1 + t_2 + t_3$

$t_1 \leftarrow t_1 + S_{91}S_{92} + S_{171}$

$t_2 \leftarrow t_2 + S_{175}S_{176} + S_{264}$

$t_3 \leftarrow t_3 + S_{286}S_{287} + S_{69}$

$(S_1, S_2, ..., S_3) \leftarrow (t_3, S_1, ...S_{92})$

$(S_{94}, S_{95}, ..., S_{177}) \leftarrow (t1, S_{94}, ...S_{176})$

$(S_{178}, S_{279}, ..., S_{288}) \leftarrow (t_3, S_{178}, ...S_{287})$

end for

Зазначимо, що операції «+» і «·» означають додавання і множення в полі GF(2), тобто це, відповідно, операції XOR і AND. Графічне подання процесу генерації ключового потоку зображено на рис. 9.

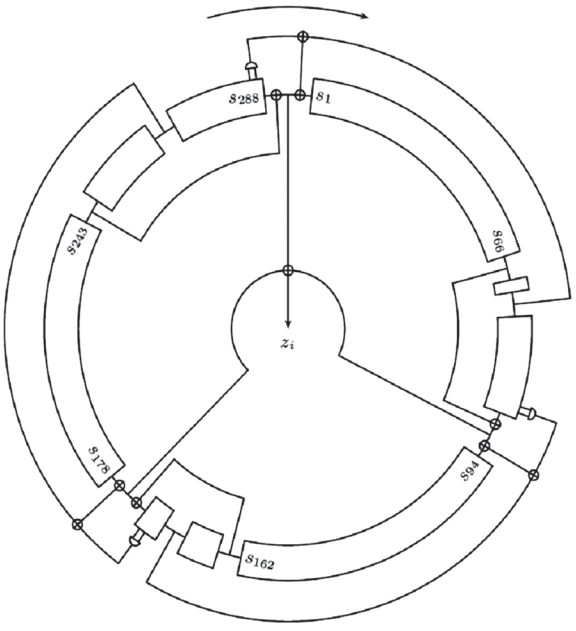


Рис. 9

Установка ключа і вектора ініціалізації. Алгоритм починається з установки 80-бітового ключа і 80-бітового вектора ініціалізації в 288-бітовий внутрішній стан, всі інші біти встановлюються в 0, крім s_{286}, s_{287}, s_{288}. Далі, значення стану перемішується протягом повних чотирьох циклів, таким же чином як було описано вище, однак без генерації потоку ключових біт. Все вищесказане описує псевдокод:

$(S_1, S_2, ..., S_3) \leftarrow (K_1, ..., K_{80}, 0, ...0)$

$(S_{94}, S_{95}, ..., S_{177}) \leftarrow (IV_1, ..., IV_{80}, 0, ...0)$

$(S_{178}, S_{279}, ..., S_{288}) \leftarrow (0, ..., 0, 1, 1, 1)$

for i =1 to 4*288 do

$t_1 \leftarrow S_{66} + S_{91}S_{92} + S_{93} + S_{171}$

$t_2 \leftarrow S_{162} + S_{175}S_{176} + S_{177} + S_{264}$

$t_3 \leftarrow S_{243} + S_{286}S_{287} + S_{288} + S_{69}$

$(S_1, S_2, ..., S_3) \leftarrow (t_3, S_1, ...S_{92})$

$(S_{94}, S_{95}, ..., S_{177}) \leftarrow (t1, S_{94}, ...S_{176})$

$(S_{178}, S_{279}, ..., S_{288}) \leftarrow (t_3, S_{178}, ...S_{287})$

Таким чином, шифр Trivium – це апаратно-орієнтований шифр, конструкція якого сфокусована на гнучкості. Його метою є компактність в середовищі з обмеженими вхідними параметрами, енергоефективність на платформах з малими джерелами енергії, а також швидкість в додатках, які вимагають високошвидкісного шифрування.

3. СТАТИСТИЧНІ ДОСЛІДЖЕННЯ ПОТОКОВИХ ШИФРІВ

Для дослідження статистичної безпеки розглянутих алгоритмів потокового шифрування було застосовано методику [12], згідно з якою оцінювалося математичне сподівання числа пройдених статистичних тестів NIST Special Publication 800-22 [13]. Отримані результат зведено у табл. 8.

В таблиці 8 наведено такі дані:

– «M096» та «M099» – оцінки математичного сподівання (вибіркові середні) числа пройдених статистичних тестів за критерієм $P_j \ge 0{,}96$ та за критерієм $P_j \ge 0{,}99$, відповідно;

– «D096» та «D099» («S096» та «S099») – оцінки дисперсій (середньоквадратичних відхилень) результатів тестування числа пройдених статистичних тестів за критеріями $P_j \ge 0{,}96$ та $P_j \ge 0{,}99$, відповідно;

– «P099» – значення довірчої ймовірності для числа пройдених статистичних тестів за критерієм $P_j \ge 0{,}99$ та при точності $\varepsilon = 2$;

– «P096» – значення довірчої ймовірності для числа пройдених статистичних тестів за критерієм $P_j \ge 0{,}96$ та при точності $\varepsilon = 1$.

В останній колонці «Min096» табл. 8 наведено мінімальні значення числа пройдених статистичних тестів за критерієм $P_j \ge 0{,}96$.

Під час проведення експериментальних досліджень розглядалися різні випадки: змінювався вектор ініціалізації без зміни секретного ключа (позначено у таблиці як «iv»), або змінювався секретний ключ без зміни вектора ініціалізації (позначено у таблиці як «key»). Отримані результати свідчать про високі характеристики досліджуваних криптографічних примітивів. У кожному з розглянутих випадків формовані ПВП мають

Таблиця 8

ПСШ	M099	D099	S099	P099	M096	D096	S096	P096	MIN
Grain_iv	131.63	33.24	5.77	1.00	186.78	0.51	0.71	1.00	182
Grain_key	133.05	24.56	4.96	1.00	186.91	0.39	0.62	1.00	182
hc-128_iv	132.49	37.51	6.12	1.00	186.52	2.30	1.52	1.00	176
hc-128_key	132.84	40.05	6.33	1.00	186.97	0.31	0.55	1.00	183
MICKEY_iv	133.17	19.82	4.45	1.00	186.78	0.47	0.68	1.00	181
MICKEY_key	131.62	35.89	5.99	1.00	186.80	0.28	0.53	1.00	183
rabbit_iv	133.01	30.32	5.51	1.00	186.65	1.03	1.01	1.00	180
rabbit_key	131.67	26.47	5.14	1.00	186.72	1.66	1.29	1.00	179
salsa20_iv	132.61	29.14	5.40	1.00	186.83	0.99	1.00	1.00	180
salsa20_key	133.64	26.58	5.16	1.00	186.97	0.31	0.55	1.00	183
sosemanuk_iv	132.42	21.44	4.63	1.00	186.77	0.53	0.73	1.00	181
sosemanuk_key	132.98	37.55	6.13	1.00	186.76	0.73	0.86	1.00	180
Trivium_iv	133.71	25.29	5.03	1.00	187.05	0.32	0.57	1.00	183
Trivium_key	132.27	31.22	5.59	1.00	186.83	0.86	0.93	1.00	181

вигляд псевдовипадкових і в статистичному сенсі не відрізняються від деякої гіпотетичної «випадкової послідовності». Експериментальні дані отримані з високою точністю і достовірністю, тобто можна стверджувати, що досліджувані алгоритми потокового симетричного криптоперетворення є статистично безпечними у будь-яких практичних застосуваннях.

ВИСНОВКИ

Проведений аналіз потокових криптоперетворень, які відібрані як переможці міжнародного проекту eSTREAM, показав, що за своєю специфікацією відповідні алгоритми поділяються на дві функціональні групи. Перша група алгоритмів це програмно-орієнтовані потокові шифри, до яких відносяться: HC-128; Rabbit; Salsa 20/12; SOSEMANUK. До другої групи апаратно-орієнтованих шифрів відносяться алгоритми Grain; MICKEY та Trivium.

Аналіз структури та основних застосованих функцій перетворення програмно-орієнтованих потокових шифрів (HC-128, Rabbit, Salsa 20/12, SOSEMANUK) показав, що у відповідних алгоритмах використовуються переважно прості та обчислювально ефективні операції побітового додавання за модулем 2, AND, циклічні зсуви, інші логічні функції, з можливістю застосовувати всі переваги розпаралелювання процесів формування ключового потоку. Стан генератора подається, як правило, або у вигляді слів, або табличним способом, що значно спрощує реалізацію обчислень при програмній реалізації. Виключенням можна вважати алгоритм SOSEMANUK, в якому для подання стану генератору використовується регістр зсуву довжини 10, але невелика довжина цього регістру не впливає суттєво на зменшення швидкодії навіть при програмній реалізації. Слід також відмітити, що потокові криптоперетворення застосовують елементи конструкції БСШ, зокрема, алгоритми SOSEMANUK та HC-128 застосовують нелінійні вузли заміни шифру AES.

Аналіз апаратно-орієнтованих шифрів, до яких відносяться алгоритми Grain, MICKEY та Trivium, показав, що в їхній структурі як правило використовуються регістри зсуву великої довжини. Це надає змогу при апаратній реалізації за один часовий такт роботи пристрою змінювати стан генератора ключового потоку, що дозволяє досягти надзвичайно високих показників швидкості формування ПВП. Для забезпечення криптографічної стійкості в структуру генератора додається також нелінійна функція, яка реалізується або шляхом нелінійного зворотного зв'язку у регістрах зсуву, або, наприклад, через застосування криптографічної булевої функції з високими показниками нелінійності. Складність криптоаналізу підвищується також через застосування нерівномірного руху регістрів, яке ускладнює вихідну ПВП.

Отримані результати свідчать про високі показники статистичної безпеки ПСШ, досліджувані алгоритми потокового криптоперетворення є статистично безпечними у будь-яких практичних застосуваннях.

Література

[1] The eSTREAM Project - eSTREAM Phase 3. Grain (Portfolio Profile 2). [Електронний ресурс]. – Режим доступу: http://www.ecrypt.eu.org/stream/grainpf.html.

[2] The eSTREAM Project - eSTREAM Phase 3. HC (Portfolio Profile 1). [Електронний ресурс]. – Режим доступу: http://www.ecrypt.eu.org/stream/hcpf.html.

[3] The eSTREAM Project - eSTREAM Phase 3. MICKEY (Portfolio Profile 2). [Електронний ресурс]. – Режим доступу: http://www.ecrypt.eu.org/stream/mickeypf.html.

[4] The eSTREAM Project - eSTREAM Phase 3. Rabbit (Portfolio Profile 1). [Електронний ресурс]. – Режим доступу: http://www.ecrypt.eu.org/stream/rabbitpf.html.

[5] The eSTREAM Project - eSTREAM Phase 3. Salsa20 (Portfolio Profile 1). [Електронний ресурс]. – Режим доступу: http://www.ecrypt.eu.org/stream/salsa20pf.html.

[6] The eSTREAM Project - eSTREAM Phase 3. SOSEMANUK (Portfolio Profile 1). [Електронний ресурс]. – Режим доступу: http://www.ecrypt.eu.org/stream/ sosemanukpf.html.

[7] The eSTREAM Project - eSTREAM Phase 3. Trivium (Portfolio Profile 2). [Електронний ресурс]. – Режим доступу: http://www.ecrypt.eu.org/stream/triviumpf.html.

[8] NESSIE Call for Cryptographic Primitives, Version 2.2 [Електронний ресурс]. – March, 2000. – Режим доступу: http://cryptonessie.org.

[9] NESSIE security report, deliverable D21, version 1.0 [Електронний ресурс]. – October 30, 2002ю – Режим доступу: http://www.cryptonessie.org.

[10] *Marcus Schafheutle.* A First Report on the Stream Cipher SNOW. [Електронний ресурс]. – Режим доступу: http://www.cryptonessie.org.

[11] *Ross J. Anderson.* Serpent: A Candidate Block Cipher for the Advanced Encryption Standard. University of Cambridge Computer Laboratory. Retrieved 2013-01-14. [Електронний ресурс]. – Режим доступу: http://www.cl.cam.ac.uk/~rja14/serpent.html.

[12] *Кузнецов А.А., Мордвинов Р.И., Колованова Е.П., Самойлова А.В.* Методика статистического тестирования криптографических алгоритмов // Спеціальні телекомунікаційні системи та захист інформації. – Київ – 2014. – №1(25). – С.54-61

[13] NIST Special Publication 800-22. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. [Электронный ресурс]. Режим доступа: http://csrc.nist.gov/publications/nistpubs/ 800-22-rev1a/SP800-22rev1a.pdf

Надійшла до редколегії 15.10.2015

Кузнецов Олександр Олександрович, доктор технічних наук, професор, професор кафедри БІСТ ХНУ ім. В.Н. Каразіна. Наукові інтереси: криптографія та стеганографія, методи обробки та передачі інформації.

Іваненко Дмитро Вікторович, кандидат технічних наук, доцент кафедри БІТ ХНУРЕ. Наукові інтереси: криптографія, криптоаналіз, атаки спеціального виду.

Мордвінов Руслан Ігорович, кандидат технічних наук, м.н.с. кафедри БІСТ ХНУ ім. В.Н. Каразіна. Наукові інтереси: криптографія, криптоаналіз, БСШ, генератори випадкових чисел.

УДК 004.056.55

Исследование поточных алгоритмов шифрования – победителей международного проекта eSTREAM / А.А. Кузнецов, Д.В. Иваненко, Р.И. Мордвинов // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 321–334.

Анализируются поточные алгоритмы шифрования, которые являются победителями международного проекта eSTREAM. Исследуется базовая структура алгоритмов-победителей, примененные функциональные связи, слои линейных и нелинейных преобразований и другое. Проводятся экспериментальные исследования статистической безопасности по методике тестирования NIST Special Publication 800-22. Установлено, что выходные последовательности исследуемых шифров соответствуют современным требованиям, рассмотреные криптопреобразования могут быть использованы при разработке разных вариантов построения национального стандарта потокового шифрования Украины.

Ключевые слова: криптографические преобразования, поточный шифр, статистические тесты.

Табл.: 8. Іл.: 9. Библиогр.: 13 назв.

UDK 004.056.55

Researching stream encryption algorithms – the winners of the international project eSTREAM / A.A. Kuznetsov, D.V. Ivanenko, R.I. Mordvinov // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 321–334.

Stream encryption algorithms are analyzed that determine the winners of the international project eSTREAM. The paper studies the basic structure of the winning algorithms, the applied functional ties, the layers of linear and nonlinear transformations etc. Pilot studies of statistical security using the method of testing NIST Special Publication 800-22 are conducted. It is found that the output sequence of investigated ciphers meet modern requirements, cryptotransformations are considered that can be used for the development of different variants of the Ukraine's national standard of streaming encryption.

Keywords: cryptographic transformation, stream cipher, static tests.

Tab.: 8. Fig.: 9. Ref.: 13 items.

УДК 621. 3.06

О МЕТОДИКЕ ОЦЕНКИ ЗАКОНОВ РАСПРЕДЕЛЕНИЯ ВЕРОЯТНОСТЕЙ МАКСИМУМОВ ПОЛНЫХ ДИФФЕРЕНЦИАЛОВ И СМЕЩЕНИЙ ЛИНЕЙНЫХ ОБОЛОЧЕК БЛОЧНЫХ СИММЕТРИЧНЫХ ШИФРОВ

К.Е. ЛИСИЦКИЙ

Представляется уточнённая (исправленная) версия известной методики определения законов распределения максимумов для множеств независимых переменных, имеющих одни и те же законы распределения, в частности для переменных с пуассоновским законом распределения вероятностей и нормальным законом распределения вероятностей.

Ключевые слова: закон распределения максимумов, независимые случайные переменные, максимумы полных дифференциалов, максимумы смещений линейных корпусов, переменные с пуассоновским законом распределения, переменные с нормальным законом распределения,

ВВЕДЕНИЕ

В этой статье речь будет идти об известной (опубликованной) методике определения законов распределения максимумов для множеств независимых переменных, имеющих одни и те же законы распределения, в частности для переменных с пуассоновским законом распределения вероятностей и нормальным законом распределения вероятностей.

Интерес к этой методике был вызван тем, что как оказалось множествами независимых переменных, имеющими одни и те же законы распределения, в частности переменными с пуассоновским законом распределения вероятностей и нормальным законом распределения вероятностей можно описать переходы таблиц полных дифференциалов и смещения таблиц линейных аппроксимаций блочных симметричных шифров, рассматриваемых как подстановочные преобразования.

Эта методика в работе [1] была успешно применена для выполнения вычислительных экспериментов по определению законов распределения максимумов переходов таблиц полных дифференциалов и максимумов смещений таблиц линейных аппроксимаций для уменьшенных моделей шифров и, в частности, шифра из нового белорусского стандарта. Были выполнены также оценки ожидаемых значений максимумов дифференциальных переходов и максимумов смещений для полномасштабного шифра Rijndael.

В соответствии с полученными для малых моделей шифров результатами сделан важный вывод о том, что все современные шифры имеют весьма малый диапазон изменения максимумов полных дифференциалов и максимумов смещений линейных корпусов. Это означает, что практически для оценки показателей доказуемой стойкости этих шифров можно пользоваться результатами оценок максимальных дифференциальных вероятностей и максимальных линейных вероятностей, вычисленных для произвольно взятого (одного) ключа зашифрования. Этот результат в соответствии с новой методологией оценки показателей стойкости блочных симметричных шифров к атакам дифференциального и линейного криптоанализа, развиваемой в работах И.В. Лисицкой [2, 3 и др.], может быть распространён и на полномасштабные версии этих шифров, что позволяет перейти к практическому вычислению показателей их стойкости к отмеченным атакам.

Математический аппарат, который успешно решает в общетеоретическом плане эту задачу, представлен в совместной работе учёных Joan Daemen, Vincent Rijmen [4].

Более глубокий анализ материалов этой работы, однако, показал, что по ходу изложения методики её авторами был допущен ряд неточностей (м.б. описок), вследствие чего излагаемые исходные положения не согласуются с результатами их применения к рассмотренным законам распределения вероятностей.

Следует отметить, что эти неточности не повлияли на правильное представление итоговых результатов. Соответственно правильными следует считать и результаты исследований, представленные в работе [1]. Тем не менее, мы посчитали важным навести порядок в описании этой важной на наш взгляд методики и представить её в исправленном виде.

В этой статье предлагается изложение этой методики вместе с исправлениями, позволяющими устранить несогласованность исходных положений и результатов их применения к конкретным законам распределения вероятностей.

1. ЗАКОН РАСПРЕДЕЛЕНИЯ МАКСИМУМОВ

Мы приведём здесь изложение материалов приложения из работы [4], в котором приводятся все необходимые теоретические сведения, составляющие сущность развиваемого подхода, тем более, что мы здесь имеем возможность исправить обнаруженные в представленном материале ошибки (описки).

Рассматривается случай, когда все значения множества независимых случайных переменных x имеют одни и те же распределения. Считается, что их плотности распределения убывают экспоненциально при больших значениях x. Обозначим, пишут авторы, число таких значений 2^Y и воспользуемся моделью интегрального распределения $D(X)$ в виде

$$D(X) = 1 - e^{-f(X)} \quad (1)$$

(на самом деле, как будет видно из дальнейшего, авторы цитируемой работы рассматривают интегральное распределение в виде $D(X) = 1 - f(X)$), с $f(X)$ – функцией, описывающей плотности распределения вероятностей независимых значений X.

Из порядковых статистик [5,6] известно, отмечают они, что это интегральное распределение максимального числа значений является произведением интегральных распределений этих значений. Поэтому мы имеем:

$$D_{\max}(X) = D(X)^{2^Y} = (1 - e^{-f(X)})^{2^Y} \approx \quad (2)$$

$$\approx e^{-2^Y e^{-f(X)}} = e^{-e^{\ln(2)Y - f(X)}}.$$

На самом деле здесь должен быть записан результат

$$D_{\max}(X) = D(X)^{2^Y} =$$

$$= (1 - f(X))^{2^Y} \approx e^{-2^Y \cdot f(X)} = e^{-e^{\ln(2)Y + \ln f(X)}}.$$

Мы можем аппроксимировать функцию $\ln(2)Y - f(X)$ (должно быть $\ln(2)Y + \ln f(X)$), говорят авторы работы [4], линейной функцией в окрестности точки, где функция близка к нулю. Пусть a будет решением уравнения

$$\ln(2)Y = f(X)$$

(должно быть уравнения $\ln(2)Y + \ln f(X) = 0$) и пусть b будет единицей, делённой на производную функции $f(x)$ (должно быть функции $\ln f(X)$) в точке a. Тогда справедливо выражение

$$D_{\max}(X) \approx e^{-e^{\frac{a-X}{b}}}. \quad (3)$$

Это распределение хорошо изучено в теории вероятностей, отмечается в [4], и известно как распределение экстремальных значений, Fisher-Tippett распределение или лог-Вейбулла распределение [5, 6]. Соответствующая плотность изображена на рис. 1, заимствованном из цитируемой работы [4]. Отмечается, что пик этой функции есть a, а ее ширина пропорциональна b. Это распределение имеет математическое ожидание $\mu(X) = a + b\gamma$ с $\gamma \approx 0{,}58$ и среднеквадратическое отклонение $\frac{\pi}{\sqrt{6}} b \approx 1{,}3b$. Авторы замечают, что справедливость выражения (3) зависит от качества линейной аппроксимации $f(x)$ вблизи точки $(a, 0)$.

Далее авторы работы [4] конкретизируют представление плотности вероятности (3) для переменных, подчиняющихся некоторым известным законам распределения. Нас будут интересовать два закона распределения: пуассоновский и нормальный. Первый соответствует закону распределения вероятностей переходов XOR таблицы случайной подстановки (шифра), второй – закону распределения вероятностей смещений случайной подстановки (шифра).

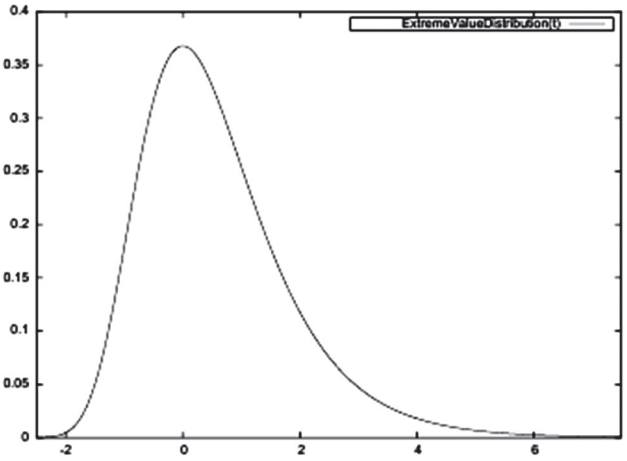


Рис. 1. Распределение экстремальных значений при $a = 0$, $b = 1$

1.1. МАКСИМУМЫ НАД X С ПУАССОНОВСКИМ РАСПРЕДЕЛЕНИЕМ

Дальнейший материал повторяет результаты, представленные в [1 и 4]. Если максимум берется по переменным с распределением Пуассона, мы, замечают авторы цитируемой работы, должен приниматься во внимание дискретный характер последнего. Однако, мы можем получить выражения для среднего значения и стандартного отклонения максимумов, если приблизим распределение Пуассона непрерывной функцией. Имеем:

$$\Phi(i;\lambda) = \sum_{x=0}^{i-1} \mathbf{Poisson}(x;\lambda) = 1 - \sum_{x \geq i} \mathbf{Poisson}(x;\lambda). \quad (4)$$

Для $i >> \lambda$, это выражение в соответствии с [5, 6] аппроксимируется так:

$$\Phi(i;\lambda) \approx 1 - \left(1 - \frac{\lambda}{i}\right) \cdot \mathbf{Poisson}(i;\lambda) \approx \quad (5)$$

$$\approx \mathbf{Poisson}(i,\lambda) = e^{-\lambda} \frac{\lambda^i}{i!}.$$

Далее, используя приближение Стирлинга для факториала [5,6], мы получим следующее выражение для функции $f(i)$:

$$f(i) = \frac{1}{2}\ln(2\pi) + \lambda + i\ln i - (1 + \ln\lambda)i + \frac{1}{2}\ln(i). \quad (6)$$

Конечно это выражение для логарифма функции $f(i)$, а не самой функции, как пишут авторы цитируемой работы, т.е. вместо (6)необходимо рассматривать

$$\ln f(i) = \frac{1}{2}\ln(2\pi) + \lambda + i\ln i - (1 + \ln\lambda)i + \frac{1}{2}\ln(i).$$

Если теперь абстрагироваться от факта, что i должно быть целым числом, пишут авторы ци-

тируемой работы, то мы можем вычислить параметр a путем решения уравнения:

$$\ln(2)y = \frac{1}{2}\ln(2\pi) + \lambda + i\ln i - (1+\ln\lambda)i + \frac{1}{2}\ln(i), \quad (7)$$

или, что эквивалентно:

$$i = \frac{\ln(2)y - \frac{1}{2}\ln(2\pi i) - \lambda}{\ln\left(\frac{i}{\lambda}\right) - 1}. \quad (8)$$

Последнее уравнение может быть решено итеративно. Производная функции $f(i)$ (должно быть функции $\ln f(i)$) определяется по формуле:

$$\ln\left(\frac{i}{\lambda}\right) + \frac{1}{2i}. \quad (9)$$

Определив a и используя условие $a >> \lambda$, мы имеем:

$$b = \frac{1}{\ln\left(\frac{a}{\lambda}\right)}. \quad (10)$$

Отсюда следует, заключают авторы цитируемой работы, что если a намного больше, чем λ, стандартное отклонение становится меньше 1.

Поскольку распределение максимума дискретное, то эта малая величина стандартного отклонения приводит к тому, что распределение сосредоточено в двух целочисленных значениях вблизи значения a.

1.2. МАКСИМУМЫ НАД X С НОРМАЛЬНЫМ РАСПРЕДЕЛЕНИЕМ

Здесь рассматривается частный случай, для переменной x со стандартным нормальным распределением. Мы имеем (цитируются результаты из работы [4]):

$$D(x) \approx \int_{-\infty}^{\infty} Z(u)du. \quad (11)$$

При больших x, это интегральное распределение, становится близким к [13, 14]:

$$D(x) \approx 1 - \frac{1}{x}Z(x) \approx 1 - \frac{1}{x\sqrt{2\pi}}e^{-\frac{x^2}{2}}. \quad (12)$$

Из этого результата мы можем получить следующее выражение для $f(x)$ (должно быть для $\ln f(x)$), и мы сразу запишем выражение так, как надо:

$$\ln f(x)):$$

$$= -\ln\left(\frac{1}{x}Z(x)\right) = \frac{1}{2}\left(\ln(2\pi) + x^2\right) + \ln(x) \quad (13)$$

Параметр, a_s (подстрочный индекс s для стандарта) является решением уравнения

$$a_s = \sqrt{2\ln(2)y - \ln(2\pi) - 2\ln(a_s)}, \quad (14)$$

которое может быть найдено итеративным путём, не обращая внимания на правый член в первой итерации. Производная функции $f(x)$ (должно быть $\ln f(x)$) определяется по формуле:

$$x + \frac{1}{x}, \quad (15)$$

и, следовательно,

$$b_s = \frac{a_s}{a_s^2 + 1} \approx \frac{1}{a_s}. \quad (16)$$

Грубо говоря, максимум имеет распределение со средним значением $1{,}17\sqrt{y}$ и стандартным отклонением $1{,}11/\sqrt{y}$. Мы, отмечают авторы работы [4], можем найти значения a и b для любого нормального закона распределения со средним $\mu(X)$ и стандартным отклонением σ, заменив x на $\frac{X - \mu(X)}{\sigma}$. Это дает:

$$\begin{aligned} a &= \sigma a_s + м(X), \\ b &= \sigma b_s. \end{aligned} \quad (17)$$

В общем, итоговые результаты работы [4] следует считать правильными.Они и были использованы при проведении экспериментов в работе [1].

Таким образом, изложенная методика оценки максимумов распределений, характерных для переходов таблиц полных дифференциалов и смещений таблиц линейных аппроксимаций шифров (случайных подстановок) с представленными правками вполне может быть применена для оценки максимумов полных дифференциалов и смещений блочных симметричных шифров.

Литература

[1] *Lisitskiy K.E.* On Maxima Distribution of Full Differentials and Linear Hulls of Block Symmetric Ciphers [Text] / K.E. Lisitskiy // I.J. Computer Network and Information Security, 2014, 1, 11-18 Published Online November 2013 in MECS (http://www.mecs-press.org/) DOI: 10.5815/ijcnis. 2014.01.02.

[2] *Лисицкая И.В.* Методология оценки стойкости блочных симметричных шифров / И.В. Лисицкая // Автоматизированные системы управления и приборы автоматики. – 2011. – № 163. – С. 123–133.

[3] Долгов Виктор Иванович. Методология оценки стойкости блочных симметричных шифров к атакам дифференциального и линейного криптоанализа: монография. / В.И. Долгов, И.В. Лисицкая. – Харьков. Издательство “Форт”, 2013. – 420 с.

[4] Joan Daemen, Vincent Rijmen Probability distributions of Correlation and Differentials in Block Ciphers. / Joan Daemen, Vincent Rijmen // April 13, 2006/ — P. 1–38.

[5] W. Feller An Introduction to Probability Theory and Its Applications, Vol.1. Wiley&Sons. 1968.

[6] Mathworld. http: //mathworld. wolfram.com.

Поступила в редколлегию 27.10.2015

Лисицкий Константин Евгеньевич, студент Харьковского национального университета им. В.Н. Каразина. Научные интересы: криптография, методы криптоанализа.

УДК 621. 3.06

Про методику оцінки законів розподілу ймовірностей максимумів повних диференціалів і зсувів лінійних оболонок блокових симетричних шифрів / К.Е. Лисицький // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 335–338.

Дається уточнена (виправлена) версія відомої методики визначення законів розподілу максимумів для множин незалежних змінних, що мають одні й ті самі закони розподілу, зокрема для змінних з пуассонівським законом розподілу ймовірностей і нормальним законом розподілу ймовірностей.

Ключові слова: закон розподілу максимумів, незалежні випадкові змінні, максимуми повних диференціалів, максимуми зміщень лінійних корпусів, змінні з пуассонівським законом розподілу, змінні з нормальним законом розподілу.

Іл.: 1. Бібліогр.: 6 найм.

UDC 621. 3.06

About the methodology for assessing the probability distribution laws of maxima of total differentials and offsets of block symmetric ciphers linear hulls / K.E. Lisitsky // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 335–338.

The paper provides a refined (corrected) version of the well-known methods for determining the maxima distribution laws for a set of independent variables with the same distribution laws, in particular for variables with Poisson probability distribution law and normal distribution law of probabilities.

Keywords: maxima distribution laws, independent random variables, total differentials maxima, maxima linear hulls offsets, variables with Poisson distribution law, variables with normal distribution law.

Fig. 1. Ref.: 6 items.

УДК 621.391

ОЦЕНКА ЭФФЕКТИВНОСТИ ПСЕВДОСЛУЧАЙНЫХ КОДОВ, СГЕНЕРИРОВАННЫХ С ПОМОЩЬЮ LFSR

Т.В. ЛАВРОВСКАЯ, О.И. ВОТЯКОВ, С.Г. РАССОМАХИН

Рассматривается возможность применения сдвиговых регистров с линейной обратной связью (LFSR) для генерации псевдослучайных помехоустойчивых кодов. Обоснована актуальность и проведена сравнительная оценка эффективности таких кодов путем сопоставления числовых характеристик спектров взаимных расстояний с аналогичными характеристиками равномерно случайных кодов. Сделано заключение о перспективности применения LFSR-кодов для снижения вычислительной сложности алгоритмов декодирования.

Ключевые слова: псевдослучайный код, LFSR, LFSR ПСК, равномерно случайный код, кодовое расстояние.

ВВЕДЕНИЕ

Постановка проблемы. В связи с глобализацией использования новых поколений сотовой связи и системы Интернет, для пользователей открываются новые горизонты применения данных сервисов, в частности все большую популярность набирают доверительные услуги, электронный документооборот, интернет-банкинг и другие. Это приводит к циркулированию в сети больших объемов информации, которая нуждается в надежной защите. Поэтому возникает проблема в разработке эффективных методов кодирования данных, которые будут обладать достаточным уровнем помехозащищенности. В связи с этим новый виток в развитии получили линейные блоковые коды, использующие в качестве канальных символов псевдослучайные числа (псевдослучайные коды – ПСК) [1].

Основной преградой для широкого использования таких кодов является то, что декодирование, как правило, основывается на реализации принципа максимального правдоподобия и возможно лишь с использованием переборных алгоритмов.

Вычислительная сложность таких алгоритмов возрастает экспоненциально с длиной блока кода и при практически требуемых значениях длины блока является неприемлемой. Это является прямым следствием того, что существующие методы получения последовательностей псевдослучайных чисел основаны на применении нелинейных операций, к которым можно отнести и вычисление по модулю. Поэтому, была предпринята попытка получения ПСК с помощью применения линейного сдвигового регистра с обратной связью (LFSR), а также оценка эффективности кодов. При этом использование LFSR дает возможность реализации линейных алгебраических методов декодирования, например, с помощью алгоритма Берлекэмпа - Мэсси или его модификаций [2].

Цель работы: разработка метода построения ПСК, который позволит реализовать декодирование, путем решения системы линейных уравнений, а также сравнение эффективности по характеристикам спектра взаимных расстояний (весов) LFSR ПСК с равномерно случайными кодами (РСК), полученными более сложными (нелинейными) алгоритмами.

1. АНАЛИТИЧЕСКАЯ МОДЕЛЬ ПСЕВДОСЛУЧАЙНОГО LFSR-КОДА

Линейные коды в контексте данной работы выступают как множество точек многомерного пространства, система координат которого образуется вектором R. В качестве начальной инициализации для элементов вектора выбираются числа из диапазона $R_i \in \left[0, 2^n - 1\right]$, где n – это степень примитивного многочлена $P(x)$, который определяет структуру LFSR генератора. Элементы регистра при его инициализации представляются в двоичной системе счисления с алфавитом «-1» и «+1». Эти числа являются разрядами двоичного слова источника, подлежащего кодированию для формирования канального кодового слова ПСК.

Реализация регистра сдвига, обозначаемого в дальнейшем R, основывается на применении LFSR с базовой конфигурацией Фибоначчи. Основным отличием от традиционных рассмотрений LFSR структур в задачах формирования сложных сигнальных конструкций, является замена нелинейной операции суммирования по модулю два простым алгебраическим (с учетом знака) сложением.

Пусть

$$P(x) = \sum_{i=0}^{n} c_i \cdot x^i, \; c_i \in [0,1] \qquad (1)$$

– полином структуры LFSR. Обозначим s – вектор, содержащий $m+1$ элементов из диапазона $[0,n]$, каждый из которых равен степени переменной x, входящей в (1) с ненулевым коэффициентом c_i. Величина m определяется числом резидентных обратных связей в структуре LFSR. Оправданным является название “вектор отводов”, используемое в дальнейшем для вектора s.

Используемая для построения LFSR ПСК схема сдвигового генератора представлена на рис. 1. В процессе генерации происходит потак-

товый сдвиг содержимого ячеек памяти регистра право на 1 позицию. При этом в отличие от традиционного алгоритма Фибоначчи каждая из ячеек памяти регистра сдвига способна хранить знаковые числа, абсолютная величина которых значительно превышает единицу. Кроме того, в отличие от обычных LFSR, в цепях обратных связей присутствует операция знакочередования, задаваемая множителями $(1)^j$, $j \in \overline{1,(n-1)}$. Сумматоры на рис. 1 являются простыми алгебраическими устройствами сложения целых знаковых чисел. Содержимое младшей ячейки памяти регистра формируется в соответствии с вектором отводов s. Текущее значение символа ПСК считывается на каждом такте из старшей (ячейки) регистра R после завершения очередного сдвига вправо. В младшую ячейку заносится результат вычисления R_0 в соответствии со следующей формулой:

$$R_0 = \sum_{j=1}^{m} (-1)^j R_{(s_j - 1)}, \qquad (2)$$

где s – определенный выше вектор отводов.

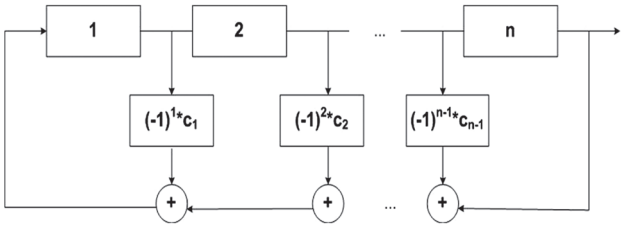


Рис. 1. Схема генератора ПСК с помощью LFSR

В результате инициализации вектора R двоичным числом i за конечное количество тактов N генерируется вектор, координатами которого являются псевдослучайные числа $Z_i = \{Z_{i,1}, Z_{i,2}, ... Z_{i,n}\}$. Перебор i от 0 до $2^n - 1$ дает возможность составления кортежей последовательностей Z_i, которые в последующем объединены в матрицу ZDV из транспонированных векторов Z_i:

$$ZDV = augment(Z_0, Z_1, ..., Z_{2^n-1})^T, \qquad (3)$$

где *augment* – функция объединения векторов Z_i в матрицу с размерностью $2^n \times N$.

Формирование кодовой книги осуществляется в два этапа.

На первом этапе производится выбор подматрицы $KB \in ZDV$, строки которой являются прототипами кодовых слов LFSR ПСК. Подматрица KB обладает размерностью $2^n \times n$, причем $\frac{n}{r} = V$ – скорость помехоустойчивого кода. Так как соотношение между n и r может быть произвольным, то допускаются значения скорости кода V как меньше, так и больше единицы, что является характерным для псевдослучайных кодов. В данной работе исследуются коды только для случая $V = 1$, т.е. $r = n$. Тогда функция извлечения подматрицы KB имеет вид:

$$KB = (submatrix(ZDV, 0, 2^n - 1, nc, nc + r - 1)), \qquad (4)$$

где *submatrix* – функция, которая формирует подматрицу из ZDV, состоящую из строк с номерами от 0 до $2^n - 1$ и столбцов с номерами от nc до $nc + r - 1 = nc + n - 1$ включительно. Здесь параметр nc обозначает начальный номер столбца ZDV для формируемой подматрицы KB. Для исследований использован параметр $nc = n$.

На втором этапе производится нормировка символов (чисел) канального кода в соответствии с задаваемым бюджетом энергии. Смысл данной операции заключается в том, чтобы средний квадрат любого канального символа подматрицы KB не превышал заданного максимального значения $q_{\max}$. Для упрощения расчетов в представленной модели значение $q_{\max}$ положено единичным. На выходе генератора последовательности канальных символов образуют псевдохаотический поток из знаковых натуральных чисел.

Для минимизации канальных затрат энергии (в предположении использования многоуровневой амплитудной модуляции)целесообразна операция центрирования канальных символов (чисел) в относительно нуля. Для этого необходимо вычисление математического ожидания для элементов выбранной на первом этапе подматрицы:

$$MX = \frac{1}{n - 2^n} \cdot \sum_{i=0}^{2^n-1} \sum_{j=0}^{n-1} (KB_i)_j. \qquad (5)$$

После выполнения (5) производится перерасчет прототипов кодовых слов и строится модифицированная матрица из центрированных относительно нуля псевдослучайных элементов $KB0$:

$$KB0 = KB - MX. \qquad (6)$$

Средняя мощность (дисперсия) канальных чисел может быть вычислена по формуле:

$$DX = \frac{1}{n \cdot 2^n} \sum_{i=0}^{2^n-1} \sum_{j=0}^{n-1} ((KB0_i)_j)^2. \qquad (7)$$

Финальной операцией второго этапа формирования кодовой книги является нормировка центрированной матрицы $KB0$ в соответствии с выделенным бюджетом энергии на передачу канального символа $q_{\max}$. При этом производится окончательное формирование кодовой книги LFSR ПСК $KB1$:

$$KB1 = \frac{KB0}{\sqrt{DX}} \cdot \sqrt{q_{\max}}. \qquad (8)$$

Очевидно, что корректность энергетической нормировки может быть подтверждена проверкой выполнения следующего равенства:

$$\frac{1}{n \cdot 2^n} \cdot \sum_{i=0}^{2^n-1} \sum_{j=0}^{n-1} ((KB1_i)_j)^2 = q_{\max}, \qquad (9)$$

которое является прямым следствием выражений (7) и (8) [3].

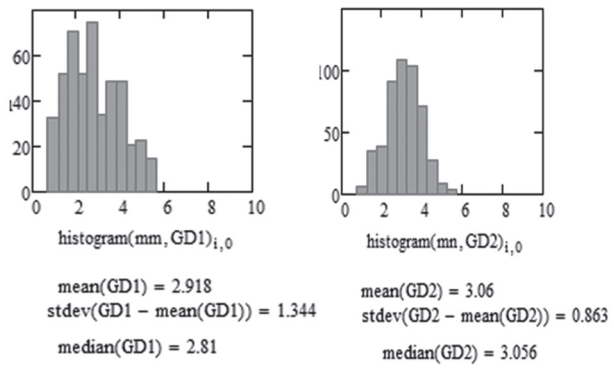
Для достижения требуемого уровня помехозащищенности необходимо использовать коды, которые обладают достаточно большим расстоянием между всеми парами кодовых слов. Поэтому

вычисление характеристик кодовой книги, количественно оценивающих взаимную удаленность кодовых слов (точек m-мерного пространства) является эффективным способом оценки характеристик помехоустойчивости LFSR ПСК. Для реализации этой цели в рассматриваемой модели используется функция оценки эффективности кодов по характеристикам спектра взаимных расстояний (весов) $GD1$:

$$GD1 = \underbrace{\left\{gd_0^1, gd_1^2, ..., gd_{2^n-2}^{2^n-1}\right\}}_{C_{2^n}^2 = \frac{2^n!}{(2^n-2)!\cdot 2!}\ -\ \text{элементов}}, \qquad (10)$$

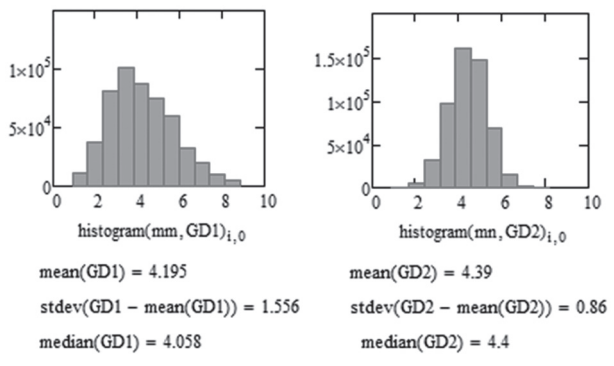
где $gd_i^j = \left|KB1_i - KB1_j\right|$ – взаимное расстояние между словами с номерами i и j в кодовой книге LFSR ПСК.

Результаты тестирования рассматриваемой модели LFSR ПСК представлены на рис. 2, 3 в виде гистограмм, которые отображают распределение взаимных расстояний кодовых слов $GD1$, полученных описанным выше способом при разных порождающих полиномах. Для осуществления возможности сравнительной оценки на этих же рисунках представлены гистограммы распределения весов для псевдослучайных кодов, символы которых генерируются программным датчиком с нелинейным алгоритмом формирования равномерно распределенных в заданном диапазоне чисел. Такие коды можно использовать в качестве эталона для сравнения, поскольку их свойства достаточно хорошо изучены и обеспечивают в определенных диапазонах отношений сигнал/шум монотонное убывание вероятности декодирования с ошибкой при увеличении длины блока [4]. Для обеспечения корректных условий сравнения спектров взаимных расстояний LFSR ПСК и РСК последние нормированы на одинаковую с (8) среднюю энергию канального символа.



а *б*

Рис. 2. Распределение кодовых слов ПСК LFSR (*а*) и РСК (*б*) при $P(x) = x^5 + x^2 + 1$



а *б*

Рис. 3. Распределение кодовых слов ПСК LFSR (*в*) и РСК (*г*) при $P(x) = x^{10} + x^3 + 1$

В качестве основных числовых параметров для сравнения использованы вычисления:

– математического ожидания взаимного евклидова расстояния между парами кодовых слов $mean(GD1)$ и $mean(GD2)$;

– среднеквадратического отклонения для гистограмм исследуемых распределений $stdev$;

– медианы распределения спектров $median(GD1)$ и $median(GD2)$.

Сопоставление полученных числовых характеристик для LFSR ПСК и РСК свидетельствует о их достаточной близости. Коды, получаемые с помощью линейного алгоритма LFSR, безусловно уступают по числовым параметрам спектров весов равномерно случайным кодам. Однако отсутствие возрастания этого проигрыша при увеличении степени порождающего многочлена LFSR, а также простота линейной алгебраической генерации, как предпосылка реализации вычислительно легких алгоритмов декодирования, дают возможность для констатации перспективности рассмотренных кодов.

ВЫВОДЫ

Построенная модель и проведенный вычислительный эксперимент позволяют сделать оптимистический прогноз о дальнейшем использовании ПСК сгенерированных на основе LFSR. Хотя по полученным характеристикам (математическое ожидание, дисперсия и среднеквадратическое отклонение) коды сгенерированные на основе LFSR, и проигрывают равномерно случайным кодам. Однако эти недостатки можно компенсировать, за счет использования порождающего неприводимого многочлена более высокой степени. Такие коды, в силу линейности алгоритма формирования символов кодовых слов являются легко декодируемыми с использованием линейных алгебраических методов. Это дает возможность увеличения длины блока, до практически требуемых значений, а также открывает перспективу для решения проблемы декодирования ПСК. В качестве направления дальнейших исследований целесообразно выбрать рассмотрение линейных алгебраических методов декодирования LFSR ПСК, например, с помощью алгоритма Бэрлекэмпа – Мэсси или его модификаций.

Литература

[1] *Григорьев, В.А.* Сети и системы радиодоступа [Текст] / В. А. Григорьев, О. И Лагутенко., Ю.А. Распаев. – М.: Эко-Трендз, 2005. – 384 с.

[2] *Скляр, Б.* Цифровая связь. Теоретические основы и практическое применение [Текст] / Б. Скляр //

Изд. 2-е» испр. : Пер. с англ. — М. : Издательский дом “Вильяме”, 2003. — С. 354–378.

[3] *Гихман, И.И.* Теория вероятностей и математическая статистика [Текст] / И.И. Гихман, А.В. Скороход, Н.И. Ядренко. – К.: Высшая шк., 1982. – 439 с.

[4] *Лавровская, Т.В.* Анализ применения правила простого округления для получения вычислительно реализуемых методов декодирования / С.Г. Рассомахин, Т.В. Лавровская // Системи обробки інформації. – 2015. – Вип. 5 (151). – С. 115–117.

Поступила в редколлегию 9.11.2015

Лавровская Тамила Валериевна, аспирант кафедры безопасности информационных систем и технологий ХНУ имени В. Н. Каразина. Научные интересы: информационная безопасность, теория передачи информации.

Вотяков Олег Иванович, кандидат технических наук, директор ЦКБ «Протон», г. Харьков. Научные интересы: информационная безопасность, системы радиомониторинга и радиоконтроля.

Рассомахин Сергей Геннадиевич, доктор технических наук, доцент, заведующий кафедрой БИСТ ХНУ им. В.Н. Каразина. Научные интересы: информационная безопасность, теория передачи информации.

УДК 621.391

Оцінка ефективності псевдовипадкових кодів згенерованих за допомогою LFSR / Т.В. Лавровська, О.І. Вотяков, С.Г. Рассомахін // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 339–342.

Розглядається можливість застосування зсувних регістрів з лінійним зворотним зв’язком (LFSR) для генерації псевдовипадкових завадостійких кодів. Обгрунтовано актуальність і проведено порівняльну оцінку ефективності таких кодів шляхом зіставлення числових характеристик спектрів взаємних відстаней з аналогічними характеристиками рівномірно випадкових кодів. Зроблено висновок про перспективність застосування LFSR-кодів для зниження обчислювальної складності алгоритмів декодування.

Ключові слова: псевдовипадковий код, LFSR, LFSR ПСК, рівномірно випадковий код, кодова відстань.

Рис.: 3. Бібліогр.: 4 найм.

UDC 621.391

Evaluating the effectiveness of pseudorandom codes generated by LFSR / T.V. Lavrovska, O.I. Votyakov, S.G. Rassomakhin // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 339–342.

The paper considers a possibility of linear shift feedback registers (LFSR) application for generation of pseudorandom error-correction codes. The actuality has been substantiated and comparative estimation of such codes effectiveness via numerical characteristics comparison of mutual distances spectra with the same characteristics of uniformly random codes has been carried out. A conclusion about the prospectiveness of LFSR-codes application for decoding algorithms computation complexity reduction has been made.

Keywords: pseudo random code, LFSR, LFSR PRC, evenly random code, code distance.

Fig.: 3. Ref.: 4 items.

УДК 004.056.55

АНАЛИЗ И СРАВНИТЕЛЬНЫЕ ИССЛЕДОВАНИЯ НЕЛИНЕЙНЫХ УЗЛОВ ЗАМЕНЫ СОВРЕМЕННЫХ БЛОЧНЫХ СИММЕТРИЧНЫХ ШИФРОВ

А.А. КУЗНЕЦОВ, И.Н. БЕЛОЗЕРЦЕВ, А.В. АНДРУШКЕВИЧ

Рассматриваются современные блочные симметричные шифры и применяемые в них нелинейные узлы замены (S-блоки). Анализируются показатели и критерии эффективности S-блоков: нелинейность, сбалансированность, корреляционный иммунитет, критерий распространения, автокорреляция и др. Проводятся сравнительные исследования эффективности нелинейных узлов замены современных блочных шифров.

Ключевые слова: блочный симметричный шифр, нелинейный узел замен, криптографическая функция.

ВВЕДЕНИЕ

Интенсивное развитие современных компьютерных систем и технологий, применение облачных вычислений, мобильных устройств удаленного подключения к информационно-коммуникационным сетям приводят к значительному усложнению процедур защиты обрабатываемой и передаваемой информации. Необходимым условием выполнения основных услуг безопасности является применение криптографических средств защиты информации [1].

Блочные симметричные шифры (БСШ) являются одним из наиболее распространенных и эффективных криптографических примитивов [1–5]. Помимо обеспечения услуги конфиденциальности они применяются как основной конструктивный элемент других примитивов (функций хеширования, генераторов псевдослучайных последовательностей и пр.). Важность разработки, исследования и обоснования условий применения современных БСШ подтверждается количеством и масштабностью международных криптографических конкурсов, проведенных в последние годы. Так, например, международные проекты AES, NESSIE, CRYPTREC и пр. были ориентированы на разработку БСШ, удовлетворяющих высоким требованиям криптографической стойкости и эффективности программной и аппаратной реализации. Результатом проведения этих и многих других исследовательских проектов являются принятые в последние годы международные и национальные стандарты криптографического преобразования [2–5].

Разработка, исследование и обоснование условий применения современных БСШ является чрезвычайно сложной и трудоёмкой задачей [1]. Стандартизированный криптоалгоритм должен обеспечивать высокий уровень стойкости, обладать требуемым быстродействием и эффективно функционировать на различных вычислительных платформах [1, 6]. При этом требуемый уровень стойкости БСШ обеспечивается эффективностью составляющих: схемы разворачивания секретного ключа (ключевого расписания), выбранной базовой структуры алгоритма, линейных и нелинейных преобразований, и пр. В данной работе акцентируется внимание на нелинейных узлах замен, критериях и показателях их эффективности. Приводятся результаты сравнительных исследований различных свойств S-блоков современных БСШ, стандартизированных на международном и национальном уровнях [2–5].

1. УЗЛЫ ЗАМЕН СОВРЕМЕННЫХ БСШ

Рассмотрим нелинейные узлы замен современных БСШ, проведем исследования свойств применяемых S-блоков по различным показателям и критериям эффективности [7–12].

БСШ «Калина» – национальный стандарт Украины ДСТУ 7624:2014 [5], разработанный ЧАО «ИИТ». Алгоритм поддерживает размер блока и длину ключа шифрования 128, 256 и 512 бит (длина ключа равна размеру блока или в два раза превышает его), обеспечивая нормальный, высокий и сверхвысокий уровень стойкости. Цикловое преобразование использует таблицы подстановки, приведенные на рис. 1–4 [5].

БСШ «Кузнечик» – алгоритм шифрования, стандартизированный в России как ГОСТ 34.12-2015 [3]. Длина входного блока – 128 бит, длина ключа – 256 бит. Процесс зашифрования основан на последовательном применении нескольких однотипных раундов, каждый из которых содержит сложение с раундовым ключом, нелинейное и линейное преобразование. Нелинейное преобразование использует фиксированный S-блок, представленный на рис. 5.

БСШ «BelT» – государственный стандарт симметричного шифрования и контроля целостности Республики Беларусь. Стандартизирован как СТБ 34.101.31-2011 [4] и введен в действие в 2011 году. Это блочный криптоалгоритм с 256-битным ключом и 8 циклами, оперирующий 128-битными словами. S-блок БСШ «BelT» приведен на рис. 6.

БСШ «AES» – криптоалгоритм, стандартизированный в США как FIPS-197. На международном уровне стандартизирован в ISO/IEC 18033-3 [2]. Обрабатывает 128-битные блоки данных, используя ключи шифрования длиной 128, 192 и 256 бит. В зависимости от длины ключа

выполняется 10, 12 или 14 раундов шифрования. Таблица подстановок БСШ «AES» приведена на рис. 7.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	A8	43	5F	06	6B	75	6C	59	71	DF	87	95	17	F0	D8	09
1	6D	F3	1D	CB	C9	4D	2C	AF	79	E0	97	FD	6F	4B	45	39
2	3E	DD	A3	4F	B4	B6	9A	0E	1F	BF	15	E1	49	D2	93	C6
3	92	72	9E	61	D1	63	FA	EE	F4	19	D5	AD	58	A4	BB	A1
4	DC	F2	83	37	42	E4	7A	32	9C	CC	AB	4A	8F	6E	04	27
5	2E	E7	E2	5A	96	16	23	2B	C2	65	66	0F	BC	A9	47	41
6	34	48	FC	B7	6A	88	A5	53	86	F9	5B	DB	38	7B	C3	1E
7	22	33	24	28	36	C7	B2	3B	8E	77	BA	F5	14	9F	08	55
8	9B	4C	FE	60	5C	DA	18	46	CD	7D	21	B0	3F	1B	89	FF
9	EB	84	69	3A	9D	D7	D3	70	67	40	B5	DE	5D	30	91	B1
a	78	11	01	E5	00	68	98	A0	C5	02	A6	74	2D	0B	A2	76
b	B3	BE	CE	BD	AE	E9	8A	31	1C	EC	F1	99	94	AA	F6	26
c	2F	EF	E8	8C	35	03	D4	7F	FB	05	C1	5E	90	20	3D	82
d	F7	EA	0A	0D	7E	F8	50	1A	C4	07	57	B8	3C	62	E3	C8
e	AC	52	64	10	D0	D9	13	0C	12	29	51	B9	CF	D6	73	8D
f	81	54	C0	ED	4E	44	A7	2A	85	25	E6	CA	7C	8B	56	80

Рис. 1

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	CE	BB	EB	92	EA	CB	13	C1	E9	3A	D6	B2	D2	90	17	F8
1	42	15	56	B4	65	1C	88	43	C5	5C	36	BA	F5	57	67	8D
2	31	F6	64	58	9E	F4	22	AA	75	0F	02	B1	DF	6D	73	4D
3	7C	26	2E	F7	08	5D	44	3E	9F	14	C8	AE	54	10	D8	BC
4	1A	6B	69	F3	BD	33	AB	FA	D1	9B	68	4E	16	95	91	EE
5	4C	63	8E	5B	CC	3C	19	A1	81	49	7B	D9	6F	37	60	CA
6	E7	2B	48	FD	96	45	FC	41	12	0D	79	E5	89	8C	E3	20
7	30	DC	B7	6C	4A	B5	3F	97	D4	62	2D	06	A4	A5	83	5F
8	2A	DA	C9	00	7E	A2	55	BF	11	D5	9C	CF	0E	0A	3D	51
9	7D	93	1B	FE	C4	47	09	86	0B	8F	9D	6A	07	B9	B0	98
a	18	32	71	4B	EF	3B	70	A0	E4	40	FF	C3	A9	E6	78	F9
b	8B	46	80	1E	38	E1	B8	A8	E0	0C	23	76	1D	25	24	05
c	F1	6E	94	28	9A	84	E8	A3	4F	77	D3	85	E2	52	F2	82
d	50	7A	2F	74	53	B3	61	AF	39	35	DE	CD	1F	99	AC	AD
e	72	2C	DD	D0	87	BE	5E	A6	EC	04	C6	03	34	FB	DB	59
f	B6	C2	01	F0	5A	ED	A7	66	21	7F	8A	27	C7	C0	29	D7

Рис. 2

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	93	D9	9A	B5	98	22	45	FC	BA	6A	DF	02	9F	DC	51	59
1	4A	17	2B	C2	94	F4	BB	A3	62	E4	71	D4	CD	70	16	E1
2	49	3C	C0	D8	5C	9B	AD	85	53	A1	7A	C8	2D	E0	D1	72
3	A6	2C	C4	E3	76	78	B7	B4	09	3B	0E	41	4C	DE	B2	90
4	25	A5	D7	03	11	00	C3	2E	92	EF	4E	12	9D	7D	CB	35
5	10	D5	4F	9E	4D	A9	55	C6	D0	7B	18	97	D3	36	E6	48
6	56	81	8F	77	CC	9C	B9	E2	AC	B8	2F	15	A4	7C	DA	38
7	1E	0B	05	D6	14	6E	6C	7E	66	FD	B1	E5	60	AF	5E	33
8	87	C9	F0	5D	6D	3F	88	8D	C7	F7	1D	E9	EC	ED	80	29
9	27	CF	99	A8	50	0F	37	24	28	30	95	D2	3E	5B	40	83
a	B3	69	57	1F	07	1C	8A	BC	20	EB	CE	8E	AB	EE	31	A2
b	73	F9	CA	3A	1A	FB	0D	C1	FE	FA	F2	6F	BD	96	DD	43
c	52	B6	08	F3	AE	BE	19	89	32	26	B0	EA	4B	64	84	82
d	6B	F5	79	BF	01	5F	75	63	1B	23	3D	68	2A	65	E8	91
e	F6	FF	13	58	F1	47	0A	7F	C5	A7	E7	61	5A	06	46	44
f	42	04	A0	DB	39	86	54	AA	8C	34	21	8B	F8	0C	74	67

Рис. 3

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	68	8D	CA	4D	73	4B	4E	2A	D4	52	26	B3	54	1E	19	1F
1	22	03	46	3D	2D	4A	53	83	13	8A	B7	D5	25	79	F5	BD
2	58	2F	0D	02	ED	51	9E	11	F2	3E	55	5E	D1	16	3C	66
3	70	5D	F3	45	40	CC	E8	94	56	08	CE	1A	3A	D2	E1	DF
4	B5	38	6E	0E	E5	F4	F9	86	E9	4F	D6	85	23	CF	32	99
5	31	14	AE	EE	C8	48	D3	30	A1	92	41	B1	18	C4	2C	71
6	72	44	15	FD	37	BE	5F	AA	9B	88	D8	AB	89	9C	FA	60
7	EA	BC	62	0C	24	A6	A8	EC	67	20	DB	7C	28	DD	AC	5B
8	34	7E	10	F1	7B	8F	63	A0	05	9A	43	77	21	BF	27	09
9	C3	9F	B6	D7	29	C2	EB	C0	A4	8B	8C	1D	FB	FF	C1	B2
a	97	2E	F8	65	F6	75	07	04	49	33	E4	D9	B9	D0	42	C7
b	6C	90	00	8E	6F	50	01	C5	DA	47	3F	CD	69	A2	E2	7A
c	A7	C6	93	0F	0A	06	E6	2B	96	A3	1C	AF	6A	12	84	39
d	E7	B0	82	F7	FE	9D	87	5C	81	35	DE	B4	A5	FC	80	EF
e	CB	BB	6B	76	BA	5A	7D	78	0B	95	E3	AD	74	98	3B	36
f	64	6D	DC	F0	59	A9	4C	17	7F	91	B8	C9	57	1B	E0	61

Рис. 4

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	FC	EE	DD	11	CF	6E	31	16	FB	C4	FA	DA	23	C5	04	4D
1	E9	77	F0	DB	93	2E	99	BA	17	36	F1	BB	14	CD	5F	C1
2	F9	18	65	5A	E2	5C	EF	21	81	1C	3C	42	8B	01	8E	4F
3	05	84	02	AE	E3	6A	8F	A0	06	0B	ED	98	7F	D4	D3	1F
4	EB	34	2C	51	EA	C8	48	AB	F2	2A	68	A2	FD	3A	CE	CC
5	B5	70	0E	56	08	0C	76	12	BF	72	13	47	9C	B7	5D	87
6	15	A1	96	29	10	7B	9A	C7	F3	91	78	6F	9D	9E	B2	B1
7	32	75	19	3D	FF	35	8A	7E	6D	54	C6	80	C3	BD	0D	57
8	DF	F5	24	A9	3E	A8	43	C9	D7	79	D6	F6	7C	22	B9	03
9	E0	0F	EC	DE	7A	94	B0	BC	DC	E8	28	50	4E	33	0A	4A
A	A7	97	60	73	1E	00	62	44	1A	B8	38	82	64	9F	26	41
B	AD	45	46	92	27	5E	55	2F	8C	A3	A5	7D	69	D5	95	3B
C	07	58	B3	40	86	AC	1D	F7	30	37	6B	E4	88	D9	E7	89
D	E1	1B	83	49	4C	3F	F8	FE	8D	53	AA	90	CA	D8	85	61
E	20	71	67	A4	2D	2B	09	5B	CB	9B	25	D0	BE	E5	6C	52
F	59	A6	74	D2	E6	F4	B4	C0	D1	66	AF	C2	39	4B	63	B6

Рис. 5

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	B1	94	BA	C8	0A	08	F5	3B	36	6D	00	8E	58	4A	5D	E4
1	85	04	FA	9D	1B	B6	C7	AC	25	2E	72	C2	02	FD	CE	0D
2	5B	E3	D6	12	17	B9	61	81	FE	67	86	AD	71	6B	89	0B
3	5C	B0	C0	FF	33	C3	56	B8	35	C4	05	AE	D8	E0	7F	99
4	E1	2B	DC	1A	E2	82	57	EC	70	3F	CC	F0	95	EE	8D	F1
5	C1	AB	76	38	9F	E6	78	CA	F7	C6	F8	60	D5	BB	9C	4F
6	F3	3C	65	7B	63	7C	30	6A	DD	4E	A7	79	9E	B2	3D	31
7	3E	98	B5	6E	27	D3	BC	CF	59	1E	18	1F	4C	5A	B7	93
8	E9	DE	E7	2C	8F	0C	0F	A6	2D	DB	49	F4	6F	73	96	47
9	06	07	53	16	ED	24	7A	37	39	CB	A3	83	03	A9	8B	F6
A	92	BD	9B	1C	E5	D1	41	01	54	45	FB	C9	5E	4D	0E	F2
B	68	20	80	AA	22	7D	64	2F	26	87	F9	34	90	40	55	11
C	BE	32	97	13	43	FC	9A	48	A0	2A	88	5F	19	4B	09	A1
D	7E	CD	A4	D0	15	44	AF	8C	A5	84	50	BF	66	D2	E8	8A
E	A2	D7	46	52	42	A8	DF	B3	69	74	C5	51	EB	23	29	21
F	D4	EF	D9	B4	3A	62	28	75	91	14	10	EA	77	6C	DA	1D

Рис. 6

		y														
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Рис. 7

БСШ «Camellia» – алгоритм шифрования с размером блока 128 бит и ключа 128, 192, 256 бит. Один из финалистов европейского конкурса NESSIE (наряду с AES и Shacal-2), разработка японских компаний Nippon Telegraph and Telephone Corporation и Mitsubishi Electric Corporation. Рекомендован CRYPTREC для промышленного и государственного использования. Стандартизирован на международном уровне в ISO/IEC 18033-3 [2]. Цикловая функция использует нелинейное преобразование, блок линейного рассеивания и байтовую перестановку. Четыре S-блока этого шифра приведены на рис. 8–11.

БСШ «CAST» – канадский криптоалгоритм, разработанный Карлайслом Адамсом (Carlisle Adams) и Стаффордом Таваресом (Stafford Tavares). Использует 64-битовый блок и 128-битовый ключ, 16 раундов преобразования. В ISO/IEC 18033-3 [2] стандартизирован алгоритм CAST-128, который использует восемь S-блоков, реализующих отображение 8-ми битных входных векторов в 32-х битные выходные. Четыре S-блока используются в раундовой функции

алгоритма, оставшиеся четыре используются в схеме разворачивания ключа. В данной работе исследованы все восемь таблиц подстановок. Их описание приведено в [2].

БСШ «SEED» – симметричный криптоалгоритм, разработанный корейским агентством информационной безопасности (Korean Information Security Agency, KISA) в 1998 году. Также стандартизирован на международном уровне в ISO/IEC 18033-3 [2]. Алгоритм широко используется финансовыми и банковскими структурами, производственными предприятиями и бюджетными учреждениями Южной Кореи, а также в протоколах TLS и S/MIME. Алгоритм представляет собой сеть Фейстеля с 16 раундами, 128-битовыми блоками и 128-битовым ключом. Использует две таблицы подстановки, полученные на основе дискретного возведения в степень (см. рис. 12, 13).

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	70	82	2c	ec	b3	27	c0	e5	e4	85	57	35	ea	0c	ae	41
1	23	ef	6b	93	45	19	a5	21	ed	0e	4f	4e	1d	65	92	bd
2	86	b8	af	8f	7c	eb	1f	ce	3e	30	dc	5f	5e	c5	0b	1a
3	a6	e1	39	ca	d5	47	5d	3d	d9	01	5a	d6	51	56	6c	4d
4	8b	0d	9a	66	fb	cc	b0	2d	74	12	2b	20	f0	b1	84	99
5	df	4c	cb	c2	34	7e	76	05	6d	b7	a9	31	d1	17	04	d7
6	14	58	3a	61	de	1b	11	1c	32	0f	9c	16	53	18	f2	22
7	fe	44	cf	b2	c3	b5	7a	91	24	08	e8	a8	60	fc	69	50
8	aa	d0	a0	7d	a1	89	62	97	54	5b	1e	95	e0	ff	64	d2
9	10	c4	00	48	a3	f7	75	db	8a	03	e6	da	09	3f	dd	94
a	87	5c	83	02	cd	4a	90	33	73	67	f6	f3	9d	7f	bf	e2
b	52	9b	d8	26	c8	37	c6	3b	81	96	6f	4b	13	be	63	2e
c	e9	79	a7	8c	9f	6e	bc	8e	29	f5	f9	b6	2f	fd	b4	59
d	78	98	06	6a	e7	46	71	ba	d4	25	ab	42	88	a2	8d	fa
e	72	07	b9	55	f8	ee	ac	0a	36	49	2a	68	3c	38	f1	a4
f	40	28	d3	7b	bb	c9	43	c1	15	e3	ad	f4	77	c7	80	9e

Рис. 8

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	e0	05	58	d9	67	4e	81	cb	c9	0b	ae	6a	d5	18	5d	82
1	46	df	d6	27	8a	32	4b	42	db	1c	9e	9c	3a	ca	25	7b
2	0d	71	5f	1f	f8	d7	3e	9d	7c	60	b9	be	bc	8b	16	34
3	4d	c3	72	95	ab	8e	ba	7a	b3	02	b4	ad	a2	ac	d8	9a
4	17	1a	35	cc	f7	99	61	5a	e8	24	56	40	e1	63	09	33
5	bf	98	97	85	68	fc	ec	0a	da	6f	53	62	a3	2e	08	af
6	28	b0	74	c2	bd	36	22	38	64	1e	39	2c	a6	30	e5	44
7	fd	88	9f	65	87	6b	f4	23	48	10	d1	51	c0	f9	d2	a0
8	55	a1	41	fa	43	13	c4	2f	a8	b6	3c	2b	c1	ff	c8	a5
9	20	89	00	90	47	ef	ea	b7	15	06	cd	b5	12	7e	bb	29
a	0f	b8	07	04	9b	94	21	66	e6	ce	ed	e7	3b	fe	7f	c5
b	a4	37	b1	4c	91	6e	8d	76	03	2d	de	96	26	7d	c6	5c
c	d3	f2	4f	19	3f	dc	79	1d	52	eb	f3	6d	5e	fb	69	b2
d	f0	31	0c	d4	cf	8c	e2	75	a9	4a	57	84	11	45	1b	f5
e	e4	0e	73	aa	f1	dd	59	14	6c	92	54	d0	78	70	e3	49
f	80	50	a7	f6	77	93	86	83	2a	c7	5b	e9	ee	8f	01	3d

Рис. 9

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	38	41	16	76	d9	93	60	f2	72	c2	ab	9a	75	06	57	a0
1	91	f7	b5	c9	a2	8c	d2	90	f6	07	a7	27	8e	b2	49	de
2	43	5c	d7	c7	3e	f5	8f	67	1f	18	6e	af	2f	e2	85	0d
3	53	f0	9c	65	ea	a3	ae	9e	ec	80	2d	6b	a8	2b	36	a6
4	c5	86	4d	33	fd	66	58	96	3a	09	95	10	78	d8	42	cc
5	ef	26	e5	61	1a	3f	3b	82	b6	db	d4	98	e8	8b	02	eb
6	0a	2c	1d	b0	6f	8d	88	0e	19	87	4e	0b	a9	0c	79	11
7	7f	22	e7	59	e1	da	3d	c8	12	04	74	54	30	7e	b4	28
8	55	68	50	be	d0	c4	31	cb	2a	ad	0f	ca	70	ff	32	69
9	08	62	00	24	d1	fb	ba	ed	45	81	73	6d	84	9f	ee	4a
a	c3	2e	c1	01	e6	25	48	99	b9	b3	7b	f9	ce	bf	df	71
b	29	cd	6c	13	64	9b	63	9d	c0	4b	b7	a5	89	5f	b1	17
c	f4	bc	d3	46	cf	37	5e	47	94	fa	fc	5b	97	fe	5a	ac
d	3c	4c	03	35	f3	23	b8	5d	6a	92	d5	21	44	51	c6	7d
e	39	83	dc	aa	7c	77	56	05	1b	a4	15	34	1e	1c	f8	52
f	20	14	e9	bd	dd	e4	a1	e0	8a	f1	d6	7a	bb	e3	40	4f

Рис. 10

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	70	2c	b3	c0	e4	57	ea	ae	23	6b	45	a5	ed	4f	1d	92
1	86	af	7c	1f	3e	dc	5e	0b	a6	39	d5	5d	d9	5a	51	6c
2	8b	9a	fb	b0	74	2b	f0	84	df	cb	34	76	6d	a9	d1	04
3	14	3a	de	11	32	9c	53	f2	fe	cf	c3	7a	24	e8	60	69
4	aa	a0	a1	62	54	1e	e0	64	10	00	a3	75	8a	e6	09	dd
5	87	83	cd	90	73	f6	9d	bf	52	d8	c8	c6	81	6f	13	63
6	e9	a7	9f	bc	29	f9	2f	b4	78	06	e7	71	d4	ab	88	8d
7	72	b9	f8	ac	36	2a	3c	f1	40	d3	bb	43	15	ad	77	80
8	82	ec	27	e5	85	35	0c	41	ef	93	19	21	0e	4e	65	bd
9	b8	8f	eb	ce	30	5f	c5	1a	e1	ca	47	3d	01	d6	56	4d
a	0d	66	cc	2d	12	20	b1	99	4c	c2	7e	05	b7	31	17	d7
b	58	61	1b	1c	0f	16	18	22	44	b2	b5	91	08	a8	fc	50
c	d0	7d	89	97	5b	95	ff	d2	c4	48	f7	db	03	da	3f	94
d	5c	02	4a	33	67	f3	7f	e2	9b	26	37	3b	96	4b	be	2e
e	79	8c	6e	8e	f5	b6	fd	59	98	6a	46	ba	25	42	a2	fa
f	07	55	ee	0a	49	68	38	a4	28	7b	c9	c1	e3	f4	c7	9e

Рис. 11

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	a9	85	d6	d3	54	1d	ac	25	5d	43	18	1e	51	fc	ca	63
1	28	44	20	9d	e0	e2	c8	17	a5	8f	03	7b	bb	13	d2	ee
2	70	8c	3f	a8	32	dd	f6	74	ec	95	0b	57	5c	5b	bd	01
3	24	1c	73	98	10	cc	f2	d9	2c	e7	72	83	9b	d1	86	c9
4	60	50	a3	eb	0d	b6	9e	4f	b7	5a	c6	78	a6	12	af	4f
5	61	c3	b4	41	52	7d	8d	08	1f	99	00	19	04	53	f7	e1
6	fd	76	2f	27	60	8b	0e	ab	a2	6e	93	4d	69	7c	09	0a
7	bf	ef	f3	c5	87	14	fe	64	de	2e	4b	1a	06	21	6b	66
8	02	f5	92	8a	0c	b3	7e	d0	7a	47	96	e5	26	80	ad	df
9	a1	30	37	ae	36	15	22	38	f4	a7	45	4c	81	e9	84	97
a	35	c6	ce	3c	71	11	c7	89	75	fb	da	f8	94	59	82	c4
b	ff	49	39	67	c0	cf	d7	b8	0f	8e	42	23	91	6c	db	a4
c	34	f1	48	c2	6f	3d	2d	40	be	3e	bc	c1	aa	ba	4e	55
d	3b	dc	68	7f	9c	d8	4a	56	77	a0	ed	46	b5	2b	65	fa
e	e3	b9	b1	9f	5e	f9	e6	b2	31	ea	6d	5f	e4	f0	cd	88
f	16	3a	58	d4	62	29	07	33	e8	1b	05	79	90	6a	2a	9a

Рис. 12

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	38	e8	2d	a6	cf	de	b3	b8	af	60	55	c7	44	6f	6b	5b
1	c3	62	33	b5	29	a0	e2	a7	d3	91	11	06	1c	bc	36	4b
2	ef	88	6c	a8	17	c4	16	f4	c2	45	e1	d6	3f	3d	8e	98
3	28	4e	f6	3e	a5	f9	0d	df	d8	2b	66	7a	27	2f	f1	72
4	42	d4	41	c0	73	67	ac	8b	f7	ad	80	1f	ca	2c	aa	34
5	d2	0b	ee	e9	5d	94	18	f8	57	ae	08	c5	13	cd	86	b9
6	ff	7d	c1	31	f5	8a	6a	b1	d1	20	d7	02	22	04	68	71
7	07	db	9d	99	61	be	e6	59	dd	51	90	dc	9a	a3	ab	d0
8	81	0f	47	1a	e3	ec	8d	bf	96	7b	5c	a2	a1	63	23	4d
9	c8	9e	9c	3a	0c	2e	ba	6e	9f	5a	f2	92	f3	49	78	cc
a	15	fb	70	75	7f	35	10	03	64	6d	c6	74	d5	b4	ea	09
b	76	19	fe	40	12	e0	bd	05	fa	01	f0	2a	5e	a9	56	43
c	85	14	89	9b	b0	e5	48	79	97	fc	1e	82	21	8c	1b	5f
d	77	54	b2	1d	25	4f	00	46	ed	58	52	eb	7e	da	c9	fd
e	30	95	65	3c	b6	e4	bb	7c	0e	50	39	26	32	84	69	93
f	37	e7	24	a4	c6	53	0a	87	d9	4c	83	8f	ce	3b	4a	b7

Рис. 13

2. КРИТЕРИИ И ПОКАЗАТЕЛИ ЭФФЕКТИВНОСТИ НЕЛИНЕЙНЫХ УЗЛОВ ЗАМЕН

Рассмотрим критерии и показатели эффективности нелинейных узлов замен, непосредственно влияющие на уровень стойкости современных БСШ к различным криптоаналитическим атакам.

Под эффективностью функционирования любой технической системы понимают соответствие полученных результатов функционирования требуемому. Очевидно, что основным требованием к нелинейным узлам является обеспечение стойкости к известным методам криптографического анализа. Другими словами, нелинейный узел будем считать эффективным, если он обеспечивает стойкость к известным на сегодняшний день методам криптографического анализа.

В большинстве известных работ в области анализа и синтеза нелинейных узлов замен современных БСШ используется математический аппарат криптографических булевых функций [7–12]. При этом каждый S-блок представляется совокупностью компонентных булевых функций, свойства которых характеризуют эффективность нелинейного узла замен. В качестве основных критериев и показателей эффективности используют [7–12]: сбалансированность и нелинейность компонентных булевых функций; корреляционный иммунитет; критерий распространения; алгебраическая степень; значение функции автокорреляции.

Введем основные понятия и определения, используемые в дальнейшем при оценке эффективности нелинейных узлов замен [7–12].

Булевой функцией f от n переменных является функция, осуществляющая отображение из поля $GF(2^n)$ всех двоичных векторов $x = (x_1, ..., x_n)$ длины n в поле $GF(2)$. Обычно булевы функции представляются в алгебраической нормальной форме. *Поле* $GF(2^n)$ состоит из 2^n векторов α_i: $\alpha_0 = (0,...,0,0)$, $\alpha_1 = (0,...,0,1)$,..., $\alpha_{2^n-1} = (1,...,1,1)$, $\alpha_i \in V_n$, где V_n – векторное пространство в $GF(2^n)$.

Последовательностью функции f называется (1,-1)-последовательность, определенная как $((-1)^{f(\alpha_0)}, (-1)^{f(\alpha_1)}, ..., (-1)^{f(\alpha_{2^n-1})})$. *Таблицей истинности функции f* называется (0,1)-последовательность, определенная как $(f(\alpha_0), f(\alpha_1), ..., f(\alpha_{2^n-1}))$.

Последовательность функции f является *сбалансированной*, если ее (0,1)-последовательность ((1,-1)-последовательность) содержит одинаковое количество нулей и единиц (единиц и минус единиц). Функция f является сбалансированной, если сбалансирована ее последовательность.

Эквивалентное определение: функция f над $GF(2^n)$ является сбалансированной, если ее выходные значения являются равновероятными:

$$|\{x|f(x)=0\}| = |\{x|f(x)=1\}| = 2^n - 1$$

Аффинной функцией f называется функция вида $f = a_1x_1 \oplus ... \oplus a_nx_n \oplus c$, где a_j, $c \in GF(2)$, $j = 1, 2,..., n$. Функция f называется линейной, если $c = 0$.

Весом Хэмминга вектора α, обозначаемым как $W(\alpha)$, является количество единиц в векторе (последовательности). *Расстоянием Хэмминга d(f,g)* между последовательностями двух функций f и g является количество позиций, в которых различны последовательности этих функций.

Нелинейность N_S преобразования – минимальное расстояние Хэмминга между выходной последовательностью S и всеми выходными последовательностями аффинных функций над некоторым полем:

$$N_s = \min\{d(S,\varphi)\},$$

где φ – множество аффинных функций.

Нелинейность функции N_f – минимальное расстояние Хэмминга N_f между функцией f и всеми аффинными функциями над $GF(2^n)$:

$$N_f = \min\{d(f,\varphi)\},$$

где φ – множество аффинных функций.

Для произвольной функции f нелинейность N_f над $GF(2^n)$ может достигать $N_f \le 2^{n-1} - 2^{n/2-1}$. Для сбалансированной функции f над $GF(2^n)$ ($n \ge 3$) нелинейность N_f может достигать:

$$N_f \le \begin{cases} 2^{n-1} - 2^{\frac{n}{2}-1} - 2, n = 2k \\ \left\lfloor\left\lfloor 2^{n-1} - 2^{\frac{n}{2}-1} \right\rfloor\right\rfloor, n = 2k = 1, \end{cases}$$

где $\lfloor\lfloor x \rfloor\rfloor$ – максимальное четное целое, меньшее либо равное x.

Функция f обладает *корреляционным иммунитетом* порядка k, если выходная последовательность функции $y \in Y$ статистически не зависит от любого подмножества из k входных координат:

$$\forall\{x_1,...,x_k\}\ P(y\in Y\mid \{x_1,...,x_k\}\in X) = P(y\in Y).$$

Эквивалентное определение корреляционного иммунитета в терминах преобразования Уолша: функция f над полем $GF(2^n)$ имеет корреляционный иммунитет порядка k, $КИ(k)$, если ее преобразование Уолша удовлетворяет равенству $F(\omega) = 0$ для всех $\omega \in V_n$ таких, что $1 \le W(\omega) \le k$:

$$\forall\omega\in V_n\ F(\varpi) = 0 \qquad (f) = k\ .$$

Преобразование Уолша $F(\omega)$ функции f над полем $GF(2^n)$ определяется как принимающая действительные значения функция

$$F(\omega) = 2^{-n}\sum\nolimits_x (-1)^{f(x)\oplus\langle\varpi,x\rangle},$$

где $\omega \in V_n$, $f(x)$, $\langle\omega,x\rangle \in N$ ($\langle\omega, x\rangle$ – скалярное произведение $w_1x_1 \oplus ... \oplus w_nx_n$).

Корреляционно-иммунная функция k-го порядка – функция, обладающая корреляционным иммунитетом порядка k. Сбалансированные корреляционно-иммунные функции называются *эластичными функциями*.

Функция f над полем $GF(2^n)$ удовлетворяет:

– *критерию распространения* относительно вектора a, КР(a), если функция $f(x) \oplus f(x \oplus a)$ является сбалансированной, $x \in V_n$, где $x = (x_1, x_2, ..., x_n)$

$$P(f(x) = f(x\oplus\alpha)) = \frac{1}{2};$$

– *критерию распространения степени* k, КР(k), если удовлетворяется критерий распространения относительно всех векторов $a \in V_n$ при $1 \le W(a) \le k$:

$$P(f(x) = f(x\oplus\alpha)) = \frac{1}{2}\ \forall\alpha : 1\le W(\alpha)\le k;$$

– *строгому лавинному критерию* (СЛК), если f удовлетворяет критерию распространения степени 1:

$$P(f(x) = f(x\oplus\alpha)) = \frac{1}{2}\ \forall\alpha : W(\alpha) = 1.$$

Алгебраическая степень deg(f) является степенью самого длинного слагаемого функции, представленной в алгебраической нормальной форме.

Автокорреляционная функция $\hat{r}(s)$ для $s\in 0...2^n-1$ определена как

$$\hat{r}(s) = \sum_{x=0}^{2^n-1}\hat{f}(x)\hat{f}(x\oplus s).$$

Говорят, что функция f удовлетворяет *характеристике распространения* m, если

$$(1\le|s|\le m)\Rightarrow\left|\hat{r}(s)\right| = 0.$$

Аналогично, автокорреляция $AC(f)$ функции f определяется как модуль наибольшего значения $\hat{r}(s)$:

$$AC(f) = \max_{s\ne 0}\left|\sum_x \hat{f}(x)\hat{f}(x\oplus s)\right| = \max_{s\ne 0}\left|\hat{r}(s)\right|.$$

Автокорреляция обеспечивает утечку информационного потока со входа на выход функции. В некоторых случаях очень сильная взаимосвязь может быть представлена как линейная структура (для которых справедливы равенства $f(x) \oplus f(x\oplus s) = 1$ или $f(x) \oplus f(x\oplus s) = 0$). Они, как правило, избегаемы.

Рассмотренные критерии и показатели эффективности S-блоков отражают способность нелинейного узла противостоять атакам определенного типа. Нелинейность, критерий распространения и корреляционная иммунность характеризуют способность противостоять корреляционным атакам, алгебраическая степень и автокорреляция – аналитическим атакам, сбалансированность – статистическим.

Проведем исследования эффективности нелинейных узлов замен современных БСШ «Калина», «Кузнечик», «BelT», «AES», «SEED», «Camellia», «CAST».

3. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЙ

Для проведения экспериментальных исследований эффективности различных S-блоков в среде IntelliJ IDEA 15 Community edition на языке Java был разработан программный вычислительный комплекс. Он позволяет рассчитать основные показатели эффективности нелинейных узлов замены: сбалансированность, нелинейность, автокорреляцию, алгебраическую степень, степень критерия распространения и корреляционного иммунитета криптографических булевых функций. Показатели эффективности S-блока оцениваются по критерию минимального риска (худший случай) по всем компонентным булевым функциям и их линейным комбинациям. Результаты экспериментальных исследований сведены в табл. 1–5. Использованы следующие обозначения: B – сбалансированность; N – нелинейность; A – автокорреляция; AD – алгебраическая степень; PC – критерий распространения; CI – корреляционная иммунность; f_n – функция, соответствующая n-му выходу S-блока; S-box – показатели эффективности S-блока по критерию минимального риска (худший случай среди всех компонентных булевых функций); Linear combinations (LC) – показатели эффективности S-блока по критерию минимального риска (худший случай среди всех линейных комбинаций булевых функций).

Полученные результаты показали разнообразие мнений разработчиков исследуемых шифров относительно концепций формирования S-блоков. Так, например, БСШ «CAST-128» имеет превосходные результаты по показателям нелинейности, автокорреляции и критерию распространения, однако имеет низкую алгебраическую степень, а также низкие показатели стойкости для линейных комбинаций булевых функций S-блока. Разработчики БСШ «SEED» также сделали упор на увеличение эффективности таблицы подстановки за счет снижения автокорреляции. Однако их S-блок имеет меньшее значение нелинейности, а также является несбалансированным.

Таблица 1

Показатели эффективности подстановок шифра «Калина»

	S_1-блок						S_2-блок						S_3-блок						S_4-блок					
	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI
f_1	+	106	48	7	0	0	+	106	48	7	0	0	+	106	56	7	0	0	+	108	56	7	0	0
f_2	+	106	64	7	0	0	+	106	64	7	0	0	+	104	64	7	0	0	+	104	56	7	0	0
f_3	+	106	56	7	0	0	+	106	56	7	0	0	+	106	56	7	0	0	+	108	64	7	0	0
f_4	+	104	72	7	0	0	+	104	72	7	0	0	+	106	56	7	0	0	+	108	48	7	0	0
f_5	+	104	56	7	0	0	+	104	56	7	0	0	+	108	48	7	0	0	+	104	64	7	0	0
f_6	+	106	56	7	0	0	+	106	56	7	0	0	+	104	56	7	0	0	+	108	56	7	0	0
f_7	+	106	64	7	0	0	+	106	64	7	0	0	+	106	56	7	0	0	+	104	64	7	0	0
f_8	+	106	64	7	0	0	+	106	64	7	0	0	+	104	56	7	0	0	+	108	48	7	0	0
S-box	+	104	72	7	0	0	+	104	72	7	0	0	+	104	64	7	0	0	+	104	64	7	0	0
LC	+	104	72	7	0	0	+	104	72	7	0	0	+	104	72	7	0	0	+	104	72	7	0	0

Таблица 2

Показатели эффективности подстановок шифров «Кузнечик», «BelT», «AES», «Camellia»

	S-блок «Кузнечик»						S-блок «BelT»						S-блок «AES»						S-блок «Camellia»					
	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI
f1	+	104	64	7	0	0	+	106	56	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
f2	+	106	56	7	0	0	+	106	72	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
f3	+	116	24	7	0	0	+	106	56	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
f4	+	104	64	7	0	0	+	104	72	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
f5	+	110	48	7	0	0	+	108	56	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
f6	+	106	64	7	0	0	+	106	72	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
f7	+	102	72	7	0	0	+	108	56	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
f8	+	104	64	7	0	0	+	108	64	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
S-box	+	102	72	7	0	0	+	104	72	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0
LC	+	102	80	7	0	0	+	102	72	7	0	0	+	112	32	7	0	0	+	112	32	7	0	0

Таблица 3

Показатели эффективности подстановок шифра «SEED»

S_1-блок	B	N	A	AD	PC	CI	S_2-блок	B	N	A	AD	PC	CI
f_1	+	110	40	7	0	0	f_1	+	112	32	7	0	0
f_2	-	111	36	8	0	0	f_2	+	112	32	7	0	0
f_3	+	112	32	7	0	0	f_3	+	112	32	7	0	0
f_4	+	112	40	7	0	0	f_4	+	112	32	7	0	0
f_5	+	110	40	7	0	0	f_5	+	111	36	8	0	0
f_6	-	111	36	8	0	0	f_6	-	113	36	8	0	0
f_7	-	111	36	8	0	0	f_7	+	112	32	7	0	0
f_8	+	111	36	8	0	0	f_8	+	111	36	8	0	0
S-box	-	110	40	7	0	0	S-box	-	111	36	7	0	0
LC	-	109	44	7	0	0	LC	+	111	36	7	0	0

Таблица 4

Показатели эффективности подстановок шифра «CAST-128»

	S_1-блок						S_2-блок						S_3-блок						S_4-блок					
	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI
f1	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f2	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f3	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f4	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f5	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f6	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f7	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f8	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f9	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f10	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f11	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f12	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f13	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f14	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f15	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f16	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f17	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f18	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f19	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f20	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0

f21	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f22	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f23	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f24	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f25	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f26	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f27	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f28	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f29	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f30	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f31	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f32	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
S-box	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
LC	-	88	112	4	0	0	+	92	96	4	0	0	-	92	96	4	0	0	+	92	112	4	0	0

Таблица 5

Показатели эффективности подстановок шифра «CAST-128»

	S_5-блок						S_6-блок						S_7-блок						S_8-блок					
	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI	B	N	A	AD	PC	CI
f1	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f2	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f3	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f4	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f5	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f6	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f7	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f8	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f9	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f10	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f11	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f12	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f13	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f14	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f15	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f16	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0
f17	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f18	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f19	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f20	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f21	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f22	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f23	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f24	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f25	-	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f26	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f27	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f28	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
f29	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f30	+	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
f31	-	120	0	4	8	0	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0
f32	+	120	0	4	8	0	-	120	0	4	8	0	+	120	0	4	8	0	+	120	0	4	8	0
S-box	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0	-	120	0	4	8	0
LC	+	92	144	4	0	0	+	80	112	4	0	0	+	92	96	4	0	0	+	92	128	4	0	0

Таблица подстановки БСШ «AES» имеет высокую алгебраическую степень, хорошие показатели по нелинейности и автокорреляции, однако имеет нулевой критерий распространения. Кроме того, S-блок БСШ «AES» сформирован на основе алгебраической конструкции Нибер-Динга, что создает предпосылки для возможной реализации алгебраического криптоанализа. БСШ «Camellia» показал одинаковые с «AES» результаты. Остальные проанализированные S-блоки не являются алгебраическими, но при этом теряют в нелинейности и автокорреляции.

Таблицы подстановки шифров, утвержденных в качестве государственных стандартов Украины, Российской Федерации и Белоруссии представляют собой наиболее сбалансированное, компромиссное решение. Среди них лучшими показателями обладает S-блок украинского шифра «Калина», вслед за ним идет белорусский «BelT» и российский «Кузнечик».

ВЫВОДЫ

Полученные результаты позволяют судить об основных концепциях формирования нелинейных узлов замен современных БСШ. Как правило, нелинейные узлы разрабатываются с учетом наиболее вероятных угроз и особое внимание уделяется специфическим показателям эффективности. Перспективные шифры должны противостоять всем наиболее распространенным существующим атакам.

Выбранные показатели эффективности нелинейных узлов замен характеризуют первоочередные требования к обеспечению их стойкости. Полученные результаты не позволяют однозначно определить лучший S-бокс, из-за различных концептуальных особенностей в проектировании каждого из них. Однако среди шифров, которые стандартизированы в постсоветских странах, лучшим по показателям эффективности нелинейных узлов замен следует отметить S-блок шифра «Калина». Также стоит обратить внимание на канадский шифр CAST-128, булевы функции в S-блоке которого удовлетворяют критерию распространения степени 8 и обладают нулевой автокорреляцией.

Перспективным направлением дальнейших исследований является анализ свойств нелинейных узлов усложнения поточных криптоалгоритмов, обоснование рекомендаций и предложений по разработке национального стандарта поточного шифрования Украины.

Литература

[1] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Теорія. Практика. Застосування: Підручник для вищих навч. закладів. – Харків: Вид-во «Форт», 2013. – 880 с.

[2] Information technology – Security techniques – Encryption algorithms, Part 3: Block ciphers (ISO/IEC 18033-3), 80 с.

[3] ГОСТ Р 34.12-2015. Информационная технология. Криптографическая защита информации. Блочные шифры. – М.: Стандартинформ, 2015г. – 25 с.

[4] СТБ 34.101.31-2011 Информационные технологии и безопасность. Защита информации. Криптографические алгоритмы шифрования и контроля целостности. – Минск: Госстандарт, 2011г. – 35 с.

[5] Інформаційні технології. Криптографічний захист інфомації. Алгоритм симетричного блокового перетворення: ДСТУ 7624:2014. – К.: Мінекономрозвитку України, 2015. – 238 с.

[6] Розробка нового блокового симетричного шифру: звіт за перший етап НДР «Алгоритм» (проміжний) / АТ «ІІТ»; кер. І.Д. Горбенко – Харків, 2014, Том 4. – 304 с.

[7] *Bart Preneel.* Analysis and Design of Cryptographic Hash Functions. [Электронный ресурс] – Режим доступа: homes.esat.kuleuven.be/~preneel/phd_preneel_feb1993.pdf

[8] *Carlet C.* Vectorial Boolean functions for // Cambridge Univ. Press, Cambridge. – 95 p. [Электронный ресурс] – Режим доступа: www.math.univ-paris13.fr/~carlet/chap-vectorial-fcts-corr.pdf

[9] *Carlet C.* Boolean functions for cryptography and error correcting codes // Cambridge Univ. Press, Cambridge. – 2007. – 148 p. [Электронный ресурс] – Режим доступа: www1.spms.ntu.edu.sg/~kkhoongm/chap-fcts-Bool.pdf

[10] Zhuo Zepeng, Zhang Weiguo On correlation properties of Boolean functions // Chinese Journal of Electronics. Jan, Vol.20, 2011, №1, 143-146 pp.

[11] O'Connor L. An analysis of a class of algorithms for S-box construction // J. Cryptology. -1994. – p. 133-151.

[12] Clark J.A., Jacob J.L., Stepney S. The Design of S-Boxes by Simulated Annealing // New Generation Computing. – 2005. – 23(3). – p.219–231.

Поступила в редколлегию 13.11.2015

Кузнецов Александр Александрович, фото и сведения об авторе см. на с. 334.

Белозерцев Иван Никитович, студент факультета компьютерных наук ХНУ им. В.Н.Каразина. Научные интересы: криптография, блочные симметричные шифры, теория обработки и передачи данных.

Андрушкевич Алина Вадимовна, аспирантка ХНУРЭ, инженер 1 кат. кафедры БИТ ХНУРЭ. Научные интересы: анализ стойкости симметричных шифров, криптография и аутентификация.

УДК 004.056.55

Аналіз та порівняльні дослідження нелінійних вузлів замін сучасних блокових симетричних шифрів / О.О. Кузнецов, І.М. Білозерцев, А.В. Андрушкевич // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 343–350.

Розглядаються сучасні блокові симетричні шифри і застосовані в них нелінійні вузли заміни (S-блоки). Аналізуються показники та критерії ефективності S-блоків: нелінійність, збалансованість, кореляційний імунітет, критерій розповсюдження, автокореляція та ін. Проводяться порівняльні дослідження ефективності нелінійних вузлів заміни сучасних блокових шифрів.

Ключові слова: блоковий симетричний шифр, нелінійний вузол замін, криптографічна функція.

Табл.: 5. Іл.: 14. Бібліогр.: 12 найм.

UDC 004.056.55

Analysis and comparative research of nonlinear substitution components of the state-of-the-art block ciphers/ A.A. Kuznetsov, I.M. Bilozertsev, A.V. Andrushkevych // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 343–350.

The paper examines state-of-the-art block ciphers and nonlinear substitution components (S-blocks) which are used in them. Properties and criteria of S-block effectiveness such as nonlinearity, balance, correlation immunity, propagation criteria, autocorrelation etc. are analyzed. A comparative research of the effectiveness of nonlinear substitution components of state-of-the-art block ciphers is conducted.

Keywords: symmetric block cipher, nonlinear substitution component, cryptographic function.

Tab.: 5. Fig.: 14. Ref.: 12 items.

УДК 004.056.55

КОМБИНАТОРНЫЕ СВОЙСТВА КЛЮЧЕВОГО РАСПИСАНИЯ БЛОЧНЫХ СИММЕТРИЧНЫХ ШИФРОВ

А.А. КУЗНЕЦОВ, Ю.И. ГОРБЕНКО, Д.В. ИВАНЕНКО

Исследуются комбинаторные свойства ключевого расписания блочных симметричных шифров в предположении, что цикловые (раундовые) ключи формируются случайно, равновероятно и независимо друг от друга. Для абстрактного описания такого формирования используется модель случайной однородной подстановки. Полученные аналитические выражения позволяют оценить мощность множества реализуемых отображений зашифрования-расшифрования, получить оценки вероятностных свойств последовательностей раундовых ключей и отношений среднего числа различных ключевых последовательностей к мощности множества различных мастер-ключей. Результаты имитационного моделирования подтверждают достоверность и обоснованность полученных аналитических выражений.

Ключевые слова: ключевое расписание, цикловые ключи, комбинаторные свойства, блочные симметричные шифры.

1. ПОСТАНОВКА ПРОБЛЕМЫ И АНАЛИЗ ЛИТЕРАТУРЫ

Важным механизмом криптографической защиты информации является шифрование, под которым понимают обратимое преобразование открытых данных для сокрытия их смыслового содержания от неуполномоченного пользователя (противника, злоумышленника). Криптографическое преобразование осуществляется под управлением специальных (ключевых) данных, которые для симметричных функций зашифрования-расшифрования совпадают [1]. Блочный шифр, как разновидность симметричной системы, оперирует группами бит фиксированной длины (блоками). Взаимнооднозначные процессы зашифрования-расшифрования блоков открытого текста (англ. plaintext) и блоков шифртекста (англ. ciphertext) параметризируется ключевыми данными. То есть каждый секретный ключ шифрования (мастер-ключ) задает конкретный вид биективного отображения «открытый текст – шифртекст».

Большинство блочных симметричных шифров (БСШ) являются *итеративными* [1], в которых шифрование реализуется посредством циклически повторяющихся обратимых раундовых функций (рис. 1). На каждой итерации БСШ для параметризации раундовых преобразований используются т.н. *раундовые (цикловые) ключи* $K_1^{(x)}, K_2^{(x)}, ..., K_t^{(x)}$. Они формируются посредством расширения (планирования, расписания) мастер-ключа $K^{(x)}$ (англ. keyscheduling) [1]. Структура ключевого расписания итеративных БСШ, простота формирования и/или взаимозависимость раундовых ключей, лежат в основе известных атак на схемы разворачивания. В частности, наибольшее развитие получили т.н. *сдвиговые атаки* (англ. slideattack) [2–4], *атаки на связанных ключах* (англ. related-keyattack) [5–7] и пр. [1].

В простейшем случае схема разворачивания может состоять в повторении мастер-ключа для каждого раунда. Подобный подход использовался при формирования цикловых ключей в советском алгоритме симметричного криптопреобразования ГОСТ 28147-89, являвшегося, до недавнего времени, и основным стандартом шифрования Украины ДСТУ ГОСТ 28147:2009 [1]. Однако в случае, когда на вход каждой раундовой функции (см. рис. 1) подается некоторый ключ, одинаковый для всех раундов, шифр становится уязвимым к сдвиговой атаке [2, 3]. К этому простейшему случаю легко сводится вариант, когда функция разворачивания предполагает циклическое повторение некоторого набора раундовых ключей (шифры с раундовым самоподобием, англ. roundself-similarityciphers) [4].

Для противодействия криптоаналитическим атакам на ключевое расписание современные БСШ используют усложненные схемы разво-

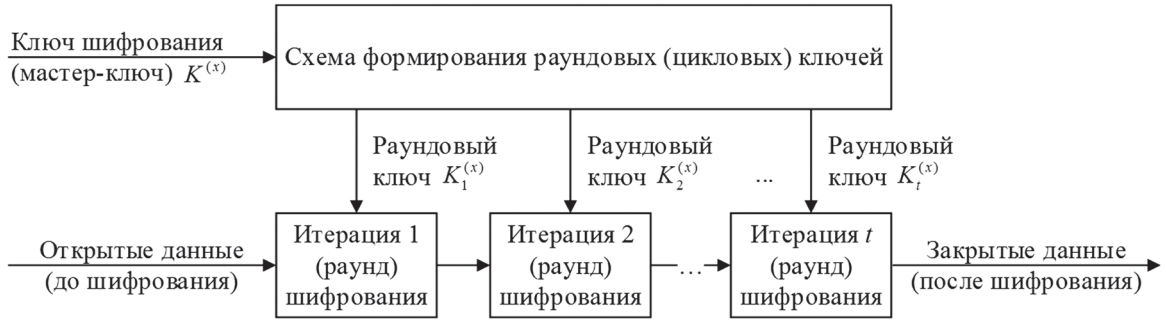


Рис. 1. Структурная схема итеративного блочного шифра

рачивания раундовых ключей, построенные, например, на использовании шифроподобных преобразований. К числу таких БСШ следует отнести, в первую очередь, национальный стандарт США FIPS-197 (AES) [8, 9], принятый в 2001 году и получивший наибольшее распространение в современных протоколах безопасности. Ключевое расписание БСШ AES представляет собой линейный массив 4-х байтовых слов. Первые элементы массива содержат мастер-ключ шифрования, остальные определяются рекурсивно посредством суммирования по модулю два предыдущих элементов. Для некоторых позиций массива дополнительно применяются шифроподобные преобразования, в частности, нелинейная подстановка блока данных, циклический сдвиг и пр. [8, 9]. В результате формируемая последовательность раундовых ключей $K_1^{(x)}, K_2^{(x)}, ..., K_t^{(x)}$ нелинейно зависит от исходного мастер-ключа $K^{(x)}$, что позволяет противостоять сдвиговым атакам на ключевое расписание [1].

Атаки на связанных ключах впервые предложены в [5] и получили дальнейшее развитие в [6, 7]. В частности, в работе [7] предложена первая криптоаналитическая атака на основе связанных ключей на полнораундовые шифры AES-192 и AES-256 (варианты FIPS-197 с длиной ключа 192 и 256 бит). Следует отметить, что приведенные в [7] атаки эффективнее полного перебора мастер-ключей, т.е. можно с уверенностью утверждать о реальном снижении стойкости стандартизированного криптоалгоритма.

Таким образом, атаки на ключевое расписание непрерывно совершенствуются и их возможное использование представляет реальную угрозу безопасности современным информационным системам [1-7]. Перспективные БСШ должны эффективно противостоять атакам на ключевое расписание, схема разворачивания не должна содержать уязвимостей, обусловленных простотой формирования и взаимной зависимостью цикловых ключей [11].

Фактически, речь идет о таком «идеальном» разворачивании раундовых ключей, при котором каждый элемент последовательности $K_1^{(x)}, K_2^{(x)}, ..., K_t^{(x)}$ формируется случайно, равновероятно и независимо от других цикловых ключей. Только в этом случае можно с уверенностью утверждать о бесполезности атак на ключевое расписание, т.к. каждый раунд БСШ параметризируется случайно выбранным значением и будет функционировать независимо от других итераций схемы шифрования (см. рис. 1).

В качестве яркой иллюстрации указанной тенденции в развитии схем ключевого расписания можно привести алгоритм «Калина», принятый в качестве национального стандарта БСШ Украины [12]. Он обладает усиленной схемой разворачивания цикловых ключей, основанной на использовании специальных однонаправленных функций. Цикловые ключи БСШ «Калина» формируются в результате нескольких раундов шифрования, параметризированных т.н. вспомогательным ключом. Вспомогательный ключ, в свою очередь, также формируется в результате нескольких раундов шифрования, параметризированных мастер-ключом. Другими словами, отдельные элементы последовательности цикловых ключей $K_1^{(x)}, K_2^{(x)}, ..., K_t^{(x)}$ формируются посредством независимого шифрования различных входных блоков данных на различных ключах. Если предположить, что применяемое шифрование реализует случайную подстановку (перестановку) блоков данных [13, 14], тогда полученные раундовые ключи будут сформированы случайно, равновероятно и независимо друг от друга [11, 15]. В частности, в работе [15] исследуются свойства ключевого расписания БСШ «Калина», утверждается о стойкости шифра к атакам на связанных ключах и к атакам на реализацию.

Следует отметить, что при случайном, равновероятном и независимом формировании раундовых ключей соответствующие последовательности могут совпадать. Это явление носит крайне негативный аспект, т.к. любое совпадение сформированных последовательностей цикловых ключей для различных мастер-ключей эквивалентно снижению мощности множества реализуемых отображений зашифрования-расшифрования, т.е. уменьшится реализуемое ключевое пространство. Даже совпадение отдельных раундовых ключей может привести к снижению криптографических свойств БСШ. Например, в режиме шифрования Galois/CounterMode & GMAC (спецификация которого приведена в стандарте NIST SP 800-38D [16]) т.н. субключ хеширования формируется посредством шифрования на мастер-ключе специальной константы (нулевой последовательности). Сформированный субключ является некоторым эквивалентом раундового ключа и его совпадение (в результате случайной подстановки, заданной шифрованием), приведет к тождественности формируемых контрольных сумм, т.е. создаст предпосылки для нарушения целостности – одной из базовых услуг безопасности.

Вопросы оценки соотношений мощностей множеств цикловых ключей и мастер-ключей шифрования (в рамках допущения про схему разворачивания как про случайное отображение) исследовались в [15]. Однако полученное выражение (27) в [15], как будет показано ниже, дает правильное решение только для последовательностей из одного раундового ключа и только для случая совпадения длин блоков данных и мастер-ключей. В общем случае полученные в [15] соотношения не верны.

Целью данной работы является исследование комбинаторных свойств ключевого расписания БСШ, при условии, что цикловые ключи формируются случайно, равновероятно и неза-

висимо друг от друга. Для абстрактного описания такого формирования используется модель случайной однородной подстановки. Таким образом, проводимые исследования выполняются в рамках допущения о т.н. «идеальном» ключевом расписании БСШ, т.е. полученные оценки следует использовать как верхние теоретические границы для вероятностных свойств последовательностей раундовых ключей. Практическая полезность результатов исследования состоит в их интерпретации для оценки свойств ключевого расписания принятого недавно национального стандарта БСШ Украины. Действительно, если верно принятое предположение, тогда полученные в данной работе выражения и конкретные оценки свойств ключевого расписания следует непосредственно отнести к БСШ «Калина». Результаты будут также полезны при оценке вероятностных свойств ключевых и других специальных данных различных режимов шифрования, в частности, субключей хеширования в режиме Galois/CounterMode& GMAC.

2. СЛУЧАЙНАЯ ПОДСТАНОВКА КАК МОДЕЛЬ ФОРМИРОВАНИЯ ЦИКЛОВЫХ КЛЮЧЕЙ

Рассмотрим определение и основные свойства случайной подстановки (перестановки) [13, 14], которые будем использовать в дальнейшем для оценки вероятностных свойств последовательностей раундовых ключей.

В комбинаторике под перестановкой понимают упорядоченный набор чисел 1, 2, ..., n, обычно трактуемый как биекция на множестве {1, 2, ..., n}, которая числу i ставит в соответствие i-й элемент из набора. Число n при этом называют порядком (степенью) перестановки [13, 14].

Под подстановкой s произвольного множества $Y=\{y_1, y_2, ..., y_n\}$ понимают некоторое правило, по которому каждому элементу y_i множества Y ставится в соответствие некоторый другой элемент $s(y_i)$ [13, 14]:

$$s=\begin{pmatrix} y_1 & y_2 & y_n \\ s(y_1) & s(y_2) & s(y_n) \end{pmatrix}.$$

В теории групп под подстановкой понимают биекцию этого множества на себя, т.е. подстановку S степени n рассматривают как перестановку элементов множества $Y = \{y_1, y_2, ..., y_n\}$ и для всех $i = 1, 2, ..., n$ соответствующие $s(y_i)\in Y$.

Значение функции $s(y_i)$ для конкретного элемента $y_i \in Y$ будем называть *реализацией подстановки* s в i-й точке.

Композиция подстановок s_u и s_w степени n определяется как последовательное выполнение перестановок элементов множества Y [13, 14]:

$$s_u \circ s_w=\begin{pmatrix} y_1 & y_2 & y_n \\ s_u(s_w(y_1)) & s_u(s_w(y_2)) & s_u(s_w(y_n)) \end{pmatrix},$$
$$s_u(s_w(y_i))\in Y .$$

Относительно операции последовательного выполнения подстановок множество всех $n!$ перестановок порядка n образует группу, которую называют симметрической группой и обозначают $s_n=\{s_1,s_2,...,s_{n!}\}$.

По определению [13, 14], *случайной подстановкой* (перестановкой) s_x называют случайный вектор $\{s_x(y_1),s_x(y_2),...,s_x(y_n)\}$, все элементы которого принимают значения из множества Y и при этом вероятность совпадения любых двух элементов равна 0. Другими словами, случайная подстановка – это случайно выбранная из множества s_n перестановка

$$s_x=\begin{pmatrix} y_1 & y_2 & ... & y_n \\ s_x(y_1) & s_x(y_2) & ... & s_x(y_n) \end{pmatrix},$$
$$s_x\in s_n,\ x\in\{1,2,...,n!\},$$

задаваемая набором (вектором) случайных значений $\{s_x(y_1),s_x(y_2),...,s_x(y_n)\}$, вероятности совпадения которых удовлетворяют условию:

$$\forall i\neq j\in\{1,2,...,n\}: P(s_x(y_i)=s_x(y_j))=0 .$$

Таким образом, под *реализацией случайной подстановки* s_x будем понимать конкретную реализацию случайного вектора $\{s_x(y_1),s_x(y_2),...,s_x(y_n)\}$, а соответствующее значение функции $s_x(y_i)$ будем называть *реализацией случайной подстановки* s_x в i-й точке.

Независимой случайной подстановкой называют такую случайную перестановку

$$\{s_x(y_1),s_x(y_2),...,s_x(y_n)\},$$

для которой

$$P(s_x)=\frac{P(s_x(y_1))P(s_x(y_2))...P(s_x(y_n))}{\sum_{u=1}^{n!}P(s_u(y_1))P(s_u(y_2))...P(s_u(y_n))}.$$

Если $\forall i,u\in\{1,2,...,n\}: P(s_u(y_i))=n^{-1}$, тогда

$$P(s_x)=\frac{P(s_x(y_1))P(s_x(y_2))...P(s_x(y_n))}{\sum_{u=1}^{n!}P(s_u(y_1))P(s_u(y_2))...P(s_u(y_n))}=$$
$$=\frac{n^{-n}}{\sum_{u=1}^{n!}n^{-n}}=\frac{1}{n!} \qquad (1)$$

и s_x называют *случайной, равновероятной и независимой* (или, сокращенно, *однородной*) *случайной подстановкой*.

Таким образом, понятие однородной случайной подстановки соответствует равномерному вероятностному распределению на множестве $s_n=\{s_1,s_2,...,s_{n!}\}$ с независимой реализацией случайных векторов

$$\{s_x(y_1),s_x(y_2),...,s_x(y_n)\}, \qquad (2)$$
$$s_x\in S_n,\ \forall x\in\{1,2,...,n!\}: P(s_x)=(n!)^{-1}.$$

Современные БСШ принято описывать моделью однородной случайной подстановки [1, 11], т.е. принимается предположение, что реализуемая функцией шифрования биекция обрабатываемых блоков данных по своим вероятностным свойствам удовлетворяет характеристикам случайной подстановки. Действительно,

если случайный, равновероятный и независимый выбор мастер-ключа $K^{(x)}$ связать с выбором подстановки $s_x \in s_n$, тогда полученное шифрующее преобразование будет соответствовать случайному, равновероятному и независимому сопоставлению блоков шифртекста блокам открытого текста на всех возможных вариантах биективных отображений, параметризированных ключом. Например, для l -битного шифра с длиной мастер-ключа k бит модель случайной подстановки будет состоять из подмножества

$$s'_n = \{s'_1, s'_2, ..., s'_{2^k}\} \subset s_n = \{s_1, s_2, ..., s_{n!}\}$$

случайно, равновероятно и независимо выбранных 2^k подстановок степени $n = 2^l$ (действующих на множестве $Y = \{y_1, y_2, ..., y_{2^l}\}$ блоков двоичных данных), причем выбор подстановки $s'_x \in s'_n \subset s_n$ (реализация случайного вектора $\{s_x(y_1), s_x(y_2), ..., s_x(y_{2^l})\}$) задается случайно, равновероятно и независимо выбранным значением k -битного мастер-ключа $K^{(x)}$.

Воспользуемся свойствами однородной случайной подстановки для исследования вероятностных характеристик последовательностей раундовых ключей. Для этого на множестве s'_n зададим равномерное вероятностное распределение:

$$\forall i \in \{1,2,...,2^l\}, \forall u \in \{1,2,...,2^k\} : P(s_u(y_i)) = 2^{-l},$$

т.е. все вероятности сопоставления i -го блока y_i u -му блоку $s_u(y_i)$ равны друг другу и не зависят ни от i, ни от u. Следовательно, вероятность случайного выбора подстановки $s'_x \in s'_n \subset s_n$ (и соответствующего ей мастер-ключа шифрования) не зависит от вида этой подстановки, а определяется как величина, обратная мощности множества мастер-ключей, что с учетом (1) примет вид:

$$P(s'_x) = \frac{P(s'_x(y_1))P(s'_x(y_2))...P(s'_x(y_{2^l}))}{\sum_{u=1}^{2^k} P(s'_u(y_1))P(s'_u(y_2))...P(s'_u(y_{2^l}))} = \frac{2^{-nl}}{\sum_{u=1}^{2^k} 2^{-nl}} = 2^{-k}. \quad (3)$$

Применим рассмотренную модель однородной случайной подстановки к исследованию вероятностных свойств элементов ключевого расписания. Оценим вероятности совпадений отдельных цикловых ключей, а также их последовательностей, предположив, что раундовые ключи формируются случайно, равновероятно и независимо друг от друга.

3. ВЕРОЯТНОСТНЫЕ СВОЙСТВА ЦИКЛОВЫХ КЛЮЧЕЙ

Введем следующие обозначения. Пусть на вход схемы разворачивания ключей (см. рис. 1) поступает случайно, равновероятно (с вероятностью 2^{-k}) и независимо сформированный мастер-ключ $K^{(x)}$ длины k бит. Обозначим последовательность из t формируемых раундовых ключей в виде $K_{\text{рк}}^{(x)} = \{K_1^{(x)}, K_2^{(x)}, ..., K_t^{(x)}\}$, где каждый $K_i^{(x)}$ является i-м цикловым ключом длины l-бит.

Предположим, что цикловые ключи $K_i^{(x)}$ являются независимыми реализациями однородной случайной подстановки в i-й точке:

$$\forall i \in \{1,2,...,t\} : K_i^{(x)} = s'_x(y_i),$$

т.е. формируются случайно, равновероятно и независимо друг от друга, причем для каждого $K_i^{(x)}$ конкретная реализация случайной подстановки $s'_x \in s'_n \subset s_n$ не зависит от $i \in \{1,2,...,t\}$.

Рассмотрим вероятностные свойства случайно формируемых значений раундовых ключей $K_i^{(x)}$ для некоторого фиксированного i, оценим среднее число различных значений цикловых ключей, которые могут быть сформированы при использовании всех 2^k значений мастер-ключей $K^{(x)}$.

Лемма. Среднее число различных l-битных цикловых ключей $K_i^{(x)}$, формируемых посредством 2^k реализаций случайной однородной подстановки, определяется выражением:

$$N(k,l) = 2^l(1-(1-2^{-l})^{2^k}) \approx 2^l\left(1-\left(\frac{1}{e}\right)^{2^{k-l}}\right) \approx 2^l\left(1-(0{,}37)^{2^{k-l}}\right). \quad (4)$$

Доказательство. Если схема формирования каждого циклового ключа из последовательности $K_{\text{рк}}^{(x)}$ описывается моделью случайной однородной подстановки с вероятностью (1) выбора $s'_x \in s'_n \subset s_n$ (по введенному мастер-ключу $K^{(x)}$), тогда, по определению, для любого фиксированного $i \in \{1,2,...,t\}$ вероятность появления $K_i^{(x)}$ не зависит ни от $y_i \in Y$, ни от $K^{(x)}$, а определяется как величина, обратная мощности множества Y, т. е. равна $P(s'_x(y_i) = K_i^{(x)}) = 2^{-l}$. Мастер-ключи $K^{(x)}$ выбираются независимо друг от друга и соответствующие события $s'_x(y_i) = K_i^{(x)}$ независимы. Следовательно, можем воспользоваться формулой для нахождения вероятности появления искомого события M раз в N испытаниях (формулой Бернулли):

$$P(N,M) = C_N^M(1-P(s'_x(y_i) = K_i^{(x)}))^{N-M}(P(s'_x(y_i) = K_i^{(x)}))^M = C_N^M(1-2^{-l})^{N-M}(2^{-l})^M.$$

Значение $P(N,M)$ задает вероятность того, что при N независимых реализациях случайной однородной подстановки в i -й точке конкретный раундовый ключ $K_i^{(x)} = s'_x(y_i)$ появится ровно M раз. Значение

$$P(2^k,0) = (1-2^{-l})^{2^k}$$

задает вероятность того, что при $N = 2^k$ независимых реализациях случайной подстановки в i -й точке (на всем множестве значений мастер-ключей $K^{(x)}$) раундовый ключ $K_i^{(x)} = s'_x(y_i)$ не появится ни разу.

Обратная величина

$$P(2^k,>0) = 1 - P(2^k,0) = \\ = \sum_{i=1}^{2^k} C_{2^k}^i (1-2^{-l})^{2^k-i} (2^{-l})^i = 1-(1-2^{-l})^{2^k} \quad (5)$$

задает вероятность такого события, когда при 2^k независимых испытаниях раундовый ключ $K_i^{(x)}$ длины l-бит будет сформирован хотя бы один раз.

Мощность множества различных l-битных значений равна 2^l, причем каждое из этих значений при 2^k независимых реализациях вектора (2) с вероятностью (5) появится в i -й точке случайной подстановки хотя бы один раз. То есть для 2^k различных мастер-ключей $K^{(x)}$, задающих реализацию вектора (2), схемой разворачивания будет сформировано в среднем

$$N(k,l) = 2^l P(2^k,>0) = 2^l (1-(1-2^{-l})^{2^k})$$

различных раундовых ключей $K_i^{(x)}$, что при подстановках $(1-2^{-l})^{2^l} \approx e^{-1}$ дает упрощенную формулу в правой части выражения (3), и, т.о., завершает доказательство.

Наиболее простой случай $k = l$ (равенство длин блоков шифртекста длине ключа) рассмотрен в докторской диссертационной работе Р.В. Олейникова, в которой получено «соотношение мощностей множества цикловых ключей и ключей шифрования» в виде рекуррентного выражения (27) в [15, с. 25]. Действительно, при $k = l$ вероятность(5),принимает вид

$$P(2^l,>0) = \sum_{i=1}^{2^l} C_{2^l}^i (1-2^{-l})^{2^l-i} (2^{-l})^i = 1 - P(2^l,0) = \\ = 1-(1-2^{-l})^{2^k} \approx 1 - e^{-1} \approx 0{,}63$$

и отношение среднего числа $N(k,l)$ различных раундовых ключей $K_i^{(x)}$ к числу 2^k различных мастер-ключей $K^{(x)}$ при $k = l$ определяется как

$$\delta(k,l) = \frac{N(k,l)}{2^k} = \frac{2^l(1-P(2^k,0))}{2^k} = \\ = P(2^l,>0) \approx 1 - e^{-1} \approx 0{,}63, \quad (6)$$

что соответствует формуле (27) в [15].

Следует, однако, отметить, что оценка среднего числа различных цикловых ключей в виде

$$N(k,l) = 2^l(1-P(2^k,0)) \approx 0{,}63 \cdot 2^l$$

справедлива только для случая $k = l$. При $k \neq l$ выражение (6), равно как и выражение (27) в [15, с. 25], выполняться не будет, отношение $\delta(k,l)$ необходимо оценивать по общей формуле

$$\delta(k,l) = \frac{N(k,l)}{2^k} = 2^{l-k}(1-(1-2^{-l})^{2^k}) \approx \\ \approx 2^{l-k}\left(1-\left(\frac{1}{e}\right)^{2^{k-l}}\right) \approx 2^{l-k}\left(1-(0{,}37)^{2^{k-l}}\right). \quad (7)$$

Приведем пример использования полученных соотношений. На рис. 2, 3 приведены, соответственно, зависимости вероятностей (5) и отношений (7) для длин блоков $0 \le l \le 16$ и ключей $0 \le k \le 16$. Очевидно, что даже для таких незначительных длин l и k, не превышающих 16 бит, наблюдается резкий переход от очень малых, практически равных нулю значений $P(2^k,>0)$ и $\delta(k,l)$, до очень больших, близких к единице. Это характерно, в особенности, для зависимости $P(2^k,>0)$, а множитель 2^{l-k} в (7) несколько «сглаживает» итоговую функцию (7), инвертируя качественный вид зависимости (5).

В табл. 1 сведены значения отношения среднего числа $N(k, l)$ различных цикловых ключей $K_i^{(x)}$ к числу 2^k различных мастер-ключей $K^{(x)}$ для наиболее распространенных в современных БСШ и их уменьшенных моделей значений l и K. Данные таблицы 1 рассчитаны по упрощенной формуле в правой части выражения (7) с использованием вычислительных алгоритмов системы WolframAlpha [17]. Приведенные значения демонстрируют конструктивность полученных аналитических выражений для оценки вероятностных характеристик раундовых ключей.

Для оценки вероятностных свойств последовательностей цикловых ключей обобщим положения леммы для случайно, равновероятно и независимо формируемых значений $K_1^{(x)}, K_2^{(x)}, ..., K_t^{(x)}$. Оценим среднее число различных последовательностей $K_{\text{рк}}^{(x)}$, формируемых при использовании всех 2^k значений мастер-ключей $K^{(x)}$. Справедлива след. теорема.

Теорема. Среднее число различных последовательностей цикловых ключей $K_{\text{рк}}^{(x)}$, формируемых посредством 2^k независимых реализаций случайной однородной подстановки в i-й точке, определяется:

$$N(k,l,t) = 2^l(1-(1-2^{-tl})^{2^k}) \approx 2^{tl}\left(1-\left(\frac{1}{e}\right)^{2^{k-tl}}\right) \approx \\ \approx 2^{tl}\left(1-(0{,}37)^{2^{k-tl}}\right). \quad (8)$$

Доказательство состоит в обобщении результатов леммы на случай формирования последовательностей $K_{\text{рк}}^{(x)}$. Действительно, в соответствии с принятым допущением о формировании цикловых ключей $K_i^{(x)}$ независимыми реализациями однородной случайной подстановки в i -й точке, для каждого $i \in \{1,2,...,t\}$ вероятность появления $K_i^{(x)}$ не зависит ни от $y_i \in Y$, ни от $K^{(x)}$. Эта вероятность равна $P(s'_x(y_i) = K_i^{(x)}) = 2^{-l}$. Совместная вероятность появления независимых событий равна произведению вероятностей этих событий, т.е:

$$P(K_{\text{рк}}^{(x)} = \{K_1^{(x)}, K_2^{(x)}, ..., K_t^{(x)}\}) = \\ = \prod_{i=1}^{t} P(s'_x(y_i) = K_i^{(x)}) = 2^{-tl}.$$

Мастер-ключи $K^{(x)}$ выбираются независимо друг от друга и соответствующие события $K_{\text{рк}}^{(x)}$ независимы. Следовательно, воспользовавшись формулой Бернулли, получим выражение

$$P(N,M,t) = C_N^M (1-2^{-tl})^{N-M} (2^{-tl})^M ,$$

которое задает вероятность того, что в N независимых реализациях последовательность $K_{\text{рк}}^{(x)}$ появится ровно M раз. Величина

$$P(2^k,>0,t) = \sum_{i=1}^{2^k} C_{2^k}^i (1-2^{-tl})^{2^k-i} (2^{-tl})^i = \\ = 1 - P(2^k,0,t) = 1-(1-2^{-tl})^{2^k} \quad (9)$$

задает вероятность того, что при 2^k независимых испытаниях конкретная последовательность $K_{\text{рк}}^{(x)}$ будет сформирована хотя бы один раз.

Мощность множества различных t-последовательностей l-битных значений равна 2^{tl}, причем каждая из этих последовательностей с вероятностью (9) появится на выходе схемы разворачивания раундовых ключей хотя бы один раз. То есть для 2^k различных мастер-ключей $K^{(x)}$, задающих реализации однородной случайной подстановки, схемой разворачивания будет сформировано в среднем $N(k,l,t) = 2^{tl}(1-P(2^k,0,t))$ различных $K_{\text{рк}}^{(x)}$, что при упрощении $(1-2^{-l})^{2^l} \approx e^{-1}$ дает формулу (8).

Доказанная теорема позволяет получить выражение для оценки отношения среднего числа различных последовательностей $K_{\text{рк}}^{(x)}$ к мощности множества различных мастер-ключей $K^{(x)}$:

$$\delta(k,l,t) = \frac{N(k,l,t)}{2^k} = 2^{tl-k}(1-(1-2^{-tl})^{2^k}) \approx \\ \approx 2^{tl-k}\left(1-\left(\frac{1}{e}\right)^{2^{k-tl}}\right) \approx 2^{tl-k}\left(1-(0{,}37)^{2^{k-tl}}\right). \quad (10)$$

Формула (10) отличается от известного выражения (27) в [15]. Результаты моделирования показывают, что оценки отношения $\delta(k,l,t)$, приведенные в работе [15], в общем случае не верны. Напротив, оценка (10), как будет показано ниже, подтверждается результатами моделирования.

Для удобства проведения расчетов соотношения $\delta(k,l,t)$ запишем формулу (10) иначе. В большинстве практически важных случаев блочного шифрования (например, для БСШ «Калина») длина мастер-ключа k кратна длине блока l, т.е. выполняется соотношение $k = ml$, что после подстановки в (10) дает форму:

$$\delta(ml,l,t) = 2^{l(t-m)}(1-(1-2^{-tl})^{2^{ml}}) \approx \\ \approx 2^{l(t-m)}\left(1-\left(\frac{1}{e}\right)^{2^{l(m-t)}}\right) \approx 2^{l(t-m)}\left(1-(0{,}37)^{2^{l(m-t)}}\right). \quad (11)$$

Выражение (11) показывает, что увеличение кратности m эквивалентно, в вероятностном смысле, соответствующему уменьшению длины t последовательности $K_{\text{рк}}^{(x)}$. И наоборот: увеличение длины t последовательности раундовых ключей уменьшает вероятность (9) также, как и кратное снижение длины мастер-ключа. Характерной демонстрацией подобного эффекта является симметричность относительно значений l и k графиков функций, приведенных на рис. 2, 3. В этом смысле, расчетные значения $\delta(ml,l,t)$ для случая $l \in \{16, 32\}$ и $m, t \in \{1, 2, 4, 8, 16\}$ могут быть полностью получены из данных таблицы 1 при выборе столбцов, с обозначениями ml, и строк, с обозначениями tl.

В качестве примера в табл. 2 приведены расчетные значения $\delta(ml,l,t)$ для $l = 32$, которые, очевидно, полностью сходятся с данными,

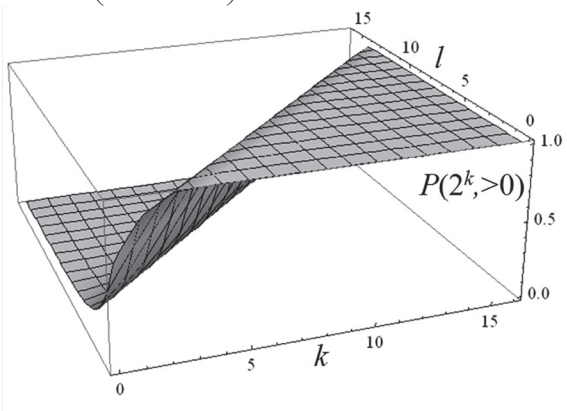


Рис. 2

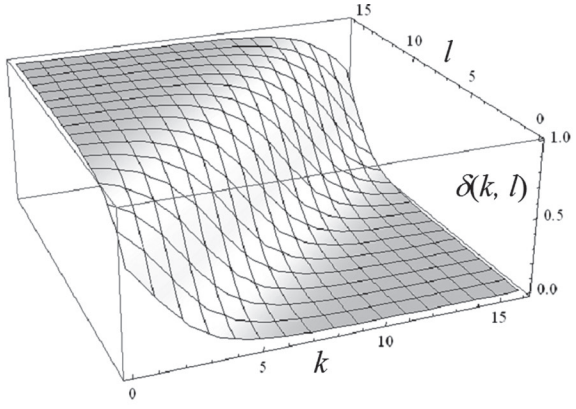


Рис. 3

Таблица 1

Отношение среднего числа различных l-битных цикловых ключей $K_i^{(x)}$, формируемых посредством $2k$ реализаций случайной однородной подстановки, к мощности множества различных мастер-ключей $K^{(x)}$

	K=16	K=32	K=64	K=128	K=256	K=512
l = 16	0,63	$1{,}52\cdot10^{-5}$	$3{,}55\cdot10^{-15}$	$1{,}93\cdot10^{-34}$	$5{,}66\cdot10^{-73}$	$4{,}89\cdot10^{-150}$
l = 32	$1-7{,}63\cdot10^{-6}$	0,63	$2{,}33\cdot10^{-10}$	$1{,}26\cdot10^{-29}$	$3{,}71\cdot10^{-68}$	$3{,}20\cdot10^{-145}$
l = 64	$1-1{,}78\cdot10^{-15}$	$1-1{,}16\cdot10^{-10}$	0,63	$5{,}42\cdot10^{-20}$	$1{,}59\cdot10^{-58}$	$1{,}38\cdot10^{-135}$
l = 128	$1-9{,}63\cdot10^{-35}$	$1-6{,}31\cdot10^{-30}$	$1-2{,}71\cdot10^{-20}$	0,63	$2{,}94\cdot10^{-39}$	$2{,}54\cdot10^{-116}$
l = 256	$1-2{,}83\cdot10^{-73}$	$1-1{,}85\cdot10^{-68}$	$1-7{,}97\cdot10^{-59}$	$1-1{,}47\cdot10^{-39}$	0,63	$8{,}64\cdot10^{-78}$
l = 512	$1-2{,}44\cdot10^{-150}$	$1-1{,}60\cdot10^{-145}$	$1-6{,}89\cdot10^{-136}$	$1-1{,}27\cdot10^{-116}$	$1-4{,}32\cdot10^{-78}$	0,63

приведенными в таблице 1. Расчетные значения $\delta(ml,l,t)$ для случаев $l\in\{64, 128, 256\}$, $m\in\{1, 2, 4, 8\}$ и $t\in\{1, 2, 4, 8, 16\}$ сведены в таблицу 3. Приведенные расчетные значения улучшают данные по оценке $\delta(k,l,t)$ в [15], т.к. основаны на выражениях (8)–(11), учитывающих соотношения длин блоков и мастер-ключей. Вывод о практическом совпадении мощностей последовательностей раундовых ключей и мастер-ключей шифрования в [15] следует считать верным, приведенные в таблице 3 данные подтверждают эту закономерность.

Для всех рассмотренных и практически значимых соотношений l и k при выполнении неравенства $t > m$ отношение среднего числа различных последовательностей раундовых ключей к мощности множества различных мастер-ключей лишь незначительно отличается от единицы. При дальнейшем повышении длины последовательности раундовых ключей t это расхождение стремительно убывает.

Для подтверждения адекватности и достоверности полученных результатов и сделанных по ним выводов проведен численный эксперимент, суть которого состояла в подсчете отношения среднего числа различных последовательностей раундовых ключей к мощности 2^k множества различных мастер-ключей. Для моделирования случайной подстановки использовалась простейшая функция генерации случайных чисел, интегрированная в среду быстрой разработки приложений Embarcadero RAD Studio [18]. Каждое наблюдение состояло в оценке выборочного (эмпирического) среднего по 100 реализациям (прогонам) модели. Каждая реализация модели состояла в подсчете отношения среднего числа различных последовательностей раундовых ключей к мощности 2^k множества различных мастер-ключей. При проведении эксперимента оценивались как выборочные средние $\delta^*(ml,l,t)$, так и выборочные дисперсии D при объеме выборки, состоящей из 100 элементов. Полученные результаты

Таблица 2

Отношение среднего числа различных последовательностей цикловых ключей к мощности множества различных мастер-ключей длины ($l = 32$)

	m=1	m=2	m=4	m=8	m=16
t=1	0,63	$2{,}33\cdot10^{-10}$	$1{,}26\cdot10^{-29}$	$3{,}71\cdot10^{-68}$	$3{,}20\cdot10^{-145}$
t=2	$1-1{,}16\cdot10^{-10}$	0,63	$5{,}42\cdot10^{-20}$	$1{,}59\cdot10^{-58}$	$1{,}38\cdot10^{-135}$
t=4	$1-6{,}31\cdot10^{-30}$	$1-2{,}71\cdot10^{-20}$	0,63	$2{,}94\cdot10^{-39}$	$2{,}54\cdot10^{-116}$
t=8	$1-1{,}85\cdot10^{-68}$	$1-7{,}97\cdot10^{-59}$	$1-1{,}47\cdot10^{-39}$	0,63	$8{,}64\cdot10^{-78}$
t=16	$1-1{,}60\cdot10^{-145}$	$1-6{,}89\cdot10^{-136}$	$1-1{,}27\cdot10^{-116}$	$1-4{,}32\cdot10^{-78}$	0,63

Таблица 3

Отношение среднего числа различных последовательностей раундовых ключей к мощности множества различных мастер-ключей

$l = 64$				
	$m = 1$	$m = 2$	$m = 4$	$m = 8$
$t = 1$	0,63	$1{,}52\cdot10^{-5}$	$5{,}66\cdot10^{-73}$	$4{,}89\cdot10^{-150}$
$t = 2$	$1-7{,}62\cdot10^{-6}$	0,63	$3{,}71\cdot10^{-68}$	$3{,}20\cdot10^{-145}$
$t = 4$	$1-1{,}78\cdot10^{-15}$	$1-1{,}16\cdot10^{-10}$	0,63	$8{,}64\cdot10^{-78}$
$t = 8$	$1-2{,}44\cdot10^{-150}$	$1-1{,}60\cdot10^{-145}$	$1-1{,}27\cdot10^{-116}$	0,63
t=16	$1-5{,}13\cdot10^{-290}$	$1-9{,}46\cdot10^{-271}$	$1-3{,}22\cdot10^{-232}$	$1-3{,}73\cdot10^{-155}$
$l = 128$				
$t = 1$	0,63	$2{,}94\cdot10^{-39}$	$2{,}54\cdot10^{-116}$	$1{,}89\cdot10^{-270}$
$t = 2$	$1-1{,}47\cdot10^{-39}$	0,63	$8{,}64\cdot10^{-78}$	$6{,}44\cdot10^{-232}$
$t = 4$	$1-1{,}27\cdot10^{-116}$	$1-4{,}32\cdot10^{-78}$	0,63	$7{,}45\cdot10^{-155}$
$t = 8$	$1-9{,}46\cdot10^{-271}$	$1-3{,}22\cdot10^{-232}$	$1-3{,}74\cdot10^{-155}$	0,63
t=16	$1-5{,}26\cdot10^{-579}$	$1-1{,}79\cdot10^{-540}$	$1-2{,}07\cdot10^{-463}$	$1-2{,}78\cdot10^{-309}$
$l = 256$				
$t = 1$	0,63	$8{,}64\cdot10^{-78}$	$6{,}44\cdot10^{-232}$	$3{,}58\cdot10^{-540}$
$t = 2$	$1-4{,}32\cdot10^{-78}$	0,63	$7{,}45\cdot10^{-155}$	$4{,}15\cdot10^{-463}$
$t = 4$	$1-3{,}22\cdot10^{-232}$	$1-3{,}73\cdot10^{-155}$	0,63	$5{,}56\cdot10^{-309}$
$t = 8$	$1-1{,}79\cdot10^{-540}$	$1-2{,}08\cdot10^{-463}$	$1-2{,}78\cdot10^{-309}$	0,63
t=16	$1-5{,}54\cdot10^{-1157}$	$1-6{,}42\cdot10^{-1080}$	$1-8{,}61\cdot10^{-926}$	$1-1{,}55\cdot10^{-617}$
$l = 512$				
$t = 1$	0,63	$7{,}46\cdot10^{-155}$	$4{,}15\cdot10^{-463}$	$1{,}28\cdot10^{-1079}$
$t = 2$	$1-3{,}73\cdot10^{-155}$	0,63	$5{,}56\cdot10^{-309}$	$1{,}72\cdot10^{-925}$
$t = 4$	$1-2{,}08\cdot10^{-463}$	$1-2{,}78\cdot10^{-309}$	0,63	$3{,}09\cdot10^{-617}$
$t = 8$	$1-6{,}42\cdot10^{-1080}$	$1-8{,}60\cdot10^{-926}$	$1-1{,}55\cdot10^{-617}$	0,63
t=16	≈1	≈1	≈1	$1-4{,}79\cdot10^{-1237}$

сведены в табл. 4. В последнем столбце таблицы приведены значения точности ε оцениваемой характеристики при заданном уровне значимости $\alpha = 0{,}05$.

Таблица 4

Результаты экспериментальных исследований и их сравнение с теоретическими расчетами

$l=4, m=1$				
	$\delta(ml,l,t)$	$\delta^*(ml,l,t)$	D	ε
t=1	0,632121	0,6453125	0,005716	0,014818
$t=2$	0,969391	0,969125	0,001758	0,008218
$t=3$	0,998049	0,9983125	0,000110	0,002056
$t=4$	0,999878	0,999875	0,000008	0,000554
$l=4, m=2$				
$t=1$	0,062500	0,062500	0	0
$t=2$	0,632121	0,633074	0,000372	0,003780
$t=3$	0,969391	0,969675	0,000103	0,001989
$t=4$	0,998049	0,997934	0,000008	0,000554
$l=4, m=3$				
$t=1$	0,003906	0,003906	0	0
$t=2$	0,062500	0,062500	0	0
$t=3$	0,632121	0,632152	0,000023	0,000940
$t=4$	0,969391	0,969492	0,000007	0,000519
$l=8, m=1$				
$t=1$	0,632121	0,632836	0,000360	0,003719
$t=2$	0,998049	0,998051	0,000007	0,000519
$t=3$	0,999992	0,999961	0,000002	0,000277
$l=8, m=2$				
$t=1$	0,003906	0,003906	0	0
$t=2$	0,632121	0,632176	0,000002	0,000277
$t=3$	0,998049	0,998063	$2{,}98\cdot10^{-8}$	$3{,}38\cdot10^{-5}$

Как видно из приведенных в табл. 4 значений, результаты экспериментальных исследований полностью подтверждают достоверность полученных теоретических положений. Для всех случаев расчетные значения $\delta(ml,l,t)$ и полученные эмпирические данные $\delta^*(ml,l,t)$ отличаются не более, чем на ε (абсолютное значение ошибки), причем вероятность, с которой достигается указанная точность (достоверность оценки) составляет 0,95. Поскольку достоверность характеризует повторяемость, устойчивость эксперимента [19], можно утверждать, что в 95% случаев величина $\delta^*(ml,l,t)$ будет отличаться от $\delta(ml,l,t)$ меньше, чем на ε.

4. ВЫВОДЫ И ПЕРСПЕКТИВЫ ДАЛЬНЕЙШИХ ИССЛЕДОВАНИЙ

Проведенные исследования вероятностных свойств раундовых ключей БСШ показали, что даже при случайном, равновероятном и независимом формировании используемые ключевые последовательности могут совпадать, что неизбежно снижает мощность множества реализуемых отображений зашифрования-расшифрования.

Для описания схемы разворачивания раундовых ключей использовалась абстрактная модель случайной подстановки, параметризированная значением мастер-ключа шифрования. Полученные с ее помощью аналитические соотношения позволяют оценить вероятностные свойства цикловых ключей БСШ. В частности, вероятность кратного совпадения раундовых ключей при заданном числе реализаций случайной однородной подстановки (заданном числе мастер-ключей) определяется по формуле Бернулли. Это соотношение дает оценку вероятности того события, когда на всем множестве мастер-ключей конкретный раундовый ключ будет сформирован хотя бы один раз, т.е. позволяет оценить среднее число различных раундовых ключей на выходе схемы формирования. Последний результат обобщается также на последовательности раундовых ключей произвольной длины, т.е. в рамках принятой модели удается получить численные оценки вероятностных свойств всех элементов ключевого расписания БСШ.

Приведенные в таблицах 2, 3 расчетные значения отношения среднего числа различных последовательностей раундовых ключей к мощности множества различных мастер-ключей дают представление о реализации шифром всего множества допустимых отображений зашифрования-расшифрования. В частности, приведенные расчетные значения для наиболее важных в практическом отношении случаев для длин блоков данных l и ключей $k = ml$ показывают, что при числе раундов $t < m$ с вероятностью, близкой к единице, конкретная последовательность раундовых ключей на всем множестве мастер-ключей сформирована не будет. Это равносильно тому, что среднее число различных последовательностей раундовых ключей будет пренебрежительно мало по сравнению с мощностью множества различных мастер-ключей, т.е. подавляющее число отображений «открытый текст – шифртекст» из всего множества 2^k отображений *реализованы не будут*. Напротив, для случая $t > m$, отношение среднего числа различных последовательностей раундовых ключей к мощности множества мастер-ключей практически не отличается от единицы. При дальнейшем повышении t это расхождение стремительно убывает и, следует полагать, что при таком ключевом расписании *будут реализованы* все допустимые отображения «открытый текст – шифртекст» из полного множества 2^k отображений.

Проведенное имитационное моделирование «идеальной» схемы разворачивания ключей позволило получить эмпирические оценки, которые совпадают с теоретическими расчетами по формулам (9) – (11), что подтверждает достоверность и обоснованность результатов исследования. В частности, для всех исследуемых случаев расчетные значения и полученные эмпирические данные отличаются незначительно (относительное значение ошибки < 3%), при-

чем вероятность, с которой указанная точность достигается (достоверность оценки) составляет 0,95. Следовательно, можно утверждать, что в 95% случаев расчетные и эмпирические данные отличаются меньше, чем на величину ошибки.

Следует отметить, что полученные аналитические выражения и приведенные расчетные значения соответствуют гипотетическому случаю случайного, равновероятного и независимого формирования раундовых ключей, т.е. «идеальному» с вероятностной точки зрения ключевому расписанию. Реальные схемы разворачивания ключей основаны на детерминированных алгоритмах, параметризованных значением мастер-ключа шифрования. Следовательно, полученные оценки на модели случайной однородной подстановки следует использовать как *верхние теоретические границы* для вероятностных свойств последовательностей раундовых ключей: ключевое расписание реального БСШ может лишь приближаться по своим характеристикам к этому «идеальному» случаю и не будет улучшать приведенные расчетные значения.

Практическая полезность полученных результатов состоит в их непосредственной интерпретации для *оценки вероятностных свойств элементов ключевого расписания нового национального стандарта БСШ Украины* [12]. Если верно предположение о том, что цикловые ключи БСШ «Калина» являются независимыми реализациями однородной случайной подстановки, т.е. формируются случайно, равновероятно и независимо друг от друга, тогда вывод о практическом совпадении мощностей последовательностей раундовых ключей и мастер-ключей шифрования следует считать теоретически доказанным, приведенные в таблице 3 расчетные значения наглядно подтверждают эту закономерность. Использование в новом стандарте «идеального» ключевого расписания помимо обеспечения стойкости шифра к атакам на связанных ключах и атакам на реализацию [15] позволяет полностью реализовать ключевое пространство мастер-ключей и соответствующее множество отображений «открытый текст - шифртекст».

Применение полученных теоретических результатов к оценке вероятностных свойств субключей хеширования в режиме Galois/Counter Mode & GMAC [16] показывает, что ключевое пространство мастер-ключей шифрования в этом режиме реализуется всего на 63%. Возможное совпадение формируемых субключей приведет к тождественности криптографических контрольных сумм, что создает реальные предпосылки для нарушения целостности информации. Изучение вероятностных свойств субключей хеширования в режиме Galois/Counter Mode & GMAC, оценка числа эквивалентных и слабых мастер-ключей является перспективным направлением дальнейших исследований.

В качестве *перспективного направления* дальнейших исследований следует также указать поиск или, по крайней мере, оценку вероятностных свойств подмножеств т.н. *эквивалентных ключей* БСШ, когда несколько различных по значению мастер-ключей приводят к формированию одинаковых последовательностей цикловых ключей, тождественно задающих биективные отображения зашифрования – расшифрования. Другими словами, наличие подмножеств эквивалентных ключей уменьшает мощность множества отображений «открытый текст – шифртекст», количество ключевой информации, численно равное мере неопределенности секретных параметров шифрования, также уменьшается. Кроме того, само существование отличных по значению, но эквивалентных по задаваемой функции шифрования нескольких мастер-ключей может быть использовано злоумышленником для реализации криптоаналитических атак, например, основанных на подмене защищаемой информации ложными данными в случае использования БСШ в режиме выработки имитовставок.

Литература

[1] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Теорія. Практика. Застосування: Підручник для вищих навч. закладів. – Харків: Вид-во «Форт», 2013. – 880 с.

[2] *Biryukov A., Wagner D.* Slide Attacks (англ.) //Fast Software Encryption. 6th International Workshop, FSE'99 Rome, Italy, March 24–26, 1999 Proceedings. - Springer Berlin Heidelberg, 1999. - С. 245-259.

[3] *Chalermpong Worawannotai, Isabelle Stanton* A Tutorial on Slide Attacks (англ.). [Електронний ресурс] – Режим доступа: http://www.eecs.berkeley.edu/~isabelle/slideattacks.pdf.

[4] *Biryukov A., Wagner D.* Advanced Slide Attacks (англ.) // Advances in Cryptology - EUROCRYPT 2000. International Conference on the Theory and Application of Cryptographic Techniques Bruges, Belgium, May 14-18, 2000 Proceedings. – Springer Berlin Heidelberg, 2000. – С. 589–606

[5] *Biham E.* New types of cryptanalytic attacks using related keys (англ.) // Springer-Verlag : журнал. – 1994. – № 4. – С. 229–246.

[6] *Ciet M., Piret G., Quisquater J.-J.* Related-Key and Slide Attacks: Analysis, Connections, and Improvements (Extended Abstract). // http://citeseer.ist.psu.edu – 2002 – Universite catholique de Louvain, Louvain-la-Neuve, Belgium

[7] *Biryukov A., Khovratovich D.* Related-Key Cryptanalysis of the Full AES-192 and AES-256 (англ.) // Springer Berlin Heidelberg : журнал. – 2009. – С. 1–18.

[8] *Daemen J.* AES proposal: Rijndael [Електронний ресурс] / J. Daemen, V. Rijmen // 1998. Режим доступа: http://www.nist.gov/aes.

[9] FIPS-197: Advanced Encryption Standard (AES) [Електронний ресурс] // National Institute of Standards and Technology. – 2001. – Режим доступа: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[10] Положення про порядок здійснення криптографічного захисту інформації в Україні, затверджене Указом Президента України від 22 травня 1998 року N 505/98.

[11] Розробка нового блокового симетричного шифру [Текст]: звіт за перший етап НДР «Алгоритм» (проміжний) / кер. І. Д. Горбенко; АТ «ІІТ». – Харків, 2014. – Том 4. – 304 с.

[12] Інформаційні технології. Криптографічний захист інфомації. Алгоритм симетричного блокового перетворення: ДСТУ 7624:2014. – К.: Мінекономрозвитку України, 2015. – 238 с.

[13] *Сачков В.Н.* Введение в комбинаторные методы дискретной математики. – М.: Наука. Гл. ред. физ.-мат. лит., 1982. – 384 с.

[14] *Сачков В.Н.* Вероятностные методы в комбинаторном анализе. – М., Наука, 1978 г. – 287 с.

[15] Олійников Р. В. Методи аналізу і синтезу перспективних симетричних криптографічних перетворень: автореф. дис. ... д-ра техн. наук : 05.13.05 / Р. В. Олійников; Харків. нац. ун-т радіоелектрон. – Харків, 2014. – 42 с.

[16] NIST Special Publication 800-38D. Block Cipher Modes. [Електронний ресурс]. – Режим доступа: http://csrc.nist.gov/groups/ST/toolkit/BCM/current_modes.html

[17] Вопросно-ответная система WolframAlpha. [Электронный ресурс] – Режим доступа: http://www.wolframalpha.com/

[18] Интегрированная среда разработки Embarcadero RAD Studio. [Электронный ресурс] – Режим доступа: http://www.embarcadero.com/ products/rad-studio

[19] *Вентцель Е.С., Овчаров Л.А.* Теория вероятностей и ее инженерные приложения. Учеб. пособие для втузов.– 2-е изд., стер. – М.: Высш. шк., 2000. – 480 с.

Поступила в редколлегию 24.11.2015

Кузнецов Александр Александрович, фото и сведения об авторе см. на с. 334.

Горбенко Юрий Иванович, фото и сведения об авторе см. на с. 290.

Иваненко Дмитрий Викторович, фото и сведения об авторе см. на с. 334.

УДК 004.056.55

Комбінаторні властивості ключового розкладу блокових симетричних шифрів / О.О. Кузнєцов, Ю.І. Горбенко, Д.В. Іваненко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 351–360.

Досліджуються комбінаторні властивості ключового розкладу блокових симетричних шифрів у припущенні, що циклічні (раундові) ключі формуються випадково, рівномірно та незалежно один від одного. Для абстрактного опису такого формування використовується модель випадкової однорідної підстановки, отримані аналітичні вирази дозволяють оцінити потужність множини, яка реалізує відображення зашифрування-розшифрування, отримати оцінки ймовірнісних властивостей послідовностей раундових ключів і відношень середнього числа різних ключових послідовностей до потужності множини різних майстер-ключів. Результати імітаційного моделювання підтверджують достовірність й обгрунтованість отриманих аналітичних виразів.

Ключові слова: ключовий розклад, циклічні ключі, комбінаторні властивості, блокові симетричні шифри.

Табл.: 4. Іл.: 3. Бібліогр.: 19 найм.

UDK 004.056.55

Combinatorial properties of the key schedule of block symmetric ciphers / A.A. Kuznetsov, Yu.I. Gorbenko, D.V. Ivanenko // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 351–360.

The paper studies the combinatorial properties of the key schedule of block symmetric ciphers on the assumption that the cyclic (round) keys are generated randomly, with equal probability and independently. To abstractly describe such a formation a homogeneous random substitution model is used. The analytical expressions obtained allow to estimate the cardinality of a set of implemented encryption-decryption maps, to obtain estimates of the probability properties of sequences of round keys and relationships of the average number of different key sequences to the power of many different master keys. Simulation results have confirmed the reliability and validity of the obtained analytical expressions.

Keywords: key schedule, cyclic keys, combinatorial properties, block symmetric ciphers.

Tab.: 4. Fig.: 3. Ref.: 19 items.

УДК 681.3.06

УНИВЕРСАЛЬНОЕ ХЕШИРОВАНИЕ ПО КРИВЫМ, АССОЦИИРОВАННЫМ С ГРУППОЙ СУДЗУКИ

Е.В. КОТУХ, Г.З. ХАЛИМОВ

Представлены результаты универсального хеширования по кривым, ассоциированным с группой Судзуки над расширениями конечного поля. Рассмотрены проективные многообразия точек кривых, поля рациональных функций. Получены сравнительные оценки вероятности коллизии универсального хеширования. Из оценки следует, что наилучший результат достигается на кривой Судзуки над полем F_q с параметрами $q = 2q_0^2$ и $q_0 = 2^s$.

Ключевые слова: универсальное хеширование, группа Судзуки, кривые Судзуки.

ВВЕДЕНИЕ

Группа Судзуки определяет многообразие алгебраических кривых, которые ассоциируются с её подгруппами. Интенсивные исследования таких кривых лежат в плоскости практических применений: схем универсального хеширования, алгебраических кодов. Основное преимущество вычислений по кривым Судзуки определяется простотой задания их точек, что нельзя сказать, например, о кривых Эрмита и других максимальных кривых. Известно, что наилучший результат универсального хеширования достигается на кривых с большим числом точек, достаточно близких к границе Хассе-Вейля. Проблематикой универсального хеширования на функциональном поле $F_q(C)$ алгебраической кривой C над конечным полем F_q является определение проективного многообразия точек кривой, поля рациональных функций и оценка параметров хеш функций.

Целью статьи является построение универсального хеширования по функциональному полю кривых, ассоциированных с подгруппами группы Судзуки. В разделе 1 приводятся определения и свойства группы Судзуки. В разделе 2 рассмотрены оценки параметров универсального хеширования по кривым Судзуки.

1. ОПРЕДЕЛЕНИЕ И СВОЙСТВА ГРУППЫ СУДЗУКИ

Семейство исключительных групп известных, как группы Судзуки, впервые представлены в [1,2]. Определения и свойства группы Судзуки $Sz(q)$ изложены в [3]. Группу $Sz(q)$ не следует путать с Судзуки 2-группой или спорадической группой Судзуки.

Замечание 1.

1. Существуют различные концептуальные определения группы Судзуки [4,5]. Явное представление группы, на основе матриц 4x4 над полями характеристики $p = 2$ дано в оригинальной работе Судзуки [1]. Параметризация группы выполнена в работах Хуберта [3] и Джонса [6 15].

Определение 1 (группа Судзуки). Пусть $a,b,\alpha,\beta \in F_q$, $q = 2^{2n+1}$ и $c,\gamma \in F_q^\times$, где $F_q^\times$ – мультипликативная группа. Определим 4×4 матрицы над F_q в виде

$$u(a,b,\alpha,\beta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \alpha & 1 & 0 & 0 \\ \alpha a+\beta & a & 1 & 0 \\ \alpha^2 a+\alpha\beta+b & \beta & \alpha & 1 \end{pmatrix},$$

$$d(c,\gamma) = \begin{pmatrix} c\gamma & 0 & 0 & 0 \\ 0 & \gamma & 0 & 0 \\ 0 & 0 & \gamma^{-1} & 0 \\ 0 & 0 & 0 & \gamma^{-1}c^{-1} \end{pmatrix},$$

$$T = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix},$$

Пусть $U(\alpha,\beta) = u(\alpha^\theta,\beta^\theta,\alpha,\beta)$ и $D(\gamma) = d(\gamma^\theta,\gamma)$, $\theta = 2^{n+1}$. Группа Судзуки имеет представление

$$Sz(q) = \begin{Bmatrix} U(\alpha,\beta)D(\gamma)TU(\alpha',\beta'): \\ \alpha,\alpha',\beta,\beta' \in F_q, \gamma \in F_q^\times \end{Bmatrix}.$$
$$\cup\left\{U(\alpha,\beta)D(\gamma): \alpha,\beta \in F_q, \gamma \in F_q^\times\right\}$$

Замечание 4.

1. Параметризация группы Судзуки по определению 1 впервые представлена в [6] и взята из работы [7]. Группа Судзуки следует из выражения для $Sz(q)$ и определяется множествами матриц 4×4 над конечным полем F_q, $q = 2^{2n+1}$ характеристики 2.

2. Свойства группы Судзуки достаточно полно исследованы в [3] с несколько отличным представлением матричных групп. Следуя работе [3] рассмотрим следующие определения.

Определение 2. Пусть $\pi(x) = x^t$, $t = 2^{m+1}$ для $\forall x \in F_q$, $q = 2^{2m+1}$, π – исключительный автоморфизм в F_q такой, что $\pi(\pi(x)) = x^2$. Для $a,b \in F_q$ и $c \in F_q^\times$ определим матрицы

$$S(a,b) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ a & 1 & 0 & 0 \\ b & \pi(a) & 1 & 0 \\ a^2\pi(a)+ab+\pi(b) & a\pi(a)+b & a & 1 \end{pmatrix},$$

$$M(c)=\begin{pmatrix} c^{1+2^m} & 0 & 0 & 0 \\ 0 & c^{2^m} & 0 & 0 \\ 0 & 0 & c^{-2^m} & 0 \\ 0 & 0 & 0 & c^{-1-2^m} \end{pmatrix},$$

$$T=\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

Замечание 2.

1. Матрицы $S(a,b)$ и $M(c)$ следуют из переопределений матриц $U(\alpha,\beta)$ и $D(\gamma)$ с заменой переменных $\alpha\to a$, $\alpha a+\beta\to b$, $\gamma\to c^{2^m}$.

Определение 3. Группа Судзуки в соответствии с выражением (1) определяется произведением матриц

$$Sz(q)=\left\langle S(a,b),M(c),T\middle|a,b\in F_q,c\in F_q^{\times}\right\rangle,$$

имеет порядок

$$\left|Sz(q)\right|=q^2(q^2+1)(q-1),$$

где три делителя попарно взаимно простые и

$$(q^2+1)=(q+t+1)(q-t+1).$$

Предложение 1 [3]**.** Для всех $a,b,a',b'\in F_q$ и $c\in F_q^{\times}$ справедливо

$$S(a,b)S(a',b')=S(a+a',b+b'+\pi(a)a'),$$

$$S(a,b)^{M(c)}=S(ca,c\pi(c)b),$$

$$M(c)M(c')=M(cc').$$

Соотношения проверяются прямыми вычислениями и следуют из свойств автоморфизма

$$\pi(\pi(a))=a^2,\ \pi(ab)=\pi(a)\pi(b)$$

$$\text{и } \pi(a+b)=\pi(a)+\pi(b),\ a,b\in F_q$$

в поле характеристики 2

Следствие 1. Для всех $a,b,a',b'\in F_q$ и $c\in F_q^{\times}$, $q=2^{2m+1}$ справедливо

$$S(a,b)^{-1}=S(a,b+\pi(a)a),$$

$$S(a,b)^{S(a',b')}=S(a,b+\pi(a)a'+\pi(a')a).$$

Предложение 2 [3]**.** Пусть,

$$\Im=\left\{S(a,b)\middle|a,b\in F_q\right\},$$

$$\mathrm{H}=\left\{M(c)\middle|c\in F_q^{\times}\right\},$$

тогда $\Im\le Sz(q)$ является 2-группой с экспонентой 4, мощности $\left|\Im\right|=q^2$ и H изоморфна циклической группе $F_q^{\times}$ порядка $q-1$.

Предложение 3 [3]. Группа $\mathrm{H}=\left\{M(c)\middle|c\in F_q^{\times}\right\}$ действием $S(a,b)^{M(c)}$ порождает группу неподвижной точки свободных автоморфизмов на $\Im=\left\{S(a,b)\middle|a,b\in F_q\right\}$ и $\Im\mathrm{H}$ является группой Фробениуса с ядром Фробениуса $\Im$.

Замечание 3.

1. Из свойства группы Фробениуса $\Im\cap H=\{1\}$ порядок группы $\Im$ H равен $q^2(q-1)$.

2. Группа Судзуки содержит подгруппы $S(a,b)$, $M(c)$, Холла и подгруппы по делителям их порядков.

2. КРИВЫЕ ДЭЛИНГЭ-ЛУСТИГА, АССОЦИИРОВАННЫЕ С ГРУППОЙ СУДЗУКИ

Кривая максимального рода над квадратичним полем F_q является кривой Эрмита. По классификации кривых Дэлигнэ-Лустига кривая Эрмита ассоциируется с проективной специальной линейной группой, покрывает максимальные плоские кривые меньшего рода в квадратичном поле и имеет наилучшую асимптотическую оценку

$$A(q)=\limsup_{g\to\infty}N_q(g)/g,$$

$$A(q)\approx 2\sqrt{q},$$

где $N_q(g)$ – число точек кривой рода g.

Вторая и третья группа кривых Дэлигнэ-Лустига ассоциируются с группой Судзуки $Sz(q)$ и группой Ри $R(q)$. Кривые Судзуки и Ри широко рассмотрены в [8–10]. Эти кривые являются оптимальными кривыми в том смысле, что имеют число F_q рациональных точек относительно рода достаточно близким к границе Хассе-Вейля.

Главный результат по кривым Дэлигнэ-Лустига второго типа ассоциированных с $Sz(q)$ определяется следующей теоремой.

Теорема 1. [10] Для положительного целого s заданы $q=2q_0^2$ и $q_0=2^s$. Пусть X кривая над F_q рода g и удовлетворяются следующие условия

1. $g=q_0(q-1)$;

2. $\#X(F_q)=q^2+1$.

Тогда X является F_q изоморфной кривой Дэлигнэ-Лустига, ассоциированной с группой Сузуки $Sz(q)$.

Кривую с точностью до F_q изоморфизма, ассоциированную с подгруппой $S(a,b)$ группы Судзуки $Sz(q)$, основанную на роде, числе точек и групповом F_q автоморфизме кривой определили Хансен Дж. и Стичтенот Х. [8].

Кривая порядка q^2 из группы Судзуки порядка $q^2(q^2+1)(q-1)$, ассоциированная с подгруппой $S(a,b)$, является F_q изоморфной плоской кривой

$$y^q-y=x^{q_0}(x^q-x).$$

Кривая Судзуки рассмотренная Хансеном Дж. и Стичтенотом Х., имеет относительно своего рода максимальное число точек. Максимальное число F_q рациональных точек кривой определяется замечательной формулой Вейля и для кривой Судзуки чуть меньше границы Хассе-Вейля. Действительно, прямая подстановка в оценку числа точек Хассе-Вейля рода кривой даёт значение

$$N_q(g)=1+q+2g\sqrt{q}=\sqrt{2}q^2-(\sqrt{2}-1)q+1$$

и имеем, что $q^2+1<\sqrt{2}q^2-(\sqrt{2}-1)q+1$.

Число F_{q^r} – рациональных точек кривой $N_{q^r}(g)$ можно определить на основе вычисления специального L полинома, который является энумератором дзета функции

$$\zeta(X,t)=\exp(\sum N_{q^r}t^r/r).$$

Известен результат Хассе-Вейля

$$\zeta(X,t)=L(X,t)/\{(1-t)(1-qt)\},$$

где $L(X,t)=\prod_{k=1}^{2g}(1-\alpha_k t)$, $L(X,t)\in Z(t)$;

$\alpha_j\alpha_{j+g}=q$, $j=1,...,g$;

$|\alpha_j|=\sqrt{q}$, $j=1,...,g$.

Следуя [6], если

$$L(X,t)=\prod_{k=1}^{2g}(1-\alpha_k t),$$

тогда число точек кривой в расширенном поле F_{q^r} определяется выражением

$$N_{q^r}(g)=q^r+1-\sum_{k=1}^{2g}\alpha_k^r.$$

Для кривой Судзуки, как показано в [9], энумератор имеет следующий вид

$$L(X,t)=(1+2q_0t+qt^2)^g.$$

Полином $L(X,t)$ имеет $2g$ корней и решения для α_k разбиваются на две группы по g одинаковых значений

$$\alpha=q_0(-1+i)\ \text{и}\ \beta=\tilde{\alpha}=q_0(-1-i).$$

Для α и β легко проверяются условия $\alpha_j\alpha_{j+g}=q$ и $|\alpha_j|=\sqrt{q}$. Для $N_{q^r}(g)$ получим результирующее выражение

$$N_{q^r}(g)=q^r+1-g(\alpha^r+\beta^r).$$

Рассмотрим основные случаи расширенного поля F_{q^r} и значения числа точек для кривой Судзуки. Подставляя в $N_{q^r}(g)$ степени расширения $r=\overline{1,4}$, получим

$$N_q(g)=q+1-g(\alpha+\beta)=q+1+2gq_0=$$
$$q+1+2q_0q_0(q-1)=q^2+1;$$
$$N_{q^2}(g)=q^2+1-g(\alpha^2+\beta^2)=$$
$$q^2+1-gq_0{}^2(-2i+2i)=q^2+1;$$
$$N_{q^3}(g)=q^3+1-g(\alpha^3+\beta^3)=$$
$$q^3+1-4gq_0{}^3=q^3+1-4q_0{}^4(q-1)=q^2+1;$$
$$N_{q^4}(g)=q^4+1-g(\alpha^4+\beta^4)=$$
$$q^4+1+8gq_0{}^4=q^4+1+2q_0(q-1)q^2.$$

Замечание 4.

1. Кривая Судзуки над полем F_q является оптимальной, по числу точек лежит близко к границе Хассе-Вейля для кривой рода $g=q_0(q-1)$.

2. Кривая Судзуки над квадратичным и кубическим полем является неоптимальной.

3. Кривая Судзуки над конечным полем степени расширения 4 является максимальной. Подстановка значения рода $g=q_0(q-1)$ в выражение Хассе-Вейля дает

$$N_{q^4}(g)=q^4+1+2q_0(q-1)q^2,$$

что равно числу точек кривой в F_{q^4}.

4. Более общий результат получен в [9], где показано, для расширений $r\equiv 0\bmod 4$ кривая Судзуки является максимальной.

Действительно

$$N_{q^{4s}}(g)=q^{4s}+1-g(\alpha^{4s}+\beta^{4s})=$$
$$q^{4s}+1+2\cdot 4^s gq_0{}^{4s}=q^{4s}+1+2q_0(q-1)q^{2s},\quad s\geq 1.$$

5. В работе [9] рассмотрена кривая Судзуки

$$y^q-y=x^{q_0}(x^q-x)$$

с параметрами $q_0=2^s$ и $q=2q_0$.

Показано, что кривая имеет род $g=q_0(q-1)$, число точек $N_q(g)=q^2+1$ так же является F_q – изоморфной кривой Дэлигнэ-Лустига, ассоциированной с группой Судзуки $Sz(q)$. По сравнению с определением из теоремы 1, кривая определена над меньшим в q_0 раз конечным полем и содержит в $q_0{}^2$ меньше точек.

6. В работе [11] получены производные кривые по подгруппам группы Судзуки для случая $q=2q_0{}^2$ и $q_0=2^s$.

По циклической подгруппе порядка r, $r|q-1$ кривая рода $g=q_0(q-1)/r$ имеет вид

$$V^{(q-1)/r}f(U)=(1+U^{q_0})(U^{q-1}+V^{2(q-1)/r}),$$

где $f(t)=1+\sum_{i=0}^{s-1}t^{2^i(2q_0+1)-(q_0+1)}(1+t)^{2^i}$.

По подгруппе Зингера порядка $q+2q_0+1$ кривая рода $g=(q+2q_0+1)(q_0-1)/r+1$ имеет вид

$$V^{(q+2q_0+1)/r}\tilde{f}(U)=U^{q+2q_0+1}+V^{2(q+2q_0+1)/r},$$

где $\tilde{f}(t)=1+\sum_{i=0}^{s-1}t^{2^iq_0}(1+t)^{2^i(q_0+1)-q_0}+t^{q/2}$.

По подгруппе Зингера порядка $q-2q_0+1$ кривая рода $g=(q-2q_0+1)(q_0-1)/r-1$ имеет вид

$$bV^{(q-2q_0+1)/r}f(U)=$$
$$=(U^{q_0-1}+V^{2q_0-1})(U^{q-2q_0+1}+V^{2(q-2q_0+1)/r}),$$

где $b=\lambda^{q_0}+\lambda^{q_0-1}+\lambda^{-q_0}+\lambda^{-(q_0-1)}$, $\lambda\in F_{q^4}$, порядка $q-2q_0+1$.

7. В работе [11] также рассмотрены кривые, ассоциированные с подгруппами группы Судзуки порядков 2^ur, $2r$, $2s$, $4s$, где $r|(q-1)$, $s|(q\pm 2q_0+1)$.

Кривые по данным подгруппами имеют число точек соответствующими порядкам подгрупп, что меньше числа точек кривой ассоциированной с подгруппой $S(a,b)$ порядка q^2.

Замечание 5.

1. Универсальное хеширование по кривой Судзуки для случая $q_0=2^s$ и $q=2q_0{}^2$, ассоциированное с дивизором $D=|(q+2q_0+1)P_0|$ определяется на F_q рациональном морфизме $\pi:=(1:x:y:v:w)$ в P^4, где x,y,v,w определяются уравнениями [8]

$$y^q-y=x^{q_0}(x^q-x),$$
$$v:=x^{2q_0+1}+y^{2q_0},$$
$$w:=xy^{2q_0}+x^{2q+2q_0}+y^{2q},$$

и порядки полюсов равны

$$div_\infty(x) = qP_0,$$

$$div_\infty(y) = (q+q_0)P_0,$$

$$div_\infty(v) = (q+2q_0)P_0,$$

$$div_\infty(w) = (q+2q_0+1)P_0.$$

Кривая Судзуки определяется множеством точек вида

$$P_{(a,b)} := (1:a:b:f(a,b):af(a,b)+b^2) \cup \pi(P_0) = (0:0:0:0:1),$$

где $a,b \in F_q$ и $f(a,b) := a^{2q_0+1} + b^{2q_0}$ [12].

Хеширование по рациональным функциям кривой Судзуки определяет универсальный хеш класс $\varepsilon - U(q^2, q^k, q)$, где q^2 – число хеш функций (объём ключевого пространства), q^k – объём пространства сообщений, q – объём пространства хеш кодов. Вероятность коллизии ε определяется соотношениями

$$\varepsilon = (i(q+2q_0) + j(q+2q_0+1) + t(q+q_0) + rq)/q^2, \text{ если } k < q_0(q-1),$$

$$\varepsilon = (k + q_0(q-1))/q^2, \text{ если } k \ge q_0(q-1),$$

где $0 \le i \le q_0 - 1, 0 \le j \le q_0 - 1, 0 \le t \le 1, 0 \le r \le q-1$.

2. Пусть $k \approx \sqrt{q}(\sqrt{q}-1)/2$. Имеем, значения параметров $i \approx s$, $j = 0$, $t = 0$, $r = 0$, $s \approx \left\lceil (3k)^{1/3} \right\rceil \approx (q)^{1/3}$ и оценку вероятности коллизии

$$\varepsilon \approx q^{1/3}(q+q_0)/q^2 \approx q^{-1/6}\varepsilon_{ЭК},$$

где $\varepsilon_{ЭК} = 1/\sqrt{q} + 1/q$ – значение вероятности коллизии универсального хеширования по кривой Эрмита в квадратичном поле F_q при $k = \sqrt{q}(\sqrt{q}-1)/2$. Из оценки $\varepsilon \approx q^{-1/6}\varepsilon_{ЭК}$ следует выигрыш в $q^{1/6}$ раз по вероятности коллизии относительно хеширования по кривой Эрмита. Размер ключевых данных по сравнению с хешированием по Эрмита больше в $\sqrt{q}$.

3. Для $k = q_0(q-1)$ имеем вероятность коллизии хеширования по кривым Сузуки

$$\varepsilon = 2q_0(q-1)/q^2 \approx \sqrt{2}q^{-1/2}.$$

Подстановка $k = q_0(q-1)$ в выражение для вероятности коллизии хеширования по кривым Эрмита дает

$$\varepsilon = k/q^3 + 1/(2q) - 1/(2q^2) = \frac{1}{\sqrt{2}}\sqrt{q}(q-1)/q\sqrt{q} + 1/(2\sqrt{q}) - 1/(2q) \approx 1/\sqrt{2}.$$

4. Универсальное хеширование по кривой Судзуки для случая $q_0 = 2^s$ и $q = 2q_0$, ассоциированное с дивизором $D = |(q+2q_0+1)P_0|$ определяется на том же F_q рациональном морфизме $\pi := (1:x:y:v:w)$ в P^4 с теми же порядками полюсов как для случая кривой с параметрами $q_0 = 2^s$ и $q = 2q_0^2$. Следовательно, справедливо определение и соотношения для вероятности коллизии из п.1 замечания 5.

Для $k = q_0(q-1)$ получим оценку вероятности коллизии хеширования

$$\varepsilon = 2q_0(q-1)/q^2 \approx (4q_0^2 - 2q_0)/4q_0^2 \approx 1,$$

что хуже по сравнению с хешированием с параметрами $q_0 = 2^s$ и $q = 2q_0^2$

$$\varepsilon = 2q_0(q-1)/q^2 \approx 1/q_0.$$

5. Функциональное поле максимальной кривой Судзуки над конечным полем F_{q^4} рассмотрено в [13]. Универсальное хеширование по максимальной кривой Судзуки над конечным полем F_{q^4} вычисляется по пяти параметрическим функциям базиса $L(lD)$ в выражении

$$S := \left\{ \begin{array}{l} x^a y^b v^c w^d (x^q + x)^r \\ aq + b(q+q_0) + c(q+2q_0) + \\ \quad + d(q+2q_0+1) + rq^2 \le l(q^2+1) \\ 0 \le a \le q-1, 0 \le b \le 1, 0 \le c \le q_0 - 1, \\ 0 \le d \le q_0 - 1, 0 \le r \le l \end{array} \right\}.$$

Функциональное поле в векторном пространстве выстраивается в порядке возрастания полюсов $f = x^a y^b v^c w^d (x^q + x)^r$. Для числа слов данных $k < q(q_0 - 1)^2$ вычисления рациональных функций осуществляется по параметрам x, y, v, w.

Для крайних параметров a,b,c,d в выражении для S будет иметь значение дискретной оценки

$$\nu_\infty(f) = (q-1)q + (q+q_0) + (q_0-1)(q+2q_0) + (q_0-1)(q+2q_0+1) = q^2 + 2g - 1.$$

Для числа слов данных $k > q(q_0-1)^2$ вычисление рациональных функций следует выполнять по пяти параметрам $x, y, v, w, (x^q + x)$, что позволяет построить функциональное пространство с непрерывной последовательностью полюсов для больших значений k.

Для $k = q_0(q-1)$ получим оценку вероятности коллизии хеширования над полем F_{q^4}

$$\varepsilon = 2q_0(q-1)/q^4 \approx \frac{1}{q^2 q_0}.$$

Определение хеширования по кривой Судзуки над полем F_p с параметрами $p_0 = 2^s$, $p = 2p_0^2$ и $p \approx q^4$ для $k = q_0(q-1)$ слов данных аналогично п.2 замечания 5 и будет иметь оценку вероятности коллизии

$$\varepsilon \approx \rho_k / p^2 \approx \frac{k^{1/3}(q+q_0)}{q^8} \approx \frac{1}{q^6 q_0},$$

что существенно лучше по сравнению с хешированием в расширенном поле F_{q^4}.

ЗАКЛЮЧЕНИЕ

Кривые по подгруппам группы Судзуки имеют число точек, соответствующим порядкам подгрупп. Наилучшие параметры универсального хеширования по вероятности коллизии и затратам ключа достигаются по кривой Судзуки, ассоциированной с подгруппой $S(a,b)$ порядка q^2.

Кривая Судзуки над конечным полем четвертой степени расширения является максимальной, но имеет относительно малое значение рода и поэтому проигрывает по вероятности коллизии. Практическое применение таких кривых может быть перспективным для алгебраического кодирования.

Кривая Судзуки с параметрами $q_0 = 2^s$ и $q = 2q_0$ имеет меньшее отношение числа точек к роду, что определяет её проигрыш хешированию с параметрами $q_0 = 2^s$ и $q = 2q_0^2$.

Кривые ассоциированные с подгруппами Зингера и подгруппами меньших порядков имеют малое число точек, сложность построения и следовательно их применение лежит в плоскости простых кодовых схем.

Литература

[1] *Suzuki M.* A new type of simple groups of finite order. / M.Suzuki // Proc. Nat. Acad. Sci. U.S.A. – 1960. – V. 46. – P. 868–870.

[2] *Suzuki M.* On a class of doubly transitive groups. / M. Suzuki // Ann. of Math. – 1962. – V. (2)75. – P.105–145.

[3] *Huppert B.* Finite groups. III. / B. Huppert, N. Blackburn // Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], vol. 243, Springer-Verlag, Berlin. – 1982. – vol. 243.

[4] *Carter R.W.* Simple groups of Lie type. / R.W. Carter // Reprint of the 1972 original. Wiley Classics Library. A Wiley-Interscience Publication. John Wiley & Sons, Inc., New York. – 1989. – P.335.

[5] *Wilson R.A.* The finite simple groups. / R.A.Wilson // Graduate Texts in Mathematics 251. Springer-Verlag London, Ltd., London. – 2009. – P. 298.

[6] *Jones G.A.* Varieties and simple groups. / G.A. Jones // J. Austr. Math. Soc.-1974. – V.17. – P. 163–173.

[7] *Landazuri V.* On the minimal degrees of projective representations of the finite Chevalley groups. / V.Landazuri and G. M.Seitz // J. Algebra. – 1974. – V.32. –P. 418–443.

[8] *Hansen J.P.* Group codes on certain algebraic curves with many rational points. / J.P.Hansen, H.Stichtenoth // AAECC 1. – 1990. – P. 67–77.

[9] *Hansen J.P.* Deligne-Lusztig varieties and group codes. / J.P. Hansen // Lect. Notes Math. – 1992. – V. 1518. – P. 63–81.

[10] *Pedersen J.P.* A function field related to the Ree group. / J.P. Pedersen // Lect. Notes Math. – 1992. – V. 1518. – P. 122–131.

[11] *Giulietti M.* Quotient curves of the Suzuki curve. / M.Giulietti, G.Korchmarros, F.Torres //Acta Arith. – 2006. – no. 3. – P. 245–274.

[12] *Tits J.* Ovo¨ıdes et groupes de Suzuki. / J.Tits // Archive Mathematics. – 1962. – N.13. –P.187–198.

[13] *Eid A.* Suzuki-invariant codes from the Suzuki curve. / A.Eid, H.Hasson, A.Ksir, J.Peachey // arXiv: 1411.6215v2[math.AG] 25 Nov 2014.

Поступила в редколлегию 26.11.2015

Котух Евгений Владимирович, аспирант кафедры БИТ ХНУРЭ. Научные интересы: аутентификация данных и кодирование.

Халимов Геннадий Зайдулович, доктор технических наук, профессор кафедры БИТ ХНУРЭ. Научные интересы: методы и средства аутентификации данных.

УДК 681.3.06

Універсальне хешування по кривих, які асоційовані з групою Судзукі / Е.В. Котух, Г.З. Халімов // Прикладна радіоелектроніка: наук.-техн. журнал. – 2015. – Том 14. – № 4. – С. 361–365.

Наведено результати універсального гешування по кривих, які асоційовані з групою Судзукі над розширеннями кінцевого поля. Розглянуто проективні розмаїття точок кривих, поля раціональних функцій. Отримано порівняльні оцінки ймовірності колізії універсального гешування. З оцінки випливає, що найкращий результат досягається на кривій Судзукі над полем F_q з параметрами $q = 2q_0^2$ і $q_0 = 2^s$.

Ключові слова: універсальне гешування, група Судзукі, криві Судзукі.

Бібліогр.: 13 найм.

UDC 681.3.06

Universal hashing by curves associated with Suzuki group / E.V. Kotukh, G.Z. Khalimov // Applied Radio Electronics: Sci. Journ. — 2015. — Vol. 14. — № 4. — P. 361–365.

The paper presents the results of universal hashing by curves associated with the Suzuki group on extensions of a finite field. A projective variety of points of the curves, fields of rational functions are considered. Comparative estimates of the probability of universal hashing collisions are obtained. The estimate has shown that the best result is achieved on the Suzuki curve over the field F_q with parameters $q = 2q_0^2$ and $q_0 = 2^s$.

Keywords: universal hashing, the Suzuki group, Suzuki curves.

Ref.: 13 items.

МЕТОДЫ И СРЕДСТВА ЭЛЕКТРОННОЙ ИДЕНТИФИКАЦИИ

УДК 681.3.06(07)

БАЗОВІ ПРИНЦИПИ ЕЛЕКТРОННОЇ ІДЕНТИФІКАЦІЇ. КОНЦЕПЦІЯ ІНФРАСТРУКТУРИ ЕЛЕКТРОННОЇ ІДЕНТИФІКАЦІЇ УКРАЇНИ

О.В. ПОТІЙ, Ю.Н. КОЗЛОВ, І.І. ДЬЯЧЕНКО

В роботі на основі аналізу досвіду країн членів ЄС пропонується модель національної інфраструктури електронної ідентифікації України. Розглядаються основні моделі взаємодії користувачів, постачальників послуг електронної ідентифікації та постачальників електронних послуг. Висуваються базові принципи побудови національних систем електронної ідентифікації.

Ключові слова: електронна ідентифікація, електронне урядування, електронні довірчі послуги.

ВСТУП

Український уряд активно проводить програму реформ, які спрямовують країну шляхом європейської інтеграції. Головними пріоритетами реформ є децентралізація, відкриті принципи державного управління, прозорість та підзвітність. Електронне урядування визначено як один з найважливіших інструментів для досягнення цих цілей.

Впровадження сучасних засобів і схем електронної ідентифікації в Україні із забезпеченням високого рівня гарантій електронної ідентифікації відкриває можливості для громадян України на новому рівні здійснювати взаємодію з державними та місцевими органами влади та отримувати електронні адміністративні послуги.

Країни-члени ЄС мають великий досвід у впровадженні національних систем електронної ідентифікації, вивчення та аналіз якого складуть надійну основу для побудови інфраструктури електронної ідентифікації України.

1. БАЗОВІ МОДЕЛІ ЕЛЕКТРОННОЇ ІДЕНТИФІКАЦІЇ

На сьогодні визначено три базові моделі електронної ідентифікації за ознакою відповідальності за збереження ідентифікаційних даних [1].

Модель, орієнтована на користувача (рис. 1). В цій моделі саме користувач бере на себе відповідальність за генерацію та підтримку ідентифікаційних даних перед постачальниками послуг електронної ідентифікації та постачальниками електронних послуг. Ідентифікаційні дані надаються виключно за запитом постачальника послуг, якщо користувач дає на це згоду. Користувач завжди є власником свої ідентифікаційних даних.

Централізована модель (рис. 2). Така модель є найбільш поширеною на практиці. В цій моделі, якщо користувачі бажають отримати доступ до електронних послуг, вони спочатку мають надати ідентифікаційну інформацію у спеціальний орган реєстрації. Ідентифікаційні дані зберігаються централізовано у реєстрі під управлінням постачальника послуг електронної ідентифікації. Під час запиту електронної послуги, користувач дає згоду на те, що постачальник послуг електронної ідентифікації надає відповідну інформацію автентифікації постачальнику електронних послуг. У цій моделі користувач не має контролю за своєю ідентифікаційною інформацією. За її збереження та використання несе повну відповідальність постачальник послуг електронної ідентифікації.

Рис. 1. Модель, орієнтована на користувача

Рис. 2. Централізована модель

Децентралізована модель (рис. 3). В цій моделі ідентифікаційні дані користувача зберігаються у різних постачальників послуг електронної ідентифікації. Але постачальники послуг електронної ідентифікації можуть легко обмінюватись ідентифікаційними даними за допомогою відповідних репозиторіїв. Довіра у цій моделі досягається за рахунок встановлення відношень між різними постачальниками послуг електронної ідентифікації та на основі узгодження політик використання ідентифікаторів.

В табл. 1 наведено дані щодо розповсюдженості перелічених моделей серед країн-членів ЄС. Досвід країн-членів ЄС свідчить що децентралізована модель найбільше підходить для національної системи національної електронної ідентифікації, як найбільш гнучка: додаткові послуги, такі як мобільна ідентифікація чи електронна ідентифікація для фізичних осіб, легко інтегруються в єдину інфраструктуру. Також ця модель оптимальна з точки зору інтероперабельності зі сторонніми системами електронної ідентифікації.

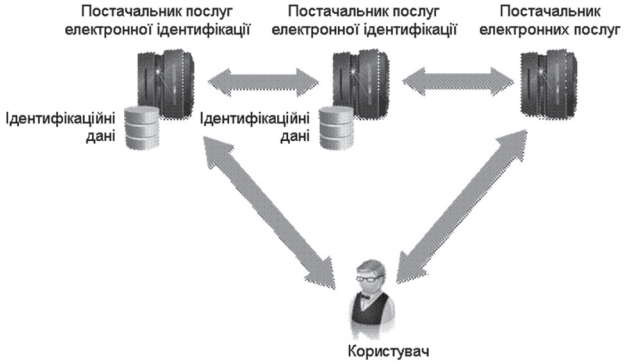


Рис. 3. Децентралізована модель

2. ОСНОВНІ ПРИНЦИПИ ДЕРЖАВНОЇ ПОЛІТИКИ ЩОДО ІДЕНТИФІКАЦІЇ ГРОМАДЯН

На основі аналізу підходів країн-членів ЄС до регламентування електронної ідентифікації громадян було виділено ключові питання державної політики, що можуть мати вплив на успіх впровадження національної системи електронної ідентифікації.

Державна монополія. *Державна* монополія на електронну ідентифікацію громадян притаманна більшості з розглянутих країн ЄС. В деяких країнах, наприклад в Естонії, державна монополія закріплена законодавством, проте в більшості випадків монополія зумовлена відсутністю конкуруючих недержавних систем. Така тенденція свідчить про те що якісне надання послуг електронної ідентифікації громадян може забезпечити лише держава.

Примусове розповсюдження засобів електронної ідентифікації. Картки eID можуть бути як заміною традиційних паспортів, так і додатковим необов'язковим ідентифікаційним посвідченням. Досвід фінської системи FINID засвідчує, що другий підхід може стати завадою для розповсюдження технології електронної ідентифікації: фінська система була введена в експлуатацію в 1999-му році, проте станом на 2009 рік показник проникнення сягав лише 5% від населення країни [2].

Принцип «одна особа = один ідентифікатор». Унікальний ідентифікатор громадянина, що єдиноразово видається певним державним реєстром на все життя, робить децентралізовану систему електронної ідентифікації більш зручною як для громадянина, так і для державних чи приватних установ. Громадянин має змогу використовувати єдиний ідентифікатор, замість декількох секторальних, таких як водійське посвідчення та картка медичного страхування. Установи, у свою чергу, отримають змогу спростити процедуру отримання інформації про громадянина, адже достатньо буде здійснити запит до потрібного реєстру за ідентифікатором громадянина. Прикладом такого підходу може слугувати естонська система електронної ідентифікації, в якій всі дані громадянина пов'язані з Персональним Ідентифікаційним Кодом(PIC).

3. КОНЦЕПЦІЯ ІНФРАСТРУКТУРИ ЕЛЕКТРОННОЇ ІДЕНТИФІКАЦІЇ УКРАЇНИ

Спираючись на вищенаведені міркування, як базова модель електронної ідентифікації в Україні пропонується модель, яка наведена на рис. 4. Місце інфраструктури ідентифікації у загальній архітектурі пояснюється на рис. 5.

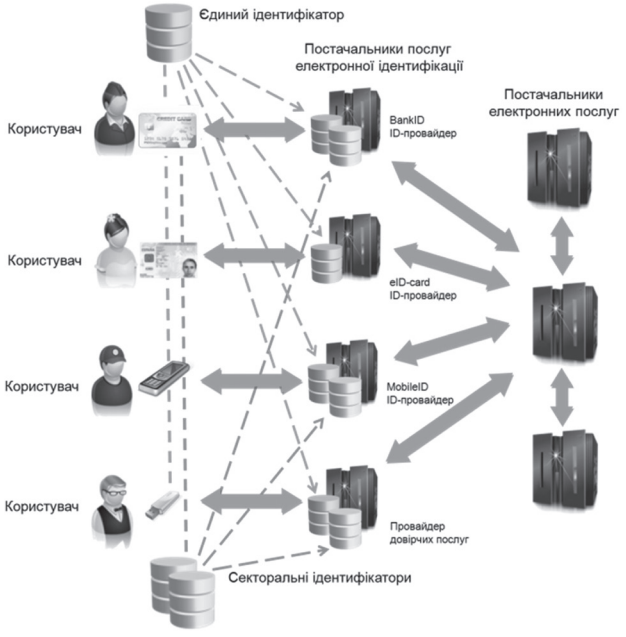


Рис. 4. Модель інфраструктури електронної ідентифікації України

Таблиця 1

Порівняння національних систем електронної ідентифікації країн-членів ЄС

	Австрія	Бельгія	Германія	Естонія	Іспанія	Польща	Фінляндія	Франція
Децентралізована модель	+	+	+	+	+	+	+	+
Послуги «мобільна ідентифікація»	+	+	+	+	+	+	+	+
eID для юридичних осіб		+		+	+			

Модель передбачає децентралізоване зберігання ідентифікаційних даних та реалізує такі принципи:

1. Інфраструктура електронної ідентифікації будується на основі унікального ідентифікатора громадянина, який має присвоюватися кожному громадянину та резиденту України.

2. Формування та підтримка унікального ідентифікатора громадянина має здійснюватися визначеним державним органом шляхом централізованої підтримки відповідного єдиного реєстру.

3. Порядок формування та формат унікального ідентифікатора громадянина має бути визначений державним нормативним актом. Унікальний ідентифікатор громадянина надається особі в порядку, визначеним законодавством з обов'язковою реєстрацією громадянина у єдиному реєстрі.

4. Унікальний ідентифікатор громадянина має бути однозначно зв'язаним із особою та діяти протягом всього життя особи (принцип одна особа = один ідентифікатор).

5. Інші ідентифікатори, які можуть використовуватися у конкретних сферах цивільних стосунків та соціальних відносин (секторальні ідентифікатори), мають бути пов'язані із унікальним ідентифікатором громадянина.

6. Політика управляння унікальним ідентифікатором громадянина має бути пов'язаною з політикою управління документів, які засвідчують особу громадянина.

7. Держава має монополію на надання та управління унікальним ідентифікатором громадянина.

ВИСНОВКИ

Розглянуто основні моделі електронної ідентифікації та ключові принципи державної політики щодо ідентифікації громадян. Ґрунтуючись на досвіді країн членів ЄС, була висунута модель інфраструктури електронної ідентифікації України, що передбачає децентралізоване зберігання ідентифікаційних даних, прив'язаних до унікального ідентифікатора громадянина. Запропонована модель оптимальна з точки зору гнучкості та інтероперабельності і є найбільш зручною для користувачів.

Література

[1] Palfrey J., Gasser U.: Digital Identity Interoperability and eInnovation, Case Study, November 2007, Berkman Publication Series.

[2] eID Interoperability for PEGS: Finnish country profile, 2009.

[3] eID Estonian experience, May 2014, E-Governance Academy.

[4] *О. Потій, Ю. Козлов:* Національная стратегія електронної ідентифікації України. Біла книга з електронного урядування, Березень 2015.

[5] National ID Cards and Electronic Identity Documents: Route to Success? Jean-Noel Georges, 15th March 2011.

Надійшла до редколегії 8.10.2015

Потій Олександр Володимирович, доктор технічних наук, професор, професор кафедри безпеки інформаційних та телекомунікаційних систем Харківського національного університету ім. В.Н. Каразіна. Наукові інтереси: проектування комплексних систем захисту інформації, криптографічних засобів захисту; методи системного аналізу процесів захисту інформації; методи оцінки безпеки об'єктів критичної інформаційної інфраструктури; електронні довірчі послуги, цифровий підпис та eID.

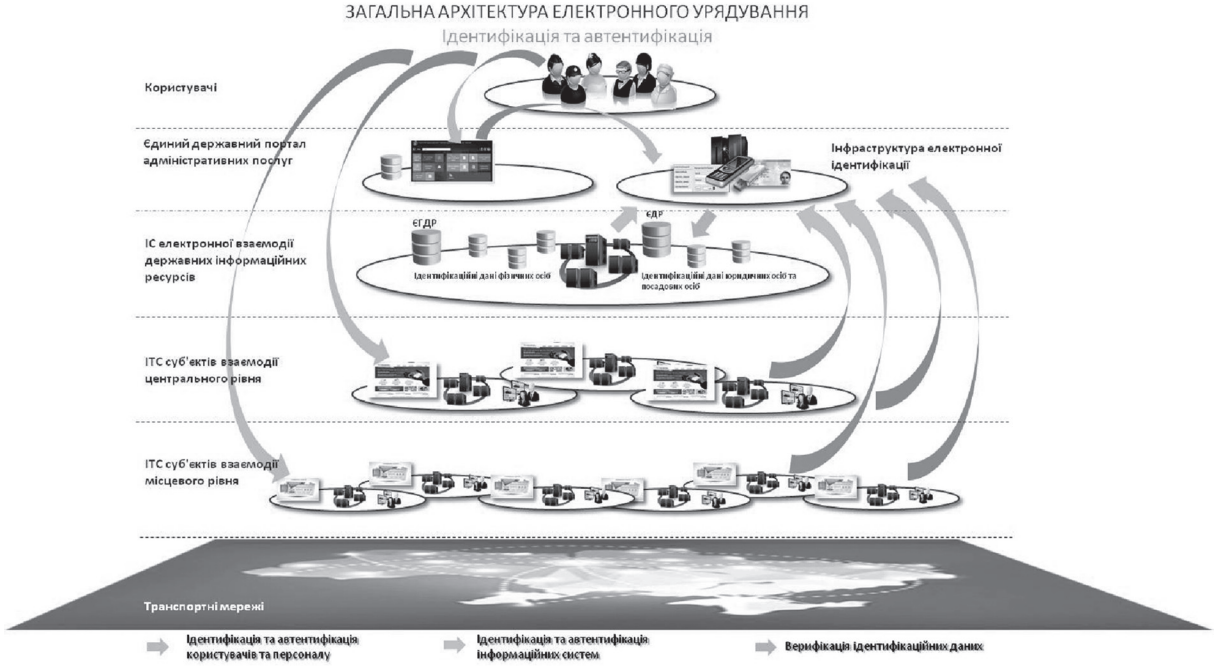


Рис. 5. Місце інфраструктури електронної ідентифікації у загальній архітектурі електронного урядування

Козлов Юрій Миколайович, ДП «Інформаційний центр», Міністерства юстиції України. Наукові інтереси: система електронного цифрового підпису, системи електронної ідентифікації, електронні довірчі послуги.

Дьяченко Ігор Ігорович, студент кафедри комп'ютерних систем та мереж Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут». Наукові інтереси: системи електронної ідентифікації

УДК 681.3.06(07)

Базовые принципы электронной идентификации. Концепция инфраструктуры электронной идентификации Украины / А. В. Потий, Ю.Н. Козлов, И.И. Дьяченко // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 366–369.

В работе на основе анализа опыта стран-членов ЕС предлагается модель национальной инфраструктуры электронной идентификации Украины. Рассматриваются основные модели взаимодействия пользователей, поставщиков услуг электронной идентификации и поставщиков электронных услуг. Выдвигаются базовые принципы построения национальных систем электронной идентификации.

Ключевые слова: электронная идентификация, электронные доверительные услуги, электронное управление.

Рис.: 5. Библиогр.: 5 назв.

UDC 681.3.06(07)

Basic principles of electronic identification. Ukrainian eID framework draft / O.V. Potii, Yu.M. Kozlov, I.I. Diachenko // Applied Radio Electronics: Sci. Journ. – 2015. – Vol. 14. – № 4. – P. 366–369.

The Ukrainian government has been actively pursuing the reforms that lead the country towards European integration. The main priorities of the reform are decentralization, transparency and accountability. E-government is considered as one of the most important tools to achieve these goals. In this paper we substantiate basic principles of national eID systems, relying on the experience of EU member countries. In addition a draft of Ukraine's eID framework is proposed, which implements those principles.

Keywords: electronic identification, electronic trust services, electronic government.

Fig.: 5. Ref.: 5 items.

УДК 681.3.06(07)

МЕТОДИ ФОРМУВАННЯ УНІКАЛЬНОГО ЕЛЕКТРОННОГО ІДЕНТИФІКАТОРА І ПРОЦЕС ВИДАЧІ ЕЛЕКТРОННИХ ПОСВІДЧЕНЬ: ДОСВІД ЄВРОСОЮЗУ

О.В. ПОТІЙ, Є.В. БРОШЕВАН

Розглядається функціонально-прикладний рівень функціональної моделі електронної ідентифікації, яка була описана в Національній стратегії електронної ідентифікації України. Аналізуються підходи до формування унікальних електронних ідентифікаторів, їх алгоритмів і необхідних ключових даних для цієї мети, беручи до розгляду досвід країн ЄС. Розглядаються принципи видачі електронних посвідчень, ролі кожного з учасників цього процесу. А також, на підставі положень Національної стратегії, пропонуються рекомендації з відповідних процесів для впровадження в Україні.

Ключові слова: електронна ідентифікація, електронне урядування, реєстраційні дані, унікальний ідентифікатор.

ВСТУП

В силу того, що в Україні починає формуватися ринок електронної ідентифікації та ринок електронних послуг, виникає необхідність в детальній розробці усіх аспектів цих сервісів. Основним документом на цей період є Національна стратегія електронної ідентифікації України(Біла книга з електронного урядування), далі – Стратегія [1].

Місія Стратегії – побудова такої інфраструктури електронної ідентифікації в Україні, завдяки якій громадяни можуть безперешкодно отримувати доступ до інформації та он-лайн послуг з різних джерел.

Стратегія спрямована на досягнення таких цілей.

1. Побудова інфраструктури електронної ідентифікації в Україні.

2. Забезпечення інтероперабельності (сумісності) інфраструктури електронної ідентифікації.

3. Створення довірчого середовища у кіберпросторі України та мотивування громадян до використання електронних послуг.

4. Забезпечення сталого розвитку національної інфраструктури електронної ідентифікації та пов'язаних з нею інших електронних послуг.

Перші дві цілі спрямовані на проектування та створення національної інфраструктури електронної ідентифікації та управління нею для надання громадянам, приватному та державному сектору національної економіки електронних послуг.

Важливим питанням у побудові інфраструктури електронної ідентифікації є процес формування унікального ідентифікатора та процес видачі електронних посвідчень, бо саме це є унікальною ідентифікуючою сутністю, яка дає можливість використовувати інші пов'язані послуги та виконувати необхідні регулятивні дії.

Задля створення системи, яка б ураховувала усі недоліки існуючих систем та працювала належним чином, треба розглянути досвід країн, де це вже успішно функціонує протягом тривалого часу.

1. ФУНКЦІОНАЛЬНО-ПРИКЛАДНИЙ РІВЕНЬ ІНФРАСТРУКТУРИ ЕЛЕКТРОННОЇ ІДЕНТИФИКАЦІЇ

Стратегія визначає такі рівні функціональної моделі: функціонально-прикладний, експлуатаційний, рівень функціональної сумісності, рівень управління. До базових операцій, що здійснюються на функціонально-прикладному рівні, входять:

- реєстрація;
- управління реєстраційними даними;
- автентифікація;
- авторизація;
- посередництво під час транзакцій.

У роботі розглядаються функції, що є складовою частиною рівню управління реєстраційними даними. Етап управління реєстраційними даними (credential management) містить процеси, пов'язані з управлінням життєвим циклом реєстраційних даних або засобів для їх створення, що дають користувачеві можливість брати участь у будь-якій діяльності або в якому-небудь контексті.

Етап управління реєстраційними даними може включати всі наступні процеси або їх частину:

- створення реєстраційних даних;
- випуск реєстраційних даних або засобів для їх створення;
- активацію реєстраційних даних або засобів для їх створення;
- зберігання реєстраційних даних;
- анулювання та/або знищення реєстраційних даних або засобів для їх створення;
- оновлення та/або заміну реєстраційних даних або засобів для їх створення;
- ведення записів.

Процес створення реєстраційних даних охоплює всі необхідні процеси для створення реєстраційних даних вперше або засобів для їх створення. Ці процеси можуть включати попередню обробку, ініціалізацію і прив'язку.

Випуск реєстраційних даних – це процес забезпечення об'єкта певними реєстраційними даними чи засобами для їх створення або встанов-

лення іншого зв'язку між ними. Складність цього процесу різниться залежно від необхідного рівня гарантій електронної ідентифікації. Для більш високих рівнів гарантій може бути потрібна захищена доставка апаратного засобу електронної ідентифікації (наприклад, смарт-картки), що містить реєстраційні дані, і може знадобитися особиста доставка. Для більш низьких рівнів гарантій процесом випуску може бути проста відправка пароля або PIN-коду на фізичну або електронну поштову адресу об'єкта.

2. ПРОЦЕС СТВОРЕННЯ РЕЄСТРАЦІЙНИХ ДАНИХ

Перший крок управління – створення реєстраційних даних, за допомогою яких можна однозначно визначати людину.

На сьогодні нормативно-правовими актами України визначено такі основні та найбільш вживані ідентифікаційні дані фізичних осіб: прізвище, ім'я, по батькові, дата народження, дата смерті, стать, місце народження, місце проживання, громадянство, відцифрований зразок підпису особи, відцифрований образ обличчя особи, відцифровані відбитки пальців рук, реєстраційний номер облікової картки платника податків – фізичної особи. Але на даному етапі однозначно не визначено який саме ідентифікатор буде унікальним для країни.

Як один з варіантів – ідентифікаційний номер платника податків, але цей номер не доцільно використовувати, бо він є не у кожного громадянина країни і видається з певного віку. Універсальний ідентифікатор має бути в усіх, та реєстр, який за це відповідає, має займатися тільки цією задачею.

У цьому контексті слід наголосити на необхідності внесення змін до нормативної бази України для забезпечення використання унікального номера запису в Єдиному державному демографічному реєстрі як єдиний ідентифікатор фізичної особи.

З цією метою було проаналізовано досвід країн, в яких існує і успішно використовується єдиний унікальний ідентифікатор. Для розгляду були вибрані країни ЄС, в яких ступінь проникнення системи електронної ідентифікації більш, ніж 50% та дані щодо структури електронної ідентифікації є публічними.

Бельгія

У Бельгії UID (National Register identification number) отримує кожна фізична особа при першому зверненні в RRN (National Register). У цій країні існує три реєстри: реєстр населення [2], реєстр іноземних громадян і реєстр очікування. Реєстр очікування призначений для осіб з бельгійським громадянством, але для тих, які проживають за кордоном, інформація для них оновлюється через відповідні консульства. Реєстрація відбувається в SHAPE Federal Police / Registration section. У цьому процесі беруть участь Національний реєстр населення (Міністерство внутрішніх справ) та FEDICT (Федеральна державна служба з питань інформаційно-комунікаційних технологій).

У реєстрі для кожного громадянина зберігається така інформація: прізвище, ім'я, місце та дата народження, національність, основне місце проживання, місце і дата смерті, професія, сімейний стан, члени сім'ї, позначки про внесення до реєстру очікування.

Код RRN складається з 11 цифр. Перші 6 – дата народження, наступні 3 – порядковий номер людей, що народилися в цей же день і стать (чоловікам – непарне число, жінкам – парне). Останні 2 цифри – контрольна сума для перевірки автентичності.

При виборі RRN дотримувались таких ключових принципів:

1. Код має містити тільки постійну інформацію про громадянина.
2. Не має містити конфіденційну інформацію.
3. Один код ідентифікує тільки одну людину, одній людині не може належати кілька кодів.
4. Справжність коду має бути контрольована.
5. Повторному використанню не підлягає.

Іспанія

В Іспанії eID-картку отримують в поліції DGP (Dirección General de Policía) [3]. Справжність перевіряється органами, уповноваженими DGP: Міністерство державного управління та монетного виробництва і FNMT-RCM (Stamp Factory – Royal Mint). Органом сертифікації є управління поліції, залежне від Міністерства внутрішніх справ. Фізичним особам Іспанії картка видається Міністерством внутрішніх справ (Ministerio del Interior), відділ DNI (Documento Nacional de Identidad).

UID (Unique Identification Number) в Іспанії називається TIN (Tax Identification Number). TIN використовується не тільки в eID картках, але й у водійському посвідченні, в паспорті і для здійснення фінансових операцій. Він складається з 9 символів: 8 цифр і 1 буква для контролю даних. Для іноземців: 1 буква описує тип особистості, 7 цифр, 1 буква для контролю даних. Детальний опис структури наведено в табл. 1.

Таблиця 1

Структура TIN

Формат	Опис	Коментарі
99999999L	8 цифр + 1 буква	Особистості іспанської національності
X9999999L	X + 7 цифр + 1 буква	Особистості без іспанського громадянства
Y9999999L	Y + 7 цифр + 1 буква	Іноземці старше 18 років
M9999999L	M + 7 цифр + 1 буква	Громадяни Іспанії старше 14 років, з іноземною пропискою, які приїхали в Іспанію на термін не більше 6 місяців
L9999999L	L + 7 цифр + 1 буква	Громадяни Іспанії до 14 років
K0000001L	K + 7 цифр + 1 буква	Іноземці до 18 років

Австрія

В Австрії всі громадяни зареєстровані в Central Register of Residence (CRR) [4] і мають свій унікальний номер (CRR Number). Ідентифікація базується на Identity Link (IL). IL – це спеціальна структура даних, яка містить ПІБ громадянина, дату народження, унікальний ідентифікатор і сертифікат.

Концепція Австрійського eID передбачає використання електронних послуг особами, які не є записаними в CRR (наприклад, іноземним громадянам). Ці люди мають бути зареєстровані в додатковому реєстрі – Supplementary Register for Natural Persons (SR). Після чого вони отримують ті ж права, що і громадяни Австрії.

Австрійська система eID використовується для трьох випадків:

– Ідентифікація та автентифікація австрійських громадян.

– Ідентифікація та автентифікація іноземних громадян.

– Ідентифікація та автентифікація правових тотожностей та електронних мандатів.

UID являє собою soursePIN. Він визначається на основі CRR. Процес утворення базується на алгоритмі шифрування 3DES, здійснюється органом реєстрації sourcePIN (The SourcePIN Register Authority), який є частиною комісії із захисту даних. sourcePIN зберігається тільки на картці.

The SourcePIN Register Authority відповідає за додаток, який витягує sourcePIN. Секретний ключ, який використовується для отримання sourcePIN, відомий тільки The SourcePIN Register Authority.

Детальний процес утворення CRR зображено на рис. 1.

Step	Derivation	Example Result
0	Base number	000247681888 (Ex: CRR number, 12-digit decimal number
1	Binary represen-tation	00 0E C3 53 60 (5 byte, hexadecimal number)
2	Expanded to 128 Bit	00 0E C3 53 60 FF 00 0E C3 53 60 00 0E C3 53 60 (16 byte, seed value, set to 'FF' as an example)
3	Triple-DES encryption, hexadecimal	42 AD 37 74 FA E0 70 7B 31 DC 6D 25 29 21 FA 49 (16 byte)
4	SourcePIN, Base64	Qo03DPRGcHsx3G0LKSH6SQ== (24 characters)

Рис. 1. Процес утворення CRR в Австрії

0. 12-значний CRR номер перетворюється в двійковий еквівалент.

1. Для збільшення стійкості шифрування додається секретне початкове значення.

2. Розширене двійкове число шифрується за допомогою 3DES.

3. Base64 використовується для кодування результату, щоб полегшити процес читання завдяки символьно-орієнтованій системі.

4. Кінцевий результат – буквено-цифровий ряд з 24 символів.

Естонія

В Естонії система ідентитету заснована на національному номері, який називається PIC (Personal Identification Code). PIC є унікальним номером [5], що привласнюється кожному громадянину і резиденту Естонії. Генерує і підтримує PIC реєстр населення (головний процесор МВД). PIC вважається виданим особі після реєстрації в реєстрі населення.

Реєстр населення видає PIC державним органам, які вперше документують особистість (при народженні або видачі посвідки на проживання або право на проживання). Як правило, люди отримують PIC з установ охорони здоров'я (новонароджені).

Рамки управління цифрового ідентитету засновані на постійному довічному цифровому ідентитеті, який з'єднує всі онлайн тотожності, які належать до того ж людині. Всі сертифікати широко поширених цифрових ідентитетів містять PIC.

Національні принципи управління цифрового ідентитету:

– Державна монополія і відповідальність ідентифікувати людину, підтвердити особистість і видача (на документах, що засвідчують особу) сертифікатів, які підтверджують особистість людини.

– Централізоване управління ідентитету.

– Принцип «одна людина = один ідентитет».

– Один-до-одного взаємовідношення сертифікатів автентифікації і цифрового підпису з користувачем документа.

– Громадська верифікація сертифікатів автентифікації і цифрового підпису через PIC.

PIC містить 11 цифр. Порядок і значення вказані на рис. 2.

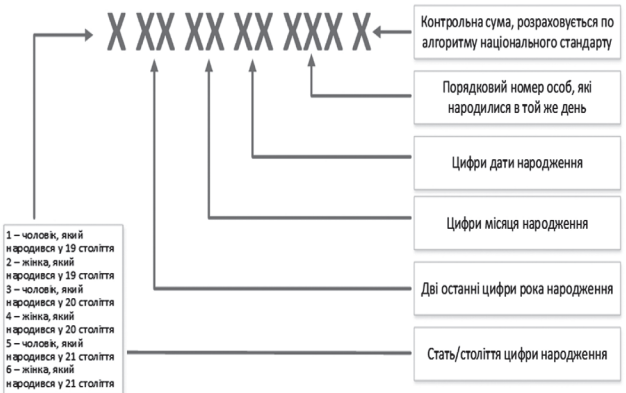


Рис. 2. Складові частини PIC в Естонії

Україна

Враховуючи державний та адміністративний устрій України, а також досвід впровадження систем електронної ідентифікації в інших країнах, як базова модель електронної ідентифікації має бути обрана децентралізована модель. Модель реалізує таки основні принципи:

1. Інфраструктура електронної ідентифікації будується на основі унікального ідентифікатора громадянина, який має присвоюватися кожному громадянину та резиденту України.

2. Формування та підтримка унікального ідентифікатора громадянина має здійснюватися певним державним органом шляхом централізованої підтримки відповідного єдиного реєстру.

3. Порядок формування та формат унікального ідентифікатора громадянина має бути визначений державним нормативним актом. Унікальний ідентифікатор громадянина надається особі в порядку, визначеним законодавством з обов'язковою реєстрацією громадянина у єдиному реєстрі.

4. Унікальний ідентифікатор громадянина має бути однозначно зв'язаним із особою та діяти протягом всього життя особи (принцип одна особа = один ідентифікатор).

5. Інші ідентифікатори, які можуть використовуватися у конкретних сферах цивільних стосунків та соціальних відносин (секторальні ідентифікатори), мають бути пов'язані із унікальним ідентифікатором громадянина.

6. Політика управляння унікальним ідентифікатором громадянина має бути пов'язаною з політикою управління документів, які посвідчують особу громадянина.

7. Держава має монополію на надання та управління унікальним ідентифікатором громадянина.

Згідно з Постановою Кабінету Міністрів України «Порядок оформлення ID-card»[6] ідентифікатором є унікальний номер запису в Єдиному державному демографічному реєстрі (далі – Реєстр). Він формується у разі, якщо інформація про особу вноситься до Реєстру вперше або у разі зміни статі особи [7].

Вимоги до Унікального номера запису в Реєстрі (далі – УНЗР):

1. Простота формування, використання та запам'ятовування.

2. Не повинен розпочинатися з нуля для уникнення помилок під час машинної обробки.

3. Для перевірки правильності та цілісності має містити контрольну цифру.

4. Має формуватися для людини будь-якого року народження, у тому числі, вік якої складає більше 100 років.

УНЗР представлений двома послідовностями з восьми та п'яти цифр, розділених текстовим символом "–".

X	X	X	X	X	X	X	X	-	X	X	X	X	X
I									II				III
1	2	3	4	5	6	7	8		9	10	11	12	13

Дата народження особи у форматі ISO 8601:2004 (РРРРММДЦ)

II. Цифровий чотиризначний код від 0001 до 9999 (для чоловіків код – непарний, для жінок – парний)

Позиції 9-12: код, який формується за таким правилом: кожній даті народження особи при визначенні УНЗР присвоюється код. Даті, яка зустрічається вперше: для чоловіків – 0001, для жінок – 0002; вдруге: для чоловіків – 0003, для жінок – 0004 і т. д.

III. Контрольна цифра

Для розрахунку контрольної цифри використовується метод розрахунку контрольних цифр у машинозчитуваній зоні паспортів (Пункт 16, Розділ IV Технічні специфікації, частини I, тому I DOC 9303 "Машинозчитувані проїзні документи" Міжнародна організація цивільної авіації (ICAO)).

Далі наведено порівняльну табл. 2 унікальних ідентифікаторів розглянених держав Євросоюзу.

3. ВИПУСК РЕЄСТРАЦІЙНИХ ДАНИХ ЯК СКЛАДОВА ПРОЦЕСУ УПРАВЛІННЯ

Бельгія

Згідно з [8], процедура отримання eID-картки виглядає так:

0: Громадянин отримує повістку на отримання картки, або сам проявляє ініціативу.

1: Відвідує муніципалітет з фотографією. (На 2009 р. у Бельгії було 589 муніципалітетів.)

2: Підписує запит на отримання eID.

3,4: CP (Card Personalizer) отримує eID запит через RRN.

Таблиця 2

Порівняльна характеристика унікальних ідентифікаторів країн ЄС

	Австрія	Естонія	Іспанія	Бельгія
Тип ID	a unique ID	National ID number	National ID number	National ID number
Назва ID	source PIN	PIC	TIN	RRN
Реєстр	Реєстр населення (головний процесор МВД)	Основний, додатковий	Поліція	Реєстр населення, реєстр іноземних громадян і реєстр очікування
Ким видається	Central Register of Residence	Державні органи, які вперше документують особистість	Міністерством внутрішніх справ	Federal Police
Коли видається	?	при народженні або при видачі посвідчення на проживання або право на проживання	?	При першому звернені в National Register
Довжина коду	24	11	9	11
Змістовність	букви, цифри	цифри	букви, цифри	цифри
Приклад коду	QQ03DPRGCHSX3G0LKSH6SQ==	99999999999	X9999999L	99999999999
Необхідні дані	CRR Number	дата народження, стать, порядковий номер осіб, які народилися в той же день	дата народження, громадянство	дата народження, стать, порядковий номер осіб, які народилися в той самий день

5: CP друкує нову eID картку, CI (Card Initializer) генерує пару ключів.

6: RRN отримує частину коду активації картки PUK1.

7: CA (Certification Authority) отримує запит на сертифікат.

8: CA випускає 2 нових сертифіката і новий CRLs (Certificate Revocation Lists).

9: CI зберігає ці сертифікати на eID картці.

10: CI записує дані громадянина (ID, адреса, ...) на картку і деактивує її.

11: CI відправляє лист-запрошення з PIN та кодом активації PUK2.

12: Громадянин отримує цей лист.

13: Державний службовець активує картку.

14: eID картка формує підпис, виходять з даних кожного закритого ключа, CA видаляє сертифікати з CRL.

Естонія

Процес отримання електронного посвідчення виглядає таким чином [7]:

1. Заява: персонально, поштою або електронною поштою заповнюється громадянином.

2. Запит персоналізації.

3. Запит сертифікатів.

4. Сертифікати згенеровано.

5. ID-картка з сертифікатами та ПІН-конвертом відправляється кур'єром.

6. Персоналізована ID-картка з сертифікатами та ПІН-конверт передається в офіс.

Україна

Згідно з [6] оформлення та видача паспорта здійснюються територіальним підрозділом ДМС за місцем проживання особи в Україні у взаємодії з Головним обчислювальним центром Єдиного державного демографічного реєстру (далі — Головний обчислювальний центр Реєстру) та Державним центром персоналізації документів (далі — Центр).

Порядок взаємодії територіальних підрозділів ДМС з Центром встановлюється шляхом укладення договору згідно із статтею 4 Закону України "Про захист персональних даних".

Внесення інформації до Реєстру здійснюється з використанням відомчої інформаційної системи ДМС, яка ведеться для обліку даних. Для внесення інформації до Реєстру формується заява-анкета.

Керівник територіального підрозділу ДМС визначає працівників, які формують заяву-анкету та на яких згідно з їх службовими обов'язками покладаються функції з оформлення та видачі паспорта.

Зазначені працівники під час виконання функцій з оформлення та видачі паспорта у день звернення надсилають в електронному вигляді захищеними каналами зв'язку запити до Мін'юсту щодо одержання відомостей з Державного реєстру

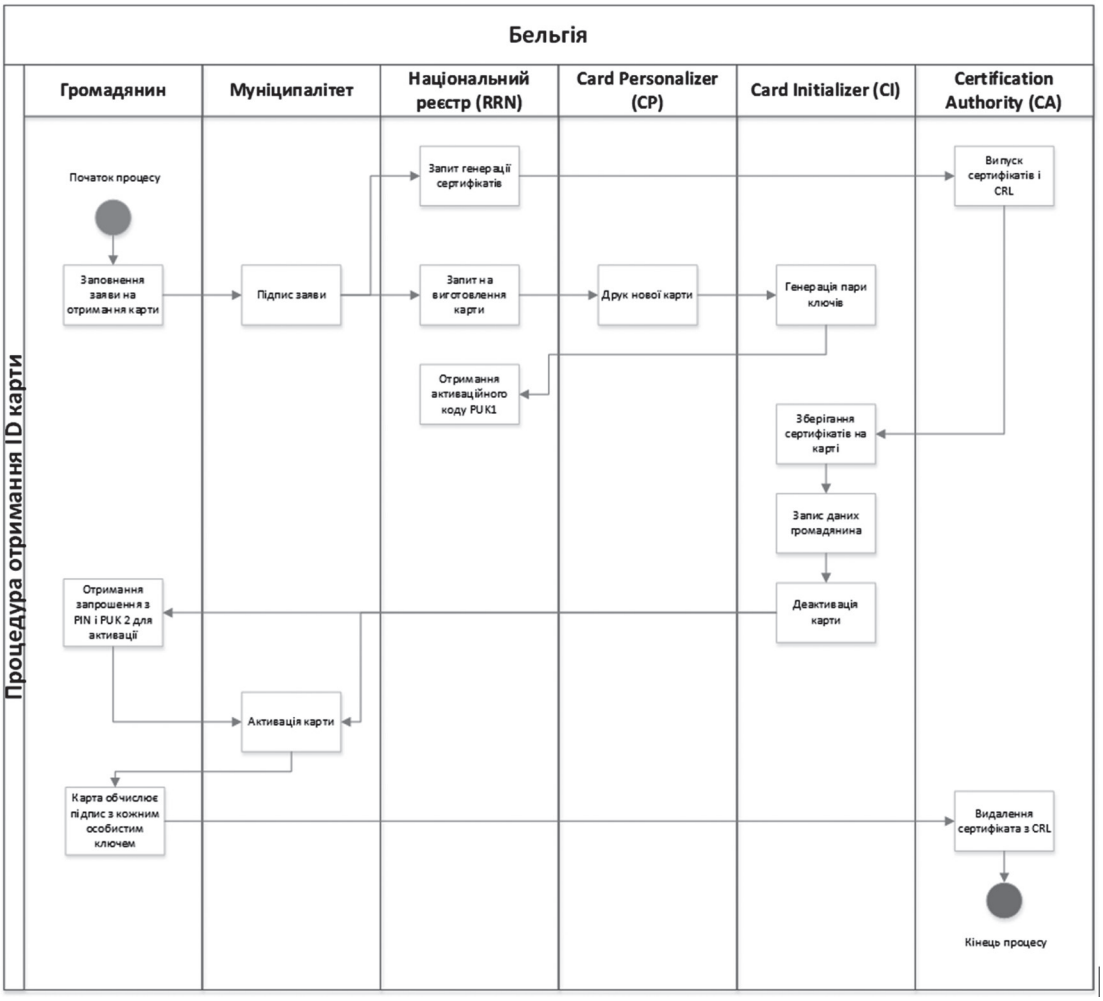


Рис. 3. Процес отримання електронного посвідчення в Бельгії

актів цивільного стану громадян про народження фізичної особи та її походження, усиновлення, позбавлення та поновлення батьківських прав, шлюб, розірвання шлюбу, зміну прізвища, власного імені, по батькові (далі — ім'я), смерть, які підлягають обов'язковому внесенню до Реєстру. Мін'юст надає зазначені відомості протягом трьох робочих днів з дня надходження такого запиту.

Головний обчислювальний центр Реєстру проводить автоматизовану перевірку, накопичення та збереження даних, що отримані від відомчої інформаційної системи ДМС, установлює їх унікальність, формує набори даних для виготовлення паспорта та передає їх до Центру, який здійснює персоналізацію бланка паспорта. У разі оформлення паспорта, що містить безконтактний електронний носій, Центр вносить до безконтактного електронного носія інформацію, що міститься на лицьовому та зворотному боці паспорта, біометричні параметри особи, додаткову змінну інформацію, а також дані про забезпечення захисту інформації, що міститься на безконтактному електронному носії, відповідно до рекомендацій IKAO Doc 9303.

Персоналізовані Центром бланки паспорта надсилаються до територіального підрозділу ДМС в установленому законодавством порядку, який видає ці паспорти громадянам.

4. РОЗРОБКА РЕКОМЕНДАЦІЙ ДЛЯ УКРАЇНСЬКОЇ ІНФРАСТРУКТУРИ ЕЛЕКТРОННОЇ ІДЕНТИФІКАЦІЇ

Проаналізувавши досвід інших країн та їх специфіку, для української моделі електронної ідентифікації розроблені такі рекомендації:

Необхідні дані для унікального ідентифікатора: дата народження, стать та порядковий номер осіб, які народилися в той самий день. (Як в Естонії та Бельгії). Можливо, доцільно використовувати іспанську модель як складову частину формування, бо вона дозволяє розмежувати іноземців та громадян країни. Але доцільним це є у випадку, якщо система, яка розробляється, надаватиме електронні посвідчення не тільки громадянам країни.

2. В моделі України не визначено що саме входить в дані щодо забезпечення захисту інформації. З досвіду країн Євросоюзу доцільно буде внести туди Сертифікат цифрового підпису та Сертифікат аутентифікації (наприклад, як в Естонії, разом з адресою е-пошти ім'я.прізвище @eesti.ee).

2. Щодо процесу видачі електронних посвідчень, то можна визначати обов'язкові компоненти цієї моделі: громадянин, муніципалітет, національній реєстр, виробник карток та сертифікаційний центр.

ВИСНОВКИ

Результатом аналізу є формулювання рекомендацій щодо вибору складу ідентифікаційних даних та алгоритму формування на їх основі унікального ідентифікатора. Також пропонується модель випуску реєстраційних даних, необхідні ключові об'єкти. Результати досліджень можуть бути застосовані для побудови процесів видачі ідентифікаційних даних (на відповідних носіях, наприклад, картках) фізичним особам. Також, аналіз досвіду країн ЄС дозволить уникнути помилок під час побудови інфраструктури електронної ідентифікації.

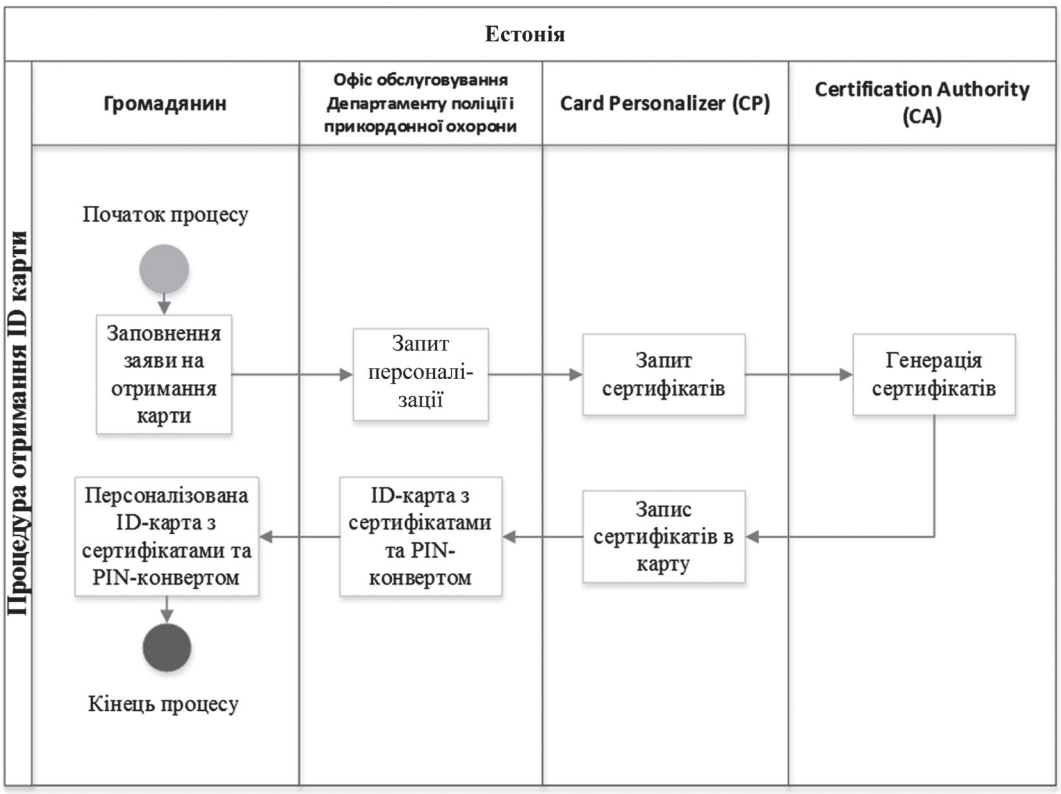


Рис. 4. Процес отримання електронного посвідчення в Естонії

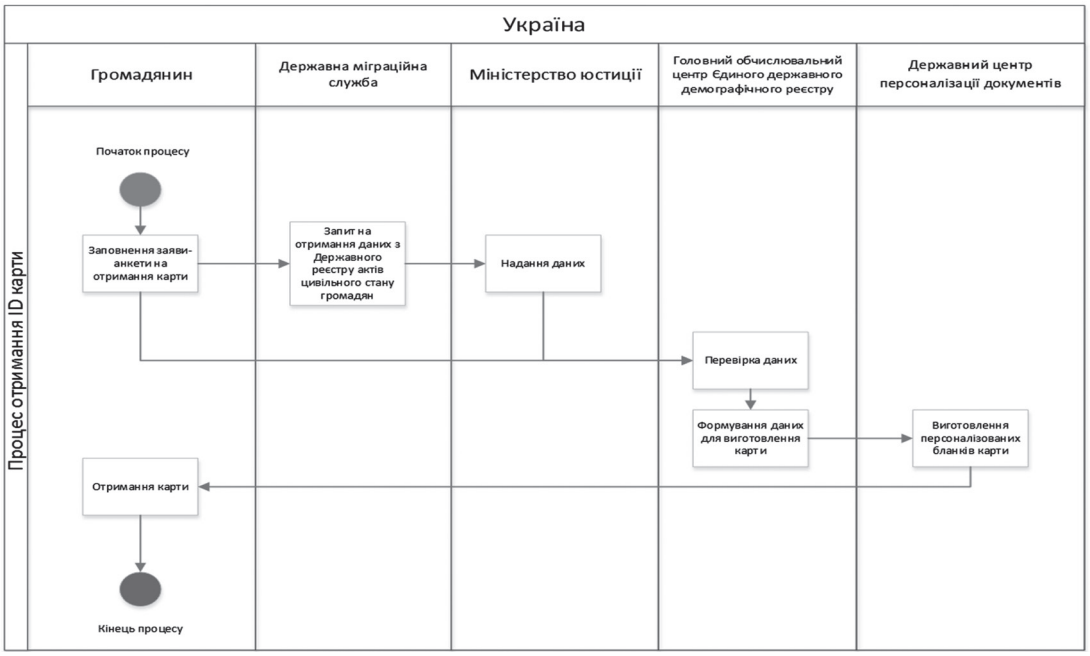


Рис. 5. Процес отримання електронного посвідчення в Україні

Література

[1] *О. Потій, Ю. Козлов.* Національная стратегія електронної ідентифікації України. Біла книга з електронного урядування, Березень 2015

[2] *Cock D.* Introducing Belgian eID cards, July 2013

[3] *Garcia J., Encinas H., Dios A.* The New Spanish Electronic Identity Card: DNI-e, Proceedings of the International Conference on Information Technologies (InfoTech-2007), September 21-23, 2007, Bulgaria

[4] *Eibl, Herwig, Karning, Kustor, Ledinger, Leitold, Medimorec, Pirker, Reichstädter, Rupp, Scheidbach, Wagner-Leimbach.* Administration on the Net. The ABC guide of eGovernment in Austria, September 2011, Vienna

[5] eID Estonian experience, May 2014, E Governance Academy

[6] Постанова КМУ 302 від 23.03.2015 «Порядок оформлення ID-card»

[7] Наказ МВС 1279 України від 26.11.2014 «Про затвердження зразка заяви-анкети для внесення інформації до Єдиного державного демографічного реєстру»

[8] *Danny De Cock.* Belgian eID Card Technicalities, <http://www.slideshare.net/beires/belgianeidcard-technicaloverview>

[9] *Педак М., Роосик Т.* Передовой опыт в области цифрового идентитета и документов удостоверяющих личность, Январь, 2015, Киев.

Надійшла до редколегії 30.10.2015

Потій Олександр Володимирович, фото та відомості про автора див. на с. 368.

Брошеван Євгенія Вікторівна, студентка кафедри комп'ютерних систем та мереж Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут». Наукові інтереси: системи електронної ідентифікації.

УДК 681.3.06(07)

Методы формирования уникального электронного идентификатора и процесс выдачи электронных удостоверений: опыт Евросоюза / А.В. Потий, Е.В. Брошеван // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 370–376.

Рассматривается функционально-прикладной уровень функциональной модели электронной идентификации, которая была описана в Национальной стратегии электронной идентификации Украины. Анализируются подходы к формированию уникальных электронных идентификаторов, их алгоритмов и необходимых ключевых данных для этой цели, принимая во внимание опыт стран ЕС. Рассматриваются принципы выдачи электронных удостоверений, роли каждого из участников этого процесса. А также, на основании положений Национальной стратегии, предлагаются рекомендации по соответствующим процессам для внедрения в Украине.

Ключевые слова: электронная идентификация, электронное правительство, регистрационные данные, уникальный идентификатор.

Табл.: 1. Рис.: 5. Библиогр.: 9 назв.

UDC 681.3.06(07)

Unique electronic identifier formation methods and eID card issuing procedure: EU experience / O.V. Potii, E.V. Broshevan // Applied Radio Electronics: Sci. Journ. – 2015. – Vol. 14. – № 4. – P. 370–376.

This paper reviews the functional-application layer of an electronic identification infrastructure functional model, which has been described in the National Strategy of Ukrainian Electronic Identification. Approaches to the formation of unique electronic identifiers, their algorithms and necessary key data for this purpose are analyzed taking into consideration the EU experience. The principles of eID card issuing procedure, key entities and the role of each participant in this process are considered. Besides, recommendations for appropriate unique identifiers and processes to be implemented in Ukraine are offered based on the National strategy points.

Keywords: electronic identification, electronic government, registration data, unique identifier.

Fig.: 5. Ref.: 9 items.

УДК 681.3.06(07)

ПРОТОКОЛИ НАДАННЯ ДОВІРЧИХ ПОСЛУГ У НАЦІОНАЛЬНИХ АРХІТЕКТУРАХ ЕЛЕКТРОННОЇ ІДЕНТИФІКАЦІЇ КРАЇН ЄВРОСОЮЗУ

О.В. ПОТІЙ, К.О. ПІДДУБНИЙ

У роботі розглядаються типові протоколи ідентифікації та аутентифікації, що реалізовані в Австрії, Естонії та Росії. Вивчення протоколів є важливим елементом попереднього аналізу для розробки протоколів, що будуть використані в Україні для надання електронних довірчих послуг.

Ключові слова: електронна ідентифікація, електронне урядування, протоколи надання довірчих послуг, архітектури електронної ідентифікації.

ВСТУП

Протягом років незалежності України різними урядами країни та каденціями органів місцевого самоврядування здійснювались заходи щодо створення тих або інших інформаційних систем загальнодержавного або місцевого масштабу для надання довірчих послуг в електронному вигляді.

Надання таких послуг передбачає он-лайн взаємодію між громадянином та органом, що надає послугу через певні інтерфейси інформаційної системи з використанням механізмів електронної ідентифікації та аутентифікації, які мають бути засновані на принципах безпеки та гарантій впевненості в ідентичності обох сторін такої взаємодії.

В цій статті проводиться аналіз протоколів надання довірчих електронних послуг на прикладі Австрії, Естонії та Росії. Ці країни мають багаторічний досвід в сфері електронної ідентифікації громадян та велику кількість документації, яка описує принцип роботи протоколів надання довірчих послуг.

1. ПОРЯДОК АУТЕНТИФІКАЦІЇ ТА ІДЕНТИФІКАЦІЇ ГРОМАДЯН В АВСТРІЇ

В Австрії всі громадяни зареєстровані в Central Register of Residence (CRR) [1] і мають свій унікальний номер (рис. 1). Зашифрований унікальний номер називається унікальним ідентифікатором (SourcePIN). SourcePIN і сертифікат зберігаються на картці громадянина у форматі XML [2]. Ця структура даних називається Identity Link і підписана SourcePIN Register Authority. Підпис встановлює і засвідчує зв'язок між ідентифікаційними даними і сертифікатом.

Постійне зберігання унікального ідентифікатора допускається тільки на самій картці. Для

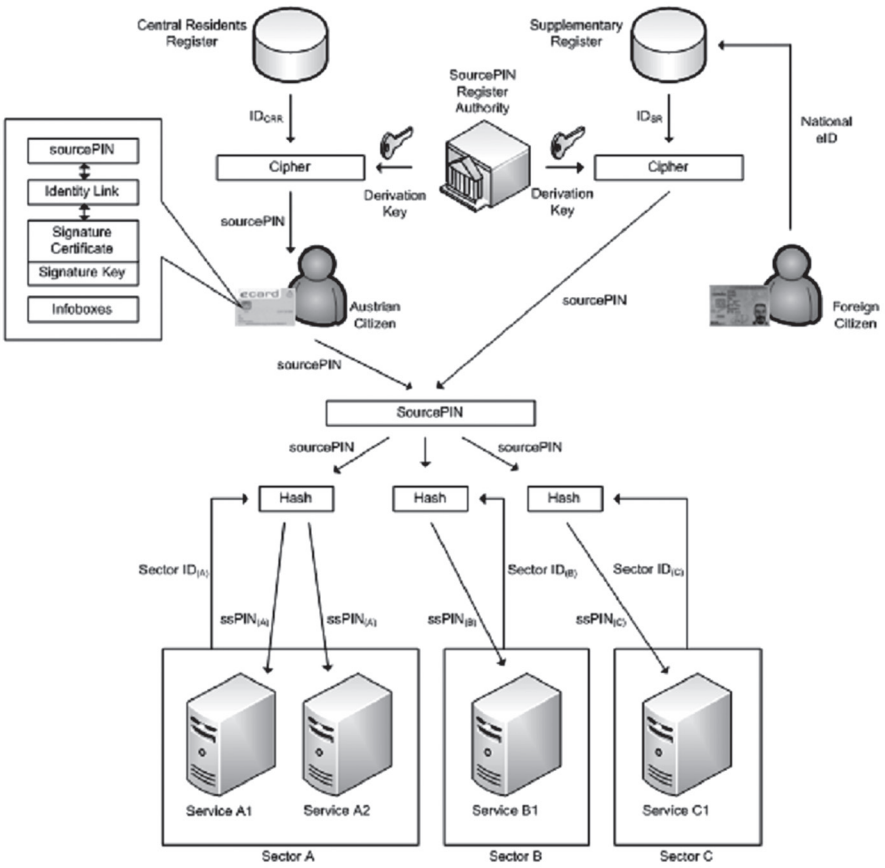


Рис. 1. Принцип ідентифікації громадян в Австрії

ідентифікації на сайті додаткам заборонено використовувати SourcePIN безпосередньо. Додатки використовують односторонню хеш-функцію за SourcePIN і державним ідентифікатором сектора. Тобто за отриманою хеш-функцією неможливо розрахувати унікальний ідентифікатор громадянина. Крім того, ssPIN різних секторів (наприклад, фінансового та податкового) відрізняється.

Концепція Австрійського eID передбачає використання електронних послуг особами, які не записанимі в CRR (наприклад, іноземним громадянам). Ці люди мають бути зареєстровані в додатковому реєстрі – Supplementary Register for Natural Persons (SR). Після чого вони отримують ті ж права, що і громадяни Австрії. Рисунок 1 ілюструє найбільш важливі етапи процесу ідентифікації. На рис. 2 зображена діаграма послідовності, що показує відповідні кроки під час ідентифікації та аутентифікації австрійських громадян в онлайн-додатках.

Безпечна ідентифікація громадянина складається з таких кроків:

1. Громадянин хоче отримати доступ до додатка постачальника послуг. Додаток виконує перевірку доступу і в разі відмови перенаправляє користувача в MOA-ID [3] для ідентифікації і аутентифікації.

2. MOA-ID повертає HTML форму, яка включає в себе запит XML для зчитування Indetity Link (InfoBoxReadRequest). Цей запит відправляється до програмного забезпечення картки громадянина (CCS).

3. CCS зчитує Identity Link і повертає його в MOA-ID.

4. MOA-ID перевіряє справжність Identity Link підписаної SourcePIN Register Authority. На цьому етапі громадянин успішно ідентифікується.

5. Ґрунтуючись на інформації, отриманій від Identity Link, MOA-ID відправляє XML запит для створення електронного підпису в CCS.

6. Дані, які громадянин хоче підписати, включені в XML запит. На цьому кроці CCS повертає підписані дані в MOA-ID.

7. MOA-ID перевіряє підписані дані, створює твердження SAML та перенаправляє громадянина в онлайн-додаток. Це перенаправлення також включає артефакт SAML [4]. Покажчик на затвердження SAML тимчасово зберігається в MOA-ID.

8. Онлайн-додаток використовує артефакт SAML як еталон для збору твердження SAML з MOA-ID.

9. На основі інформації з твердження SAML додаток визначає, чи отримає громадянин доступ. І, нарешті, громадянин буде перенаправлений на захищений ресурс.

Ідентифікація та аутентифікація іноземних користувачів може бути здійснена за допомогою двох технологій:

SourcePIN Register Gateway;

STORK.

На рис. 3, 4 зображені діаграми послідовності, що показують відповідні кроки під час ідентифікації та аутентифікації іноземних громадян

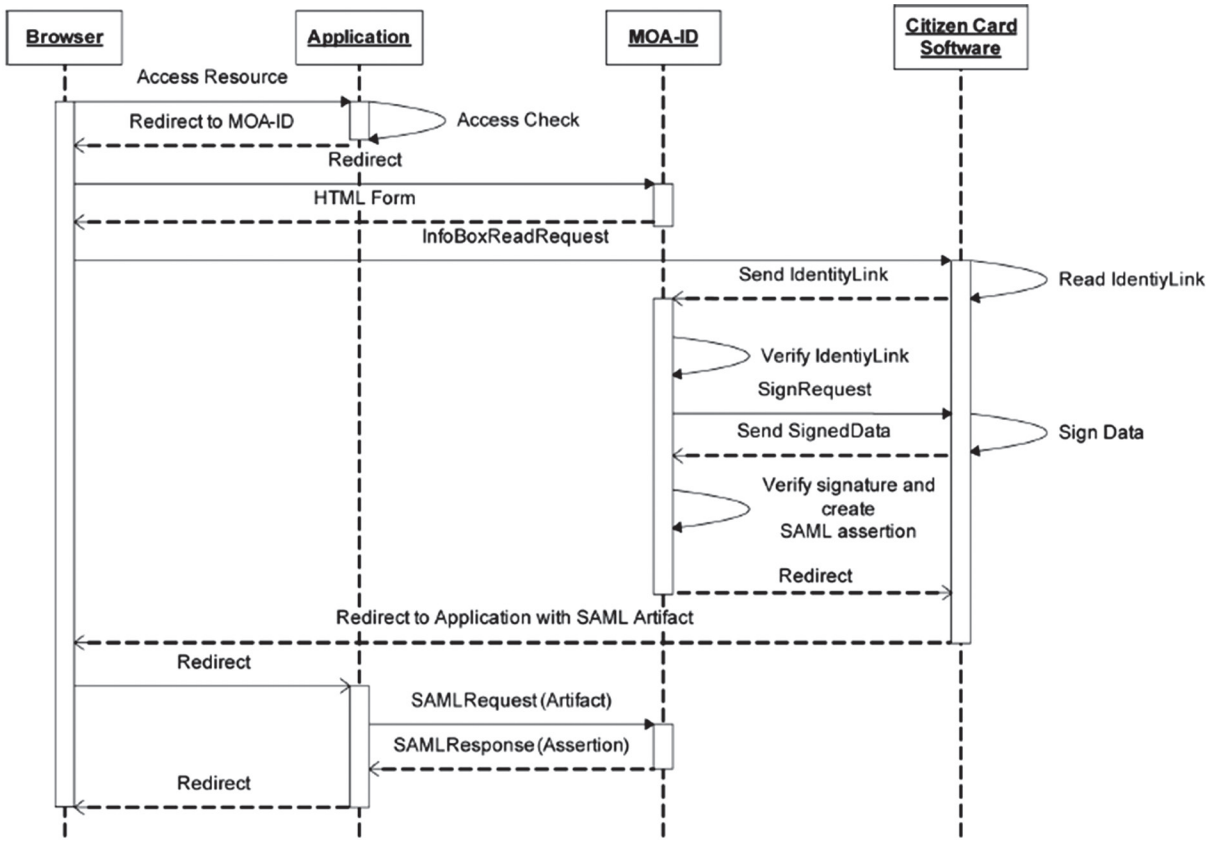


Рис. 2. Діаграма послідовності кроків під час ідентифікації та аутентифікації австрійських громадян в онлайн-додатках

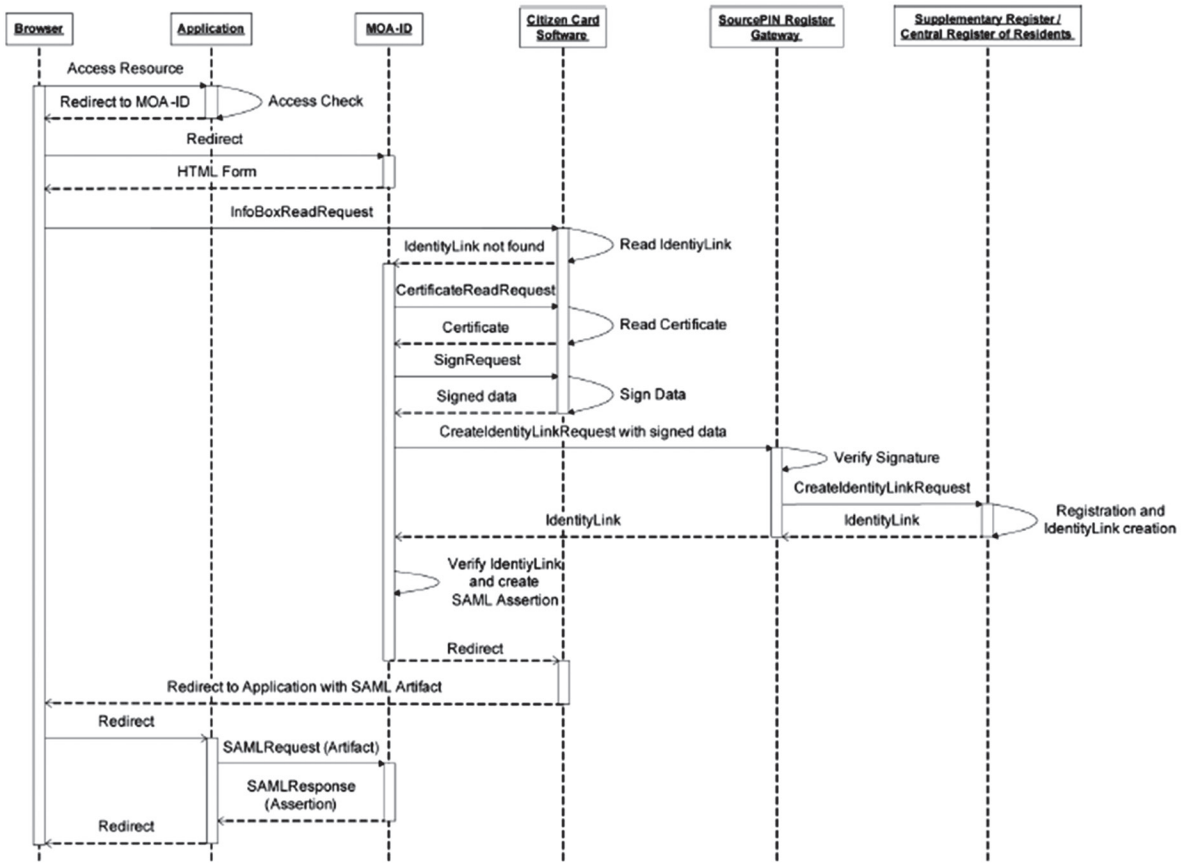


Рис. 3. Діаграма послідовності кроків під час ідентифікації та аутентифікації іноземних громадян в онлайн-додатках за допомогою SourcePIN Register Gateway

в онлайн-додатках за допомогою SourcePIN Register Gateway та STORK відповідно.

1. Початок процесу аутентифікації еквівалентний крокам 1–2 для громадян Австрії.

2. CCS намагається прочитати Identity Link, але через використання іноземного eID процес зчитування закінчується з помилкою. Повідомлення про помилку відправляється назад в MOA-ID.

3. Замість читання Identity Link, MOA-ID створює XML запит з метою отримання сертифіката від іноземного eID. Цей запит відправляється до CCS. CCS зчитує сертифікат і повертає його в MOA-ID.

4. На основі сертифікату MOA-ID відправляє XML запит для створення електронного підпису до CCS.

5. CCS повертає підписані дані в MOA-ID.

6. MOA-ID створює XML запит, який включає підписані дані, і відправляє його в SourcePIN Register Gateway.

7. SourcePIN Register Gateway перевіряє підпис і перенаправляє запит в SourcePIN Register.

8. SourcePIN Register здійснює пошук іноземного громадянина в Central Register of Residents і в Supplementary Register for Natural Persons. В результаті пошуку можливі два варіанти:

• іноземний громадянин вже зареєстрований в одному з двох реєстрів. Таким чином, Identity Link може бути створена і передана в SourcePIN Register Gateway;

• іноземний громадянин не зареєстрований в одному з реєстрів. У цьому випадку, громадянин автоматично реєструється в додатковому реєстрі Supplementary Register for Natural Persons. Після завершення процесу реєстрації Identity Link може бути створена і передана в SourcePIN Register Gateway.

9. SourcePIN Register Gateway приймає Identity Link і перенаправляє її в MOA-ID.

10. MOA-ID перевіряє електронний підпис Identity Link і створює SAML твердження. Після чого перенаправляє громадянина в онлайн-додаток.

11. Наступні кроки еквівалентні крокам 8–9 з процесу аутентифікації австрійських громадян.

1. Початок процесу аутентифікації еквівалентний першому кроку для громадян Австрії.

2. На цьому етапі громадянин обирає свою країну. MOA-ID відправляє запит на аутентифікацію до STORK.

3. STORK з'єднується з національним провайдером ідентифікації, щоб отримати твердження SAML 2.0. Провайдер використовує національну інфраструктуру eID.

4. Провайдер ідентифікації повертає SAML 2.0 твердження в STORK. STORK відправляє його назад в MOA-ID.

5. MOA-ID створює запит XML на основі отриманих з SAML 2.0 даних твердження і передає цей запит в SourcePIN Register Gateway.

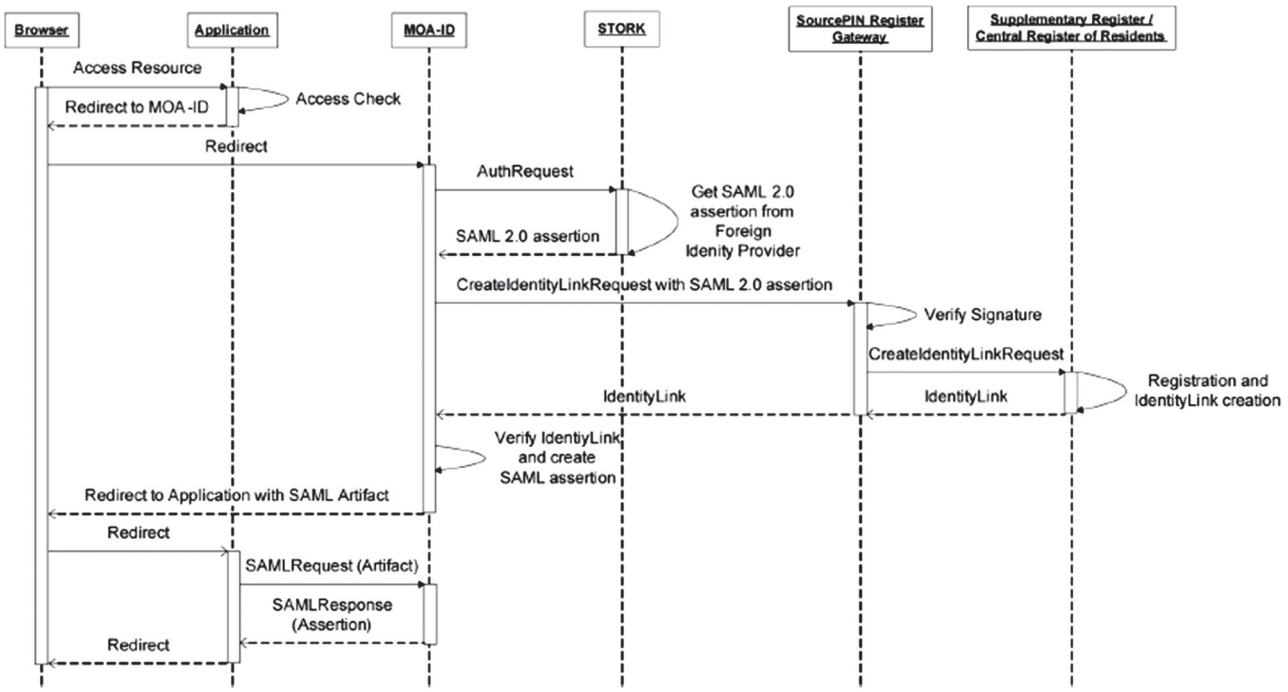


Рис. 4. Діаграма послідовності кроків під час ідентифікації та аутентифікації іноземних громадян в онлайн-додатках за допомогою STORK

6. SourcePIN Register Gateway перевіряє підпис громадянина і перенаправляє запит до SourcePIN Register.

7. Наступні кроки еквівалентні крокам 9–11 з використанням SourcePIN Register Gateway.

2. ПОРЯДОК АУТЕНТИФІКАЦІЇ ТА ІДЕНТИФІКАЦІЇ ГРОМАДЯН В ЕСТОНІЇ

В Естонії використовується технічне й організаційне середовище X-Road [5], яке дозволяє безпечно обмінюватися даними з інформаційними системами держави. На рис. 5 показана діаграма послідовності кроків під час передачі даних в середовищі X-Road.

Всі інформаційні системи підприємств і установ державного та приватного секторів можна підключити до X-Road, де вони зможуть запропонувати свої електронні послуги.

X-Road дозволяє громадянам використовувати державні бази даних для своїх цілей і контролювати інформацію, що відноситься до них.

Чиновники можуть використовувати сервіси призначені для них (наприклад, пункт обміну документами) в інформаційних системах власних установ. Це полегшує роботу посадових осіб, оскільки це дозволяє уникнути трудомісткої задачі обробки паперових документів. Це означає, що зв'язок з іншими посадовими особами, підприємцями та громадянами є швидшим і набагато точнішим. На рис. 6 показано естонську інформаційну систему.

Естонія не має офіційної політики, що визначає ієрархію різних методів електронної аутентифікації. Тим не менш, переважним є метод аутентифікації на інфраструктурі відкритих ключів (PKI).

Аутентифікація у веб-середовищі здійснюється з використанням стандартних протоколів TLS та SSL з перевіркою достовірності клієнта. Протокол передбачає, що постачальник послуг отримає повний сертифікат користувача з персональним ідентифікаційним кодом (PIC).

1. Користувач аутентифікується та ідентифікується за допомогою ID-картки, пароля, біометричних даних.

2. Інформаційна система дає користувачеві доступ до методів, які користувач має право використовувати.

3. Користувач вибирає потрібний метод.

4. Повідомлення про виклик методу відправляється до серверу безпеки інформаційної системи. Повідомлення містить заголовок з персональним кодом користувача, назву інформаційної системи, унікальний ідентифікатор повідомлення і т. д.

5. Сервер безпеки підписує повідомлення закритим ключем і через DNSSEC (Domain Name System Security Extensions) запитує у центрального сервера IP-адрес сервера(ів) безпеки БД.

6. Сервер безпеки інформаційної системи відкриває TCP з'єднання з сервером безпеки БД і відправляє свій сертифікат для використання протоколу TLS (Transport Layer Security).

7. Сервер безпеки БД перевіряє через DNSSEC сертифікат, отриманий від сервера безпеки інформаційної системи. Якщо сертифікат є дійсним, сервер безпеки БД посилає свій сертифікат, щоб закінчити створення безпечного з'єднання.

8. Сервер безпеки інформаційної системи перевіряє через DNSSEC сертифікат, отриманий від сервера безпеки БД.

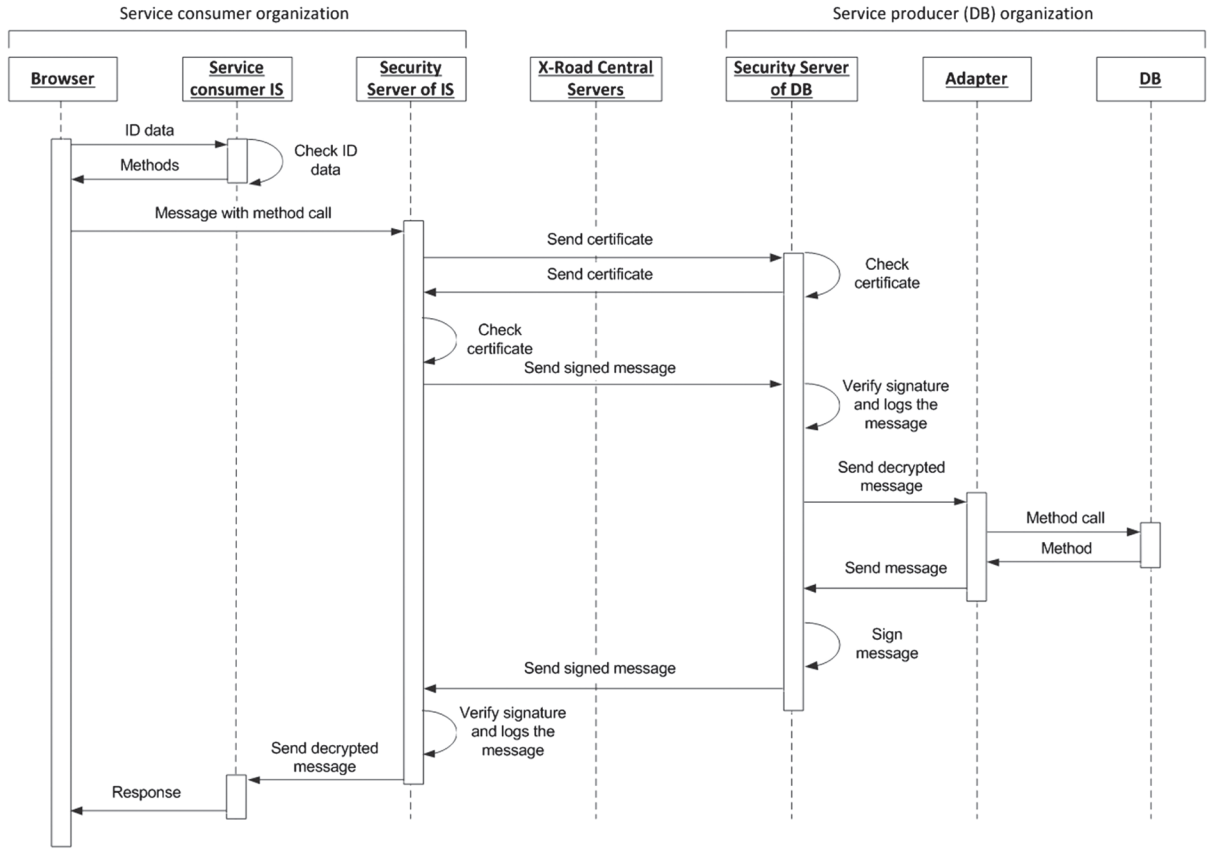


Рис. 5. Діаграма послідовності кроків під час передачі даних в середовищі X-Road

9. Після створення безпечного каналу сервер безпеки інформаційної системи посилає підписане повідомлення на сервер безпеки БД.

10. Сервер безпеки БД перевіряє підпис повідомлення і реєструє його. Потім перевіряє, чи є інформаційна система дозволеною для цього методу і відправляє розшифроване повідомлення адаптеру.

11. Серверний адаптер викликає метод у базі даних.

12. Сервер безпеки БД підписує повідомлення у відповідь і посилає його серверу безпеки інформаційної системи.

13. Сервер безпеки інформаційної системи перевіряє підпис повідомлення і реєструє його, після чого посилає розшифроване повідомлення інформаційній системі.

14. Користувач отримує відповідь.

Обидва сервери безпеки гешують свої журнали і періодично відправляють їх геш до центральних серверів.

3. ПОРЯДОК АУТЕНТИФІКАЦІЇ ТА ІДЕНТИФІКАЦІЇ ГРОМАДЯН У РОСІЇ

Розробники Російських сайтів, порталів та інших веб-додатків надають своїм користувачам можливість входити в систему, використовуючи обліковий запис ЄСІА [6]. Це позбавляє розробників від необхідності робити власне сховище облікових записів, забезпечувати безпеку зберігання паролів, розробляти механізми реєстрації, аутентифікації користувачів, підтримувати їх у робочому стані.

В ЄСІА реалізовано два альтернативних механізми, що дозволяють розробнику використовувати найбільш підходящий для його системи:

– механізм, заснований на стандарті SAML версії 2.0;

– механізм, заснований на моделі OpenID Connect 1.0.

Перший механізм включає такі кроки (рис. 6):

1. Користувач натискає на сторінці системи постачальника послуг кнопку «Увійти через ЄСІА».

2. Постачальник послуг формує і відправляє в ЄСІА запит на аутентифікацію і перенаправляє браузер користувача на сторінку аутентифікації ЄСІА.

3. ЄСІА перевіряє статус аутентифікації користувача. Якщо користувач в ЄСІА НЕ аутентифікований, то для продовження процесу він повинен пройти аутентифікацію одним з доступних способів. Якщо користувач ще не зареєстрований у ЄСІА, то він може перейти до процесу реєстрації.

4. Коли користувач аутентифікований, ЄСІА перевіряє, що рівень достовірності ідентифікації користувача відповідає вимогам системи, які зафіксовані в метаданих.

5. Коли користувач успішно аутентифікований, ЄСІА передає в систему відповідь на запит аутентифікації, який містить набір тверджень SAML про користувача.

6. Постачальник послуг приймає рішення про авторизацію користувача на основі отриманої з ЄСІА інформації.

Рис. 6. Ідентифікація та аутентифікація користувачів за допомогою ЄСІА при використанні SAML 2.0

Механізм, заснований на моделі OpenID включає такі кроки (рис. 7):

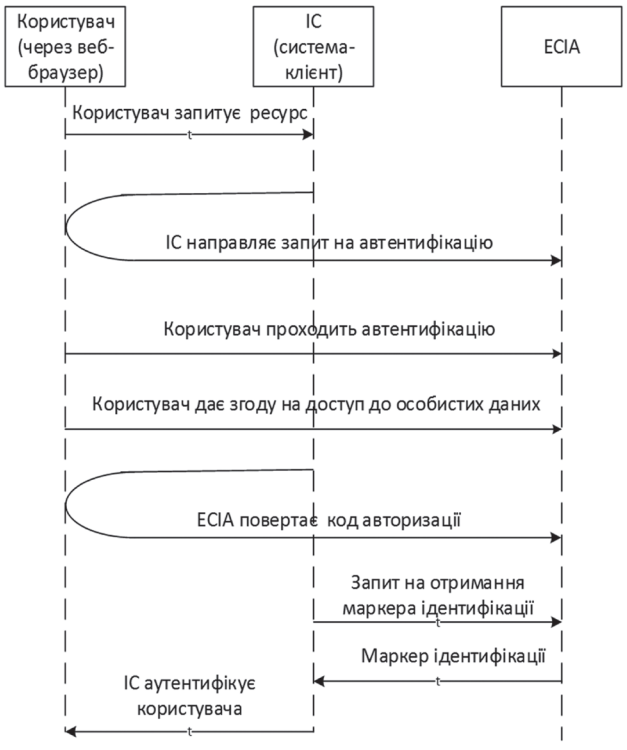


Рис. 7. Ідентифікація та аутентифікація користувачів при використанні механізму OpenID Connect 1.0

1. Користувач натискає на веб-сторінці системи-клієнта кнопку «Увійти через ЄСІА»

2. Система-клієнт формує і відправляє в ЄСІА запит на аутентифікацію і перенаправляє браузер користувача на спеціальну сторінку надання доступу.

3. ЄСІА здійснює аутентифікацію користувача одним з доступних способів. Якщо користувач ще не зареєстрований у ЄСІА, то він може перейти до процесу реєстрації.

4. Коли користувач аутентифікований, ЄСІА повідомляє користувачеві, що система-клієнт запитує дані про нього з метою проведення ідентифікації і аутентифікації, надаючи перелік запитуваних системою-клієнтом відомостей.

5. Якщо користувач дає дозвіл на проведення аутентифікації системою-клієнтом, то ЄСІА видає системі-клієнту спеціальний авторизаційний код.

6. Система-клієнт формує на адресу ЄСІА запит на отримання маркера ідентифікації, включаючи в запит отриманий раніше авторизаційний код.

7. ЄСІА перевіряє коректність запиту (наприклад, що система-клієнт зареєстрована в ЄСІА) і авторизаційного коду і передає системі-клієнту маркер ідентифікації.

8. Система-клієнт отримує ідентифікатор користувача з маркера ідентифікації. Якщо ідентифікатор отриманий, а маркер перевірений, то система-клієнт вважає користувача аутентифікованим.

4. АНАЛІЗ ПРОТОКОЛІВ НАДАННЯ ДОВІРЧИХ ПОСЛУГ

Аутентифікація у веб-середовищі в Австрії, Естонії та Росії здійснюється за допомогою стандартних протоколів TLS та SSL з перевіркою достовірності клієнта, але кроки під час аутентифікації та ідентифікації громадянина суттєво відрізняються.

В Австрії використовується модуль для інтернет-додатків (MOA), який є набором програмних інструментів. Ці інструменти реалізують функції створення та перевірки електронного цифрового підпису, ідентифікації і аутентифікації громадян.

Архітектура електронної ідентифікації Естонії передбачає використання технічного середовища X-Road, яке дозволяє безпечно обмінюватися даними з інформаційними системами.

У Росії функції ідентифікації та аутентифікації користувачів забезпечує Єдина система ідентифікації та аутентифікації.

ВИСНОВКИ

У доповіді проаналізовано типові протоколи ідентифікації, що реалізовані в Австрії, Естонії та Росії. Вивчення протоколів є важливим елементом попереднього аналізу, для розробки протоколів, що будуть використані в Україні для надання електронних довірчих послуг. Основною складністю є те, що необхідно забезпечити доступ до багатьох систем, що мають різні архітектури, програмне забезпечення тощо. Тому прийняття базової моделі протоколу ідентифікації є актуальним.

Література

[1] Chapter 11 The Austrian Identity Ecosystem: An E-Government Experience (2014)

[2] World Wide Web Consortium. (2008). XML signature syntax and processing (2nd ed)

[3] Federal Chancellery of Austria. (2003). Module for online applications (MOA) – Identification (ID)

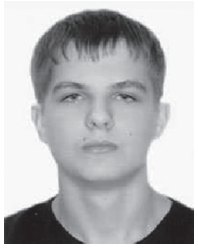
[4] OASIS. (2012). Security assertion markup language (SAML), OASIS security services (SAML) TC

[5] Aleksander Reitsakas: X-Road (2015)

[6] Единая система идентификации и аутентификаци. Версия 2.3 (2014.)

Надійшла у редколегію 3.11.2015

Потій Олександр Володимирович, фото та відомості про автора див. на с. 368.



Піддубний Кирило Олександрович, студент кафедри комп'ютерних систем та мереж Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут». Наукові інтереси: системи електронної ідентифікації.

УДК 681.3.06(07)

Протоколы предоставления доверительных услуг в национальных архитектурах электронной идентификации стран Евросоюза / А.В. Потий, К.А. Поддубный // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 377–383.

В работе рассматриваются типичные протоколы идентификации и аутентификации, реализованные в Австрии, Эстонии и России. Изучение протоколов является важным элементом предварительного анализа для разработки Протоколов, которые будут использованы в Украине для предоставления электронных доверительных услуг.

Ключевые слова: электронная идентификация, электронное правительство, протоколы предоставления доверительных услуг, архитектуры электронной идентификации.

Рис.: 7. Библиогр.: 6 назв.

UDC 681.3.06(07)

Protocols of trust services in national architectures of electronic identification of the European Union / O.V. Potii, K.O. Piddubniy // Applied Radio Electronics: Sci. Journ. – 2015. – Vol. 14. – № 4. – P. 377–383.

The paper deals with the typical identification and authentication protocols implemented in Austria, Estonia and Russia. Studying the protocols is an important element of the preliminary analysis to develop protocols that will be used in Ukraine to provide electron-governmental trust services.

Keywords: electronic identification, electronic government, protocols of trust services, architectures of electronic identification.

Fig.: 7. Ref.: 6 items.

УДК 004.056:003.26

АНАЛІЗ ТА ПОРІВНЯЛЬНІ ДОСЛІДЖЕННЯ МЕТОДІВ МЕРЕЖНОЇ СТЕГАНОГРАФІЇ

О.О. КУЗНЄЦОВ, В.А. ТИМЧЕНКО, Б.С. ДУБРОВНИЙ

Розглядаються можливості стеганографічних методів захисту інформації, побудова яких заснована на використанні особливостей функціонування мережного стеку протоколів передачі даних сімейства TCP/IP (Transmission Control Protocol / Internet Protocol). Досліджуються різні методи мережної стеганографії, які засновані: на кодуванні вмісту інформаційних пакетів; на маніпуляціях потоку переданих пакетів; гібридні методи. Обґрунтовуються перспективні напрямки подальших досліджень із вдосконалення методів мережної стеганографії для їх використання з метою приховування факту існування скритої передачі інформації.

Ключові слова: мережна стеганографія, протоколи сімейства TCP/IP.

ВСТУП

У сучасному світі надзвичайно велику роль у виробничій, торгівельній, сервісній та інших сферах економіки та у житті людини відіграє інформація. Тому для забезпечення доступності, цілісності та конфіденційності інформації широко використовуються різноманітні криптографічні та стеганографічні методи захисту інформації [1, 2].

Нагадаємо, що криптографія – це наука, що вивчає методи та засоби для приховування значення змісту даних, що передаються [1, 2]. Стеганографія має на меті ті ж цілі, однак приховує при цьому сам факт існування таємного повідомлення в даних. Хоча методи стеганографії використовуються вже тривалий час, але завдяки розвитку комп'ютерної техніки і телекомунікаційних технологій, отримав стрімкий розвиток новий вид прихованої передачі даних – цифрова стеганографія [2], а з розвитком і вдосконаленням технологій передачі інформації комп'ютерними мережами, з'являються нові різноманітні методи непомітної передачі інформації – мережна стеганографія [3].

У мережній стеганографії роль носія виконує переданий по мережі пакет або сукупність пакетів даних [3, 4]. Це відкриває великі перспективи для тих, хто хоче непомітно передавати повідомлення через будь-які кордони і створює небезпеку для установ, що займаються захистом інформації від несанкціонованого витоку інформації. У загальному вигляді, мережна стеганографія є сімейством методів з модифікації даних у заголовках мережних протоколів і у полях корисного навантаження пакетів, зміни структури передачі пакетів та гібридних методів у тому чи іншому мережному протоколі [4].

Основні параметри мережної стеганографії – це пропускна здатність прихованого каналу, ймовірність виявлення і стеганографічна стійкість [4]. Пропускна здатність – обсяг секретних даних, який може бути відправлений за одиницю часу. Ймовірність виявлення визначається за можливістю виявлення стеганографії на певному носії. Найбільш популярний спосіб виявлення прихованого повідомлення – це аналіз статистичних властивостей отриманих даних і порівняння їх з типовими значеннями для цього носія. Стеганографічна стійкість характеризує ступінь зміни носія після впливу на нього стеганометоду.

Метою даної статті є аналіз методів прихованої передачі інформації з використанням мережної стеганографії, які є перспективними на сьогодні.

1. ПРОТОКОЛИ СІМЕЙСТВА TCP/IP

Термін «мережна стеганографія» уперше ввів Кжиштоф Джипйорскі (Krzysztof Szczypiorski) у 2003 році [3]. Хоча цей метод стеганографії з'явився відносно недавно, він має таку саму структуру стеганосистеми як і класична або цифрова стеганографія. Особливістю є те, що як контейнер, в якому приховано передається конфіденційне повідомлення, виступають певні комунікаційні компоненти: правило змін довжини пакетів або процедура перепопиту помилкових даних; службові поля пакетів даних, прапорці та ін. [3–11]. Розглянемо особливості побудови найбільш поширених мережних протоколів TCP/IP, які можуть бути застосовані для реалізації стеганографічних методів захисту інформації.

Стек сімейства протоколів TCP/IP є відкритим комунікаційним протоколом. Відкритість означає, що він забезпечує зв'язок у будь-яких комбінаціях пристроїв незалежно від того, наскільки вони різняться на фізичному рівні [6].

Переваги TCP/IP. Протокол TCP/IP забезпечує можливість міжплатформових мережних взаємодій. Наприклад, мережа під управлінням Windows може містити робочі станції Unix, і навіть інші мережі нижчого порядку. TCP/IP володіє такими характеристиками [6]:

– ефективні засоби відновлення після збоїв;

– можливість додавання нових мереж без переривання поточної роботи;

– стійкість до помилок;

– незалежність від платформи реалізації;

– низькі непродуктивні витрати на пересилання службових даних.

Протоколи TCP і IP спільно управляють потоками даних (як вхідними, так і вихідними)

у мережі. Але, якщо протокол IP тільки передає пакети, не звертаючи уваги на результат, TCP повинен простежити за тим, щоб пакети прибули в належне місце. Зокрема, TCP відповідає за виконання таких завдань [6]:

– відкриття та закриття сеансу;
– управління пакетами;
– управління потоком даних;
– виявлення та обробка помилок.

Розглянемо структуру заголовка IP версії 4, яку наведено на рис. 1 (на рис. 2 наведено структуру заголовка IP версії 6).

Поле версії. Визначає поточну версію IP, реалізовану мережною станцією.

Тип служби. Складається з таких полів: пріоритет, затримка, пропускна здатність і надійність [7].

Загальна довжина. Довжина дейтаграми (не пакета), виміряна в байтах (довжина цього поля становить 16 біт, отже, область даних IP-дейтаграми може мати довжину 65535 байт).

Поле ідентифікації. Показує, які фрагменти належать одній і тій самій дейтаграмі, щоб їх не переплутати. Приймаючий IP-рівень використовує це поле і IP-адреса джерела, щоб визначити, які фрагменти належать одній дейтаграмі [7].

Прапорці. Показує, чи мають ще надійти фрагменти або дані для поточної дейтаграми більше не передаватимуться (останній фрагмент), а також фрагментувати чи ні дейтаграму (біт відмови від фрагментації) [7].

Зсув фрагмента. Це поле використовується приймачем для зворотного збирання фрагментованою дейтаграмою.

Час життя. Визначає інтервал часу, протягом якого дейтаграмі дозволено перебувати у мережі.

На рис. 3 показано поля заголовка TCP у тому вигляді, у якому вони інкапсулюються в заголовок IP-дейтаграми [7]:

Порт-джерело. Номер порту передавальної станції.

Порт-приймач. Номер порту приймаючої станції.

Порядковий номер. Значення, присвоєне дейтаграмі TCP, визначає номер стартового байту пакета, якщо тільки не встановлений біт SYN. Якщо ж цей біт встановлений, то порядковий номер є початковим порядковим номером (ISN) і перший байт даних дорівнює ISN + 1.

Номер підтвердження. Значення, надіслане цільовою станцією станції відправника, яке підтверджує прийом переданого раніше пакета (пакетів).

Зсув даних. Визначає довжину заголовка TCP (тобто кількість 32-бітових слів у заголовку TCP).

Резерв. Зарезервовано для майбутнього використання.

Біти управління [7]:

URG – індикатор міцності, застосовується при посиланні повідомлення цільового вузла, що очікує прийом екстреної інформації.

ACK – якщо даний біт встановлений, отже, пакет містить підтвердження надісланої раніше дейтаграми.

PSH – функція виштовхування, негайне відсилання даних після зчитування сегмента (даних цього пакета).

RST – обрив з'єднання з метою відмови на запит з'єднання.

SYN – служить для ініціалізації та установки порядкового значення.

FIN – означає, що в ініціатора з'єднання даних більше немає.

Рис. 1

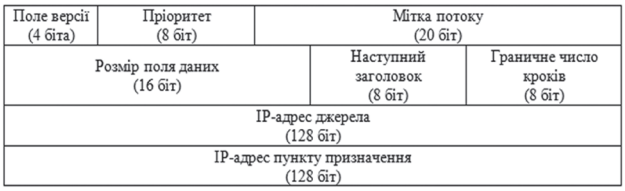


Рис. 2

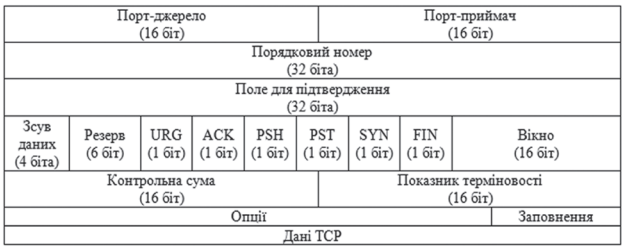


Рис. 3

Вікно. Кількість октет даних, починаючи з октету, зазначеного у полі підтверджень, які відправник сегмента може прийняти.

Контрольна сума. Значення, призначене для виявлення помилок.

Показник термінових даних. Задає порядковий номер байта, наступного за рядковими даними.

Опції. Поле змінної довжини, дозволяє використати такі опції TCP: кінець списку опцій, немає операції і максимальний розмір сегменту.

Розглянемо методи мережної стеганографії, які реалізуються з використанням особливостей функціонування протоколів TCP/IP, обґрунтуємо їх використання для приховування факту існування скритої передачі інформації.

2. АНАЛІЗ МЕТОДІВ TCP/IP СТЕГАНОГРАФІЇ

Методи мережної стеганографії можна розділити на три групи [4]:

– методи, що змінюють вміст інформаційних пакетів;

– методи, які засновані на маніпуляціях над потоком переданих пакетів;

– гібридні методи (об'єднання методів двох попередніх груп).

Типові методи мережної стеганографії включають зміну властивостей одного з мережних протоколів. Крім того, може використовуватися взаємозв'язок між двома або більше різними протоколами з метою більш надійного приховування факту передачі таємного повідомлення [8, 9].

Розглянемо більш докладно наведені методи мережної стеганографії, проведемо дослідження їх можливостей щодо прихованої передачі інформаційних повідомлень.

2.1. Методи модифікації вмісту інформаційних пакетів. Це найпростіша група методів, сутність яких полягає в записі інформаційного повідомлення у різні поля пакетів даних. Наприклад, у роботах [10] розглянуто *методи зміни даних у полях заголовків мережного протоколу*. Приклад подібного методу, заснований на модифікації полів заголовків протоколу IP, наведено на рис. 1 та на рис. 2. У даному методі розглядається модифікація заголовка IPv4 не використовуваних полів для створення прихованого каналу. Оскільки поле «Ідентифікатор» в IP заголовку може містити 16 біт інформації, в полі «Прапорці» доступний 1 біт, а в полі «Номер послідовності» в TCP заголовку доступний обсяг інформації в 32 біта можна зробити висновок, що загальна пропускна здатність стеганографії дорівнює 49 бітам. Але слід зазначити, що в даному методі ми використовуємо поле «Ідентифікатор» для передачі зашифрованого ключа в стеганограмі, який служить для вилучення секретної інформації з поля «Номер послідовності», а біт в полі «Прапорці» використовується як мітка.

Методи модифікації полів заголовків IP і TCP володіють певними особливостями, які виділяють їх на тлі інших методів [10]:

– як носії стеганограми використовуються найпоширеніші й стандартні протоколи;

– у сумі дають пропускну здатність 49 біт за 1 пакет;

– реалізуються на будь-якій операційній системі, реалізація не вимагає довгих налаштувань і підготовок;

– зміни в пакеті не вплинуть на його поведінку в мережі, у випадку, якщо він не буде фрагментований.

Розглянемо алгоритм вбудовування інформації в поле «Ідентифікатор» (див. рис. 4), де X (0 або 1) біти представляють ідентифікатор пакета IP і C (0 або 1) біти є секретним символом для вбудовування.

Алгоритм вбудовування. На вхід подаємо вибрані пакети та конфіденційну інформацію. Для початку прочитаємо символ впровадженого файла (наприклад, secret.txt), після зчитування перетворюємо символ в еквівалентне значення ASCII. Отриманий пакет на вході, зчитуємо значення 16-розрядного поля ідентифікації, скинувши молодші 8-розрядів (Tem = {Поле ID} AND {Маска 65280}) та додаємо секретний 8-бітний ASCII ({New IP ID} = Tem + Ch). Після закінчення повертаємося до пакета для ін'єкції з {New IP ID} області. На виході отримуємо стегано-пакет [10].

Рис. 4

Алгоритм виймання (див. рис. 5). На вході отримуємо стегано-пакет, з якого зчитуємо 16 біт інформації із поля ідентифікації при цьому скинемо найбільш значущі 8 біт інформації. Отримане значення перетворюємо на символ який еквівалентний в ASCII. Отриманий символ, збережемо у файл (recsecret.txt). У даному випадку стегано-ключ – це правило виймання інформації або інвертована маска вбудовування [10].

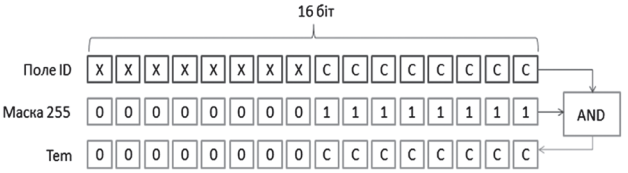


Рис. 5

До переваг методів зміни даних у полях заголовків мережного протоколу слід віднести [9]: простоту реалізації; високу стеганографічну пропускну здатність; не потрібно ніякої синхронізації між відправником та отримувачем. Як недоліки слід зазначити такі: погіршення якості роботи протоколу; легкість виявлення.

Іншим прикладом методів модифікації вмісту інформаційних пакетів є методи, що безпосередньо *змінюють власні дані пакетів* [4].

У цьому випадку застосовуються різні алгоритми цифрових водяних знаків, мовних кодеків та інших стеганографічних технік приховування даних [2].

Наприклад, до цієї групи можна віднести метод, який заснований на зміні розміру пакетів переданих даних. Особливість цього методу полягає у вбудовуванні інформації у розмір пакета даних, тобто саме розмір пакета міститиме інформацію про конфіденційне повідомлення.

Алгоритм методу «Розмір пакетів переданих даних» є наступним.

На комп'ютері відправника вводиться стеганограма. Відправник відсилає перший пакет, який містить у собі «стегано-ключ».

На основі «стегано-ключа» формується словник по одному і тому ж алгоритму на комп'ютері-відправника та комп'ютері-отримувача.

Починається передача файлів. Розмір кожного пакета відповідає букві або символу конфіденційного повідомлення. Пропускна здатність цього методу дорівнюватиме одній букві (символу) за 1 пакет.

До очевидних переваг таких методів слід віднести такі особливості: методи не потребують ніякої додаткової синхронізації між відправником та отримувачем інформаційного повідомлення; простота реалізації; інформаційне повідомлення важче виявити, ніж у випадку використання методів, які змінюють дані у полях заголовків протоколу. Як недоліки слід зазначити низьку стеганографічну пропускну здатність. Крім того, слід відмітити також потенційне погіршення якості отриманих даних користувача.

У роботах [4, 9, 11] розглянуто також *методи змішаних технік*, які побудовані через об'єднання двох попередніх методів. Прикладом реалізації цього методу є алгоритм HICCUPS (Hidden Communication system for CorrUPted networkS), який детально описується у роботах [9, 11]. До їх переваг слід віднести відсутність вимог щодо додаткової синхронізації між відправником та отримувачем, складність виявлення вбудованих повідомлень, високу стеганографічну пропускну здатність. Як недоліки слід зазначити високу складність практичної реалізації та підвищення рівня втрат мережних ресурсів.

2.2. Методи маніпуляції над потоками переданих пакетів. До цієї групи можна віднести метод затримки пакетів. Принцип функціонування методу затримки пакетів виглядає наступним чином (див. рис. 6) [3].

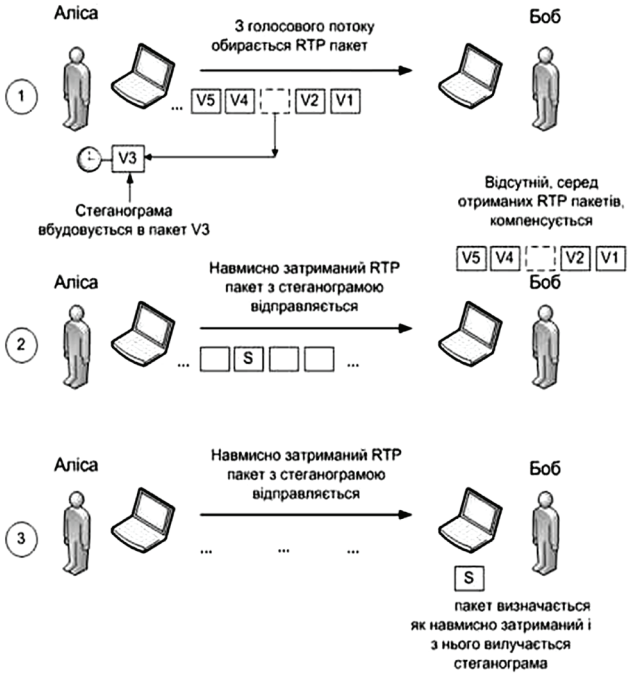


Рис. 6

Відправник (Аліса) обирає один з пакетів потоку і його корисне навантаження замінюється бітами секретного повідомлення – стеганограмою, яка вбудовується в пакет V3 (1). Потім обраний пакет навмисне затримується (2). Щоразу, коли надмірно затриманий пакет досягає отримувача, незнайомого з стеганографічною процедурою, він відкидається. Однак, якщо одержувач (Боб) знає про прихований зв'язок, то замість видалення отриманих пакетів, витягує приховану інформацію [3]. RTP – транспортний протокол, який належить до сімейства стека протоколів TCP/IP.

Як переваги даного методу слід відзначити простоту реалізації та важкість виявлення стеганоканалу. Як недоліки слід зазначити такі: дуже низька стеганографічна пропускна здатність; потрібна синхронізація між відправником й отримувачем та збільшення затримки передачі.

2.3. Гібридні методи. Метод RSTEG [9] (один із гібридних методів) заснований на механізмі повторної посилки пакетів, суть якої полягає у тому, що: коли відправник посилає пакет, то одержувач не відповідає пакетом з прапорцем підтвердження, і таким чином має спрацювати механізм повторної висилки пакетів і повторно посилається пакет зі стеганограмою всередині, на який також не приходить підтвердження. При наступному спрацьовуванні даного механізму надсилається оригінальний пакет без прихованих вкладень, на який приходить пакет з підтвердженням про вдалий отриманий (див. рис. 7). Перевагами даного методу є простота реалізації; важкість виявлення стеганоканалу; висока стеганографічна пропускна здатність; не потрібно ніякої синхронізації між відправником та отримувачем. Недоліками цього методу є потенційне погіршення якості даних користувача та підвищення рівня втрат у мережі.

Рис. 7

Результати проведеного аналізу відомих методів мережної стеганографії узагальнено у табл. 1, в якій наведено ранжування різних підходів до побудови захищених каналів скритої передачі повідомлень. Як показники ранжування використовувалися: складність практичної реалізації (у порядку ускладнення); надійність методів стеганозахисту (за спаданням); пропускна здатність (за спаданням).

Таблиця 1

Порівняльна характеристика методів

№	Методи	Реалізація	Надійність методів	Пропускна здатність
1	Методи, що змінюють тільки заголовки пакетів	1	5	4
2	Методи, що змінюють тільки передані дані	2	3	3
3	Методи, засновані на маніпуляціях над потоком переданих пакетів (сортування пакетів)	3	4	5
4	Методи, що включають одночасну модифікацію пакетних даних і маніпуляцію над потоком	4	1	1
5	Змішані методи, що використовують і заголовки пакетів, і дані	5	2	2

ВИСНОВКИ

Проведені дослідження показали, що методи мережної стеганографії останнім часом інтенсивно розвиваються та вдосконалюються. Це зумовлено стрімким поширенням мережних технологій та розповсюдженням сучасних інформаційно-телекомунікаційних засобів. Зокрема останніми роками відбулася справжня «революція мобільних пристроїв», коли майже кожна фізична особа експлуатує високопродуктивні «гаджети», які здатні надавати різні інформаційні послуги, в тому числі і за допомогою доступу до мережних обчислювальних та комунікаційних ресурсів. Таке стрімке поширення сучасних інфокомунікаційних технологій створює всі передумови для виникнення принципово нових технологій захисту інформації, зокрема методів скритної передачі конфіденційних повідомлень. Фактично, людство стоїть на порозі створення надтехнології прихованої передачі даних, коли застосовуючи відкриті канали зв'язку, зокрема засоби і технології глобальних мереж, можна миттєво та скритно організувати передачу потрібної інформації між будь-якими фізичними або юридичними особами, що розташовані у будь-якій точці земної кулі. При цьому приховується не тільки смисловий вміст переданих даних, але і сам факт існування скритої передачі інформації.

В цій роботі проведено аналіз найпростіших методів мережної стеганографії. І навіть такі методи виявили високі показники ефективності. У табл. 1 наведено порівняння досліджених методів за їх основними характеристиками та можливістю практичної реалізації. Позиція кожного методу в даній таблиці показує, на скільки його характеристики перевершують або поступаються іншим. На основі даних із таблиці можна зробити висновок про пряму залежність основних характеристик один від одного та перспективні шляхи щодо їх подальшого розвитку та вдосконалення. Проведені дослідження можуть бути використані як основа для розробки нових методів стеганографії або для захисту інформації від витоків по прихованих каналх, створених за допомогою розглянутих методів мережної стеганографії.

Література

[1] *Горбенко І.Д., Горбенко Ю.І.* Прикладна Криптологія. – Харків. Форт. – 2012. – С. 867.

[2] *Конахович Г.Ф., Пузеренко О.Ю.* Комп'ютерна стеганографія. Теорія і практика. – Київ. МК-Пресс. – 2006. – С. 288.

[3] *Коркач И.В., Пирогова Ю.И.* Использование технологий IP-телефонии для скрытой передачи информации. – [Електронний ресурс]. – Режим доступу: http://tzi.ua/ ru/vikoristannya_tehnologj_ip-telefon_ dlya_prihovano_ peredach_nformac.html

[4] *Пескова О.Ю., Халабурда Ю.Г.* Применение сетевой стеганографии для защиты данных, передаваемых по открытым каналам Интернет. - [Електронний ресурс] – Режим доступу: http://ojs.ifmo.ru/index.php/IMS/article/ view/132/132

[4] *Генне О.В.* Основні положення стеганографії [Електронний ресурс] – Режим доступу: http://easy-code.com.ua/2010/11/osnovni-polozhennya-stenografii/

[4] *Паркер Т., Сиян К.* TCP/IP. Для профессионалов. 3-е изд. – СПб.: Питер, 2004. – С. 859

[4] *Ногл М.* TCP/IP. Иллюстрированный учебник – М.: ДМК Пресс. – 480 с.

[4] *Thurston R.* Steganography developers turn their attention to hiding information in VoIP. [Електронний ресурс] – Режим доступу: http://www.scmagazineuk.com/ steganography-developers-turn-their-attention-to-hiding-information-in-voip/article/112102/

[4] *Sekhar A.* International Journal of Advanced Research in Computer and Communication Engineering. Vol. 4, Special Issue 1, June 2015 [Електронний ресурс] – Режим доступу: http://www.ijarcce.com/upload/2015/ si/icrtcc-15/IJARCCE%2017.pdf

[4] *Ziyad Tariq Mustafa, Authman Waleed Khalid.* Diyala journal for pure sciences. Packet Steganography Using IP ID. [Електронний ресурс]. – Режим доступу: http://www.sciencesmag.uodiyala.edu.iq/uploads/Volume%2010/Issue%204/English/1-10%20E.pdf

[4] *K. Szczypiorski,* HICCUPS: Hidden Communication System for Cor- rupted Networks, In Proc. of The Tenth International MultiConference on Advanced Computer Systems ACS–2003. – P. 31–40.

Надійшла до редколегії 10.11.2015

Кузнецов Олександр Олександрович, доктор технічних наук, професор, професор кадедри БІКС ХНУ ім. В.Н. Каразіна. Наукові інтереси: криптографія, теорія обробки і передачі даних, стеганографічні методи захисту інформації.

Тимченко Владислав Анатолійович, студент факультету комп'ютерних наук ХНУ ім. В.Н. Каразіна. Наукові інтереси: криптографія, стеганографія, теорія обробки і передачі даних.

Дубровний Богдан Сергійович, студент факультету комп'ютерних наук ХНУ ім. В.Н. Каразіна. Наукові інтереси: адміністрування баз даних, криптографія, стеганографія.

УДК 004.056:003.26

Анализ и сравнительные исследования методов сетевой стеганографии / А.А. Кузнецов, В.А. Тимченко, Б.С. Дубровный // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 384–389.

Рассматриваются возможности стеганографических методов защиты информации, создание которых основано на использовании особенностей функционирования сетевого стека протоколов передачи семейства TCP/IP (Transmission Control Protocol / Internet Protocol). Исследуются разные методы сетевой стеганографии: на кодировании содержания информационных пакетов; на манипуляциях потока переданных пакетов; гибридные методы. Обосновываются перспективные направления дальнейших исследований по улучшению методов сетевой стеганографии для их использования с целью скрытия факта наличия скрытой передачи информации.

Ключевые слова: сетевая стеганография, семейство протоколов TCP/IP.

Табл.: 2. Рис.: 8. Библиогр.: 11 назв.

UDC 004.056:003.26

Analysis and comparative research of network steganography methods / Kuznetsov A.A., Tymchenko V.A., Dubrovny B.S. // Applied Radio Electronics: Sci. Journ. – 2015. – Vol. 14. – № 4. – P. 384–389.

The paper reviews capabilities of steganography methods, the construction of which is based on using of peculiarities of functioning the transmission protocols network stack of TCP/IP family. Different methods of network steganography are studied, namely: those based on content encoding of information packets, on manipulations of a stream of packets transmitted as well as hybrid ones. Perspective ways of further research with improving methods of network steganography to be used to conceal the existence of secret information transfer are substantiated.

Keywords: network steganography, TCP/IP protocols family.

Tab.: 2. Fig.: 8. Ref.: 11 items.

УДК 004.056.5

ПОБУДОВА МОДЕЛІ ПОРУШНИКА ДЛЯ ЗАГРОЗИ ПІДМІНИ ЧИ МОДИФІКАЦІЇ АПАРАТНОГО ЗАСОБУ КЗІ ТА АНАЛІЗ СПОСОБІВ ПРОТИДІЇ

В.А. ПОНОМАР

У роботі розглядається модель порушника для загрози підміни чи модифікації апаратного засобу КЗІ. Проводиться аналіз способів захисту від такої загрози.

Ключові слова: засоби КЗІ, фізичний захист, автентифікація засобів КЗІ.

ВСТУП

В області інформаційних технологій все зростаючою є необхідність використання криптографічних механізмів, таких як захист даних від несанкціонованого розкриття чи маніпуляції, для автентифікації об'єктів і неспростовності. Безпека та надійність таких механізмів безпосередньо залежить від засобів криптографічного захисту інформації (КЗІ), які реалізують дані послуги. Більшість таких засобів реалізовано у вигляді криптографічних модулів. Криптографічний модуль – це набір апаратного забезпечення, програмного забезпечення або програмно-апаратного забезпечення, який реалізує функції безпеки і містяться в межах криптографічної границі.

Розвиток ринку засобів криптографічного захисту інформації, збільшення кількості самих засобів та розширення сфери їх використання надає актуальності як задачам захисту криптомодуля від дій зловмисника, так і підтвердження користувачу своєї автентичності та цілісності. Зростання кількості сторін, що приймають участь у виробництві засобів КЗІ, поставці, інсталяції, супроводження в експлуатації, використанні тощо, вимагає особливої уваги захисту криптографічних модулів від підміни та небезпеки реалізації цієї загрози.

1. ВИКОРИСТАНА ТЕРМІНОЛОГІЯ

Апаратний модуль [1] є модулем, криптографічна границя якого задається апаратними компонентами. Програмно-апаратне та програмне забезпечення, яке може включати операційну систему, може міститися в межах цієї апаратної криптографічної границі.

Програмний модуль є модулем, криптографічна границя якого охоплює виключно програмні компоненти (може бути один або декілька програмних компонентів), які виконуються в змінному експлуатаційному середовищі. Обчислювальна платформа і операційна система експлуатаційного середовища, в якому виконується програма, є зовнішніми по відношенню до визначеної границі програмного модуля.

Програмно-апаратний модуль є модулем, криптографічна границя якого охоплює виключно програмно-апаратні компоненти, які виконуються в обмеженому або незмінному експлуатаційному середовищі. Обчислювальна платформа і операційна система експлуатаційного середовища, в якому виконується програмно-апаратне забезпечення, є зовнішніми по відношенню до визначеної границі програмно-апаратного модуля, але явно пов'язані з модулем.

Криптографічна границя – це чітко визначений неперервний периметр, який встановлює фізичні та логічні межі криптографічного модуля і містить всі компоненти апаратного, програмного та програмно-апаратного забезпечення криптографічного модуля.

Апаратне забезпечення – фізичне обладнання та компоненти у межах криптографічної границі, що використовуються для обробки даних і програм.

Програмне забезпечення – виконуваний код криптографічного модуля, який зберігається на носії, який стирається, і може бути динамічно написаний і змінений під час виконання під час роботи в змінному експлуатаційному середовищі.

Програмно-апаратне забезпечення – це виконуваний код криптографічного модуля, який зберігається в апаратному забезпеченні в межах криптографічної границі і не може бути динамічно записаний або змінений під час виконання під час роботи в незмінному або обмеженому експлуатаційному середовищі.

Незмінне експлуатаційне середовище – експлуатаційне середовище, яка не здатне приймати зміни.

Обмежене експлуатаційне середовище – експлуатаційне середовище, яке здатне приймати тільки контрольовані зміни програмно-апаратного забезпечення, які успішно проходять тестування при завантаженні.

Змінне експлуатаційне середовище – операційне середовище, яке здатне приймати функціональні зміни, які можуть містити неконтрольоване програмне забезпечення (тобто, недовірене).

Інвазивні атаки – атаки, що вимагають відкриття корпусу модуля, щоб отримати прямий доступ до внутрішніх компонентів (компонентів, що містяться всередині криптографічної границі модуля).

Неінвазивні атаки – атаки, що використовують інформацію, що доступна ззовні, поява чи випромінювання якої може не передбачатися розробником: час роботи, споживання електроенергії, побічне електромагнітне випромінювання.

Виявлення вторгнення – автоматичне виявлення криптографічним модулем спроб компрометації безпеки модуля.

Доказ вторгнення – явне відображення того, що була зроблена спроба скомпрометувати безпеку криптографічного модуля.

Реагування на вторгнення – автоматичні дії, вжиті криптографічним модулем при виявленні вторгнення.

Довірений канал – надійна і безпечна лінія зв'язку, встановлена між криптографічним модулем і відправником або одержувачем для створення безпечного доступу до незахищених критичних даних у вигляді відкритого тексту, ключових компонентів і даних автентифікації.

2. АНАЛІЗ ВХІДНОГО ПОЛОЖЕННЯ

До побудови моделі порушника [2] та аналізу загроз приймаємо такі положення:

– реалізація загрози на будь-якому етапі циклу використання засобу може призвести до компрометації особистого ключа користувача;

– виробник апаратного засобу є стороною, якій довіряє користувач;

– виробник гарантує якість засобу КЗІ;

– компрометація засобу не має призводити до компрометації інших засобів;

– виробник не може мати відомості щодо усіх можливих користувачів засобу, але має мати відомості про усі пристрої, що було створено;

– користувач до персоналізації засобу може не мати можливості його використання.

За цими вимогами виробник не має доступу до інформації користувача і не має змогу зв'язати певний модуль з відповідним користувачем. Користувач довіряє виробнику та послугам, що той надає. З цього виникає задача для виробника забезпечення цієї довіри та підтвердження користувачу, що цей засіб дійсно виконаний виробником і на нього поширюється довіра користувача.

Сам апаратний засіб КЗІ у своєму циклі життєдіяльності проходить шість етапів:

1) виготовлення;
2) доставка;
3) інсталяція;
4) персоналізація;
5) експлуатація;
6) зняття з експлуатації.

На етапі виготовлення задіяно лише виробника. Доставку може виконувати виробник, користувач, третя сторона (залежно від договору). Інсталяцію на робочому місці можуть виконувати також виробник, користувач, третя сторона. Під час персоналізації, тобто вироблення та запису в криптографічний модуль персональної та критичної інформації бере участь лише користувач. Під час експлуатації модуль може використовуватися користувачем чи третьою стороною, залежно від призначення інформаційно-телекомунікаційної системи (ІТС) в якій цей засіб КЗІ задіяно. Зняття з експлуатації та деінсталяцію повинен виконувати користувач (видалення критичних даних), та можуть бути задіяні виробник або третя сторона.

3. МОДЕЛЬ ПОРУШНИКА

В нашому випадку порушник – це фізична чи юридична особа, що навмисно здійснює в системі неправомірні дії, які направлені на компроментацію критичних даних користувача та його даних, які обробляються за допомогою модифікованого криптографічного модуля.

За наших вхідних умов зловмисник не має змоги вплинути на криптографічний модуль на етапі вироблення, оскільки на цьому етапі приймає участь лише виробник,а оскільки ми прийняли твердження, що користувач повністю довіряє виробнику, то він не може бути порушником. На всіх інших етапах порушник може бути задіяний (рис. 1).

Під час доставки та інсталяції порушник може (якщо має можливість) виконати необхідну йому модифікацію чи підміну аппаратного засобу КЗІ.

На етапі персоналізації зловмисник не може мати прямий доступ до криптографічного модуля (оскільки на цьому етапі задіяний лише користу-

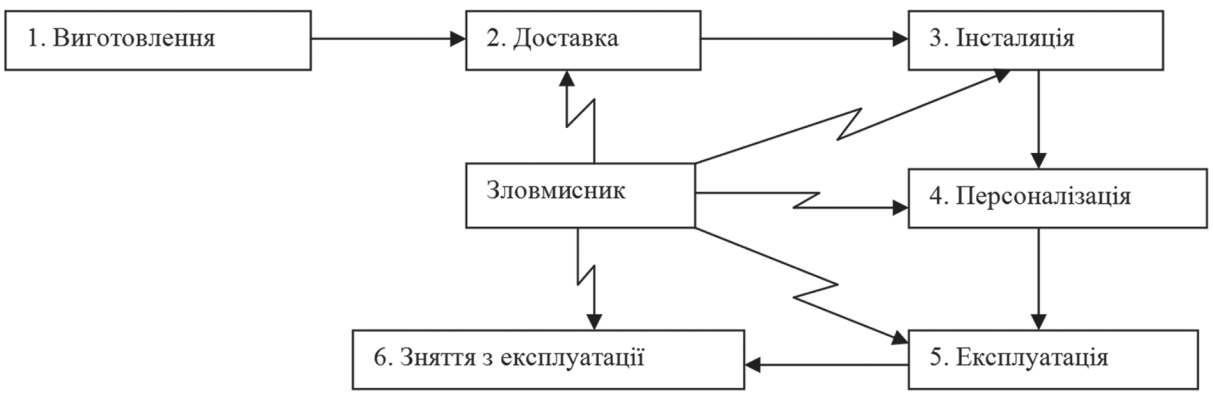


Рис. 1. Схема можливості впливу зловмисника

вач), але, якщо була виконана модифікація на етапах 2 чи 3, то це може призвести до компрометації критичної інформації під час персоналізації:

– вироблення криптографічно слабких параметрів[3];

– підсунення слабкого чи відомого зловмиснику ключа[3];

– збереження ключа в незахищеному вигляді чи в контейнері, з якого порушник зможе його отримати[4];

– викривлення вказаних користувачем ролей так, щоб порушник мав доступ до захищеної інформації;

– збереження та використання побічної інформації, що може бути в подальшому використана для порушення анонімності, якщо засіб КЗІ використовується для створення сліпого електронно-цифрового підпису (ЕЦП).

На етапі експлуатації порушник, при доступі до засобу КЗІ, може виконати необхідні йому дії з модулем чи отримати результат дій на попередніх етапах.

На етапі зняття з експлуатації порушник може отримати результати від попередніх впливів на засіб КЗІ.

За своїми можливостями, а виходячи з цього й можливими загрозами можна виділити два типи зловмисника:

1. Порушник 2 класу, який володіє усією інформацією про:

– особливості роботи криптографічного модуля;

– складові компоненти, їх зв'язки та функціонування;

– алгоритми та протоколи, що використовуються;

– заходи захисту критичної інформації та інформації, що обробляється.

Має необхідні засоби для виконання атак з фізичним доступом.

2. Порушник 3 класу, що володіє усіма можливостями зловмисника попереднього типу, але й ще володіє необхідним технічним та програмним забезпеченням для виготовлення копії засобу КЗІ, що матиме весь необхідний функціонал, але також будуть внесені модифікації, необхідні зловмиснику. Можливість виготовлення такої копії і подальшої заміни справжнього модуля на неї, дозволить порушнику отримати переваги, описані вище без прямої атаки засіб КЗІ.

Тому в подальшому ми розглянемо методи захисту від модифікації криптографічного модуля (захист від зловмисника 1 та 2 типу) та заміни криптографічного модуля на його копію (захист від зловмисника 2 типу).

4. ЗАХИСТ ВІД МОДИФІКАЦІЇ ПРОГРАМНОЇ СКЛАДОВОЇ

При доступі до криптографічного модуля порушник може спробувати замінити програмне забезпечення модуля, чи завантажити нове, що матиме або вразливості, або можливості обходу функцій захисту.

Основні методики захисту від такої модифікації це:

– організаційне розподілення ролей, що зашкоджує оновленню програмного забезпечення модуля оператору, якому це не дозволено його роллю;

– попереднє самотестування, що виявляє зміни в програмному забезпеченн і, які непередбачені виробником чи були внесені з порушенням організації доступу до програмного забезпечення. Код такого тестування має бути захищений в зміни.

Згідно з міжнародним стандартом ISO/IEC 19790:2012 [1] наводяться такі три вимоги до оновлення програмного забезпечення та самотестування:

1. Якщо криптографічний модуль має можливість завантаження програмного або програмно-апаратного забезпечення з зовнішнього джерела, то мають виконуватися такі вимоги:

– завантажене програмне, програмно-апаратне забезпечення має бути перевірене засвідчувальним органом до завантаження для отримання засвідчення;

– виведення даних через інтерфейс виведення даних має бути заборонене доки програмне, програмно-апаратне забезпечення завантажується і тестування процесу завантаження не буде успішно завершене;

– тестування процесу завантаження програмного, програмно-апаратного забезпечення має бути здійснене до того, як завантажений код буде виконаний;

– криптографічний модуль має затримати виконання будь-яких завантажених або змінених затверджених функцій безпеки, доки не буде успішно виконане попереднє самотестування;

– до інформації про версію модуля мають бути внесені зміни, що відображають додавання та оновлення завантаженого нового програмного або програмно-апаратного забезпечення.

Якщо завантаження нового програмного або програмно-апаратного забезпечення повністю замінює образ модуля, то модуль має розглядатися як зовсім новий модуль, який потребує перевірки засвідчувальним органом для отримання засвідчення. Новий образ програмного або програмно-апаратного забезпечення має виконуватися тільки після перезавантаження при увімкненому живленні. Всі критичні параметри безпеки мають бути обнулені перед виконанням нового образу.

2. Попередні та умовні самотестування криптографічного модуля надають оператору гарантії, що не виникнуть збої, які перешкоджатимуть правильній роботі модуля. Всі самотестування мають виконуватися, і модуль має визначати проходження або провал тестування без зовнішнього керування, наданих зовні вхід-

них текстових векторів, очікуваних на виході результатів, або втручання оператора, і незалежно від того, чи працює модуль в затвердженому або незатвердженому режимі.

Умовні самотестування мають виконуватись, коли викликається застосовна функція безпеки або процес (тобто функція безпеки, для якої необхідне самотестування).

Всі самотестування, визначені в основних алгоритмічних стандартах, мають застосовуватися криптографічним модулем. Всі самотестування, визначені додатково або замість самотестувань, зазначених в основних алгоритмічних стандартах, мають бути реалізовані для кожної затвердженої функції безпеки, методу встановлення критичних параметрів і механізму автентифікації.

Якщо криптографічний модуль провалює самотестування, модуль має увійти в стан помилки і вивести індикатор помилки. Криптографічний модуль не має виконувати ніякі криптографічні операції і виводити інформацію керування і дані через інтерфейси виведення керування і даних, перебуваючи у стані помилки. Криптографічний модуль не має використовувати ніяку функціональність, яка спирається на функції або алгоритми, що не пройшли самотестування, поки відповідні самотестування не будуть повторені і успішно пройдені. Якщо модуль не виводить статус помилки при провалі самотестування, оператор модуля повинен бути в змозі визначити, чи увійшов модуль в стан помилки неявно через однозначні процедури, описані в політиці безпеки.

3. До умовного самотестування при завантаженні програмного, програмно-апаратного забезпечення висуваються додаткові вимоги. Якщо криптографічний модуль володіє можливістю завантаження програмного, програмно-апаратного забезпечення з зовнішніх джерел, то мають виконуватись такі вимоги:

– криптографічний модуль має реалізовувати затверджений метод автентифікації для перевірки справжності програмного, програмно-апаратного забезпечення, яке завантажується;

– відповідний ключ автентифікації має бути незалежно завантажений в модуль до завантаження програмного, програмно-апаратного забезпечення;

– застосовний затверджений метод автентифікації має бути успішно пройдений, або тестування завантаження програмного, програмно-апаратного забезпечення буде провалено. Завантажене програмне, програмно-апаратне забезпечення не має використовуватися, якщо тестування завантаження програмного, програмно-апаратного забезпечення було провалено.

Крім того для оновлення програмного, програмно-апаратного забезпечення розробником може передбачатися використання скриптових мов [5]. Тоді для унеможливлення виникнення загроз в ході використання скриптової мови управління засобом КЗІ висуваються додаткові вимоги:

– користувач скриптової мови не повинен мати безпосередній доступ до скритих системних даних. Ця вимога стосується, по-перше, критичної інформації системи. Якщо є необхідність використання цих даних у програмі, то вказується індекс чи ім'я необхідної інформації і під час виконання цієї програми система вирішує надавити їй чи ні даний вид доступу. По-друге, це стосується звичайних системних параметрів, що використовуються для функціонування команд скриптової мови;

– скриптова мова повинна не давати можливості написання небезпечних команд управління. Виконання цієї вимоги не дає можливості зловмиснику створювати та завантажувати до системи програм для порушення послуг безпеки інформації;

– завантажувати написані команди має право тільки автентифікований оператор, для якого в політиці безпеки надається така можливість;

– команди користувача не можуть замінити команди розробника.

5. ЗАХИСТ ВІД МОДИФІКАЦІЇ АПАРАТНОЇ СКЛАДОВОЇ

Будь-яка спроба прямого фізичного доступу до внутрішніх компонентів засобу КЗІ, їх модифікація, заміна, встановлення нових компонентів називається вторгненням. На різних етапах роботи криптографічного модуля порушник, під час вторгнення, переслідує різні цілі [3, 4]:

1) модифікація, мета якої зміна процесу генерації критичних параметрів, що призведе до генерації криптографічно-слабких чи відомих зловмиснику ключів (2–4 етапи);

2) модифікація чи встановлення компонентів, що призведуть до компрометації критичних параметрів чи інформації, що оброблятиметься (2–5 етапи);

3) вторгнення, мета якої отримання необхідної для зловмисника інформації напряму з внутрішніх компонентів криптомодуля чи завдяки проведеному на попередніх етапах впливу (5–6 етапи).

Тому для запобігання модифікації апаратної складової, криптографічний модуль повинен вміти виявляти під час самотестування спроби таких вторгнень. Міжнародний стандарт ISO/IEC 19790:2012 [1] пропонує виділити чотири зростаючі, якісні рівні вимог безпеки, які призначені для покриття широкого спектра потенційних застосувань і середовищ. Вимоги фізичної безпеки визначені для трьох різних фізичних виконань криптографічного модуля.

1. Одночіпові криптографічні модулі – це фізичні виконання, у яких один чіп з інтегральною схемою (ІС) може використовуватися як автономний пристрій, або вбудований в корпус чи в продукт, який може бути фізично не захище-

ним. Прикладами одночіпових криптографічних модулів є поодинокий ІС-чіп або смарт-картка з одним ІС-чіпом.

2. Багаточіпові вбудовані криптографічні модулі – це фізичні виконання, у яких два або більше ІС-чіпів взаємопов'язані і вбудовані в корпус або в продукт, який може бути фізично не захищеним. Прикладами багаточіпових вбудованих криптографічних модулів є адаптери і плати розширення.

3. Багаточіпові автономні криптографічні модулі – це фізичні виконання, у яких два або більше ІС-чіпів взаємопов'язані і весь корпус фізично захищений. Прикладами багаточіпових автономних криптографічних модулів є шифруючі маршрутизатори, безпечні радіостанції або USB-токени.

У табл. 1 наведено вимоги фізичної безпеки, як загальні, так і для трьох конкретних виконань для кожного із чотирьох рівнів безпеки. Вимоги фізичної безпеки, що залежать від варіанта виконання, на кожному рівні безпеки посилюють загальні вимоги для цього рівня, і залежні від виконання вимоги попереднього рівня.

Вимоги безпеки визначаються і для інтерфейсу технічного обслуговування, якщо конструкція криптографічного модуля забезпечує фізичний доступ (наприклад, для розробника чи постачальника модулів, або іншої уповноваженої особи).

Механізми виявлення вторгнення та реагування на вторгнення не замінюють доказ вторгнення. Криптографічний модуль має містити механізми реагування на вторгнення, які повинні постійно стежити за виявляючою вторгнення оболонкою і, при виявленні вторгнення, повинні відразу обнулити всі незахищені критичні дані (залежно від політики безпеки можливо обнулити взагалі усі критичні дані).

Криптографічний модуль має вести спостереження і правильно реагувати, коли робоча температура і напруга (чи темпи їх зміни) виходять за межі визначеного нормального робочого діапазону.

Якщо температура або напруга виходить за межі нормального робочого діапазону криптографічного модуля, механізм захисту має або виключити модуль, щоб запобігти подальшій експлуатації, або відразу обнулити критичні дані.

Система виявлення та реагування на вторгнення не має залежати від зовнішнього живлення засобу КЗІ. Ця вимога дозволить протидіяти вторгненню, навіть якщо воно відбулося коли криптографічний модуль було вимкнуто (наприклад, на етапі 2).

Але, якщо порушником було здійснено вторгнення до завершення етапу персоналізації, і одним з результатів якого буде те, що під час подальшого використання криптомодуль не виявлятиме цього вторгнення (обнуління відбудеться відразу, але оскільки засіб ще не персоналізовано, то користувач не знатиме про це, оскільки його критчних даних ще немає). У разі, якщо метою вторгнення було встановлення закладки, то отримання її інформації можна виконати на етапі 6 і тоді користувач не знатиме, що засіб КЗІ був модифікований.

Щоб протидіяти даній атаці, необхідно щоб система реагування на вторгнення унеможливлювала подальшу роботу будь-яких компонентів у складі даного засобу КЗІ так, щоб робота криптографічного модуля була можлива тільки після заміни усіх компонентів, а це рівнозначно виготовленню нового засобу КЗІ, що за нашою моделлю порушника може зробити тільки порушник другого типу і не відноситься до атаки виду модифікації криптографічного модуля.

Таблиця 1

Вимоги фізичної безпеки до засобу КЗІ залежно від виконання та призначення

	Загальні вимоги для всіх варіантів виконань	Одночіпові	Багаточіпові
Рівень безпеки 1	Заводські компоненти. Стандартна пасивація. Процедурне або автоматичне обнуління при здійсненні доступу до інтерфейсу технічного обслуговування	Немає додаткових вимог	Корпуси виробничого класу або знімне кришки
Рівень безпеки 2	Доказ вторгнення. Непрозорі або напівпрозорі в межах видимого спектра. Попереджають пряме спостереження через отвори і щілини	Покриття, що доказує вторгнення на чіпі або корпусі	Герметизуючий матеріал, що доказує вторгнення, або корпус з печатками, що доказують вторгнення, або стійкі замки на дверцятах та знімних кришках
Рівень безпеки 3	Схема реагування на вторгнення і обнуління. Автоматичне обнуління при здійсненні доступу через інтерфейс технічного обслуговування. Запобігання зондуванню через отвори і щілини. Захист від помилок, спричинених середовищем для температури і напруги	Жорстке покриття, що доказує вторгнення на чіпі або міцний, стійкий до видалення та проникнення корпус	Жорсткий герметизуючий матеріал, що доказує вторгнення, або міцний корпус
Рівень безпеки 4	Оболонка з виявленням та реагуванням на вторгнення. EFP для температури та напруги. Захист від індукції помилки	Жорстке стійке до видалення покриття на чіпі	Оболонка з виявленням та реагуванням на вторгнення та з можливістю обнуління

6. ЗАХИСТ ВІД ПІДРОБКИ КРИПТОГРАФІЧНОГО МОДУЛЯ

Виконання підміни може виконуватися лише до завершення етапу персоналізації, тому що із відповідністю вимог, які були викладені вище, виконання підміни на підроблений засіб КЗІ після персоналізації призведе до втрати робочим модулем критичної інформації користувача, що буде відразу виявлено.

Оскільки зловмисник типу 2 може створити повністю новий криптографічний модуль зі своїми модифікаціями, то задача захисту від такої підробки націлена на переконання користувача, що використовується справжній засіб КЗІ – тобто задача автентифікації криптомодуля.

Це вимагає від виробника записати у засіб КЗІ додаткову ключову інформацію, наявність якої перевірятиметься під час початку роботи (процес автентифікації засобу). Прямий доступ та керування цією інформацією заборонені, при обнуленні через виявлення вторгнення має бути знищена (для запобігання її отримання зловмисником і використання у копії криптографічного модуля).

Пропонується для автентифікації використати протоколи стандарту ISO/IEC 9798 – 3 [6], тобто автентифікація на основі цифрового підпису. Оскільки нам необхідно автентифікація лише засобу КЗІ, то використовуватимемо однобічну автентифікацію, а для підвищення стійкості, вона відбуватиметься у два проходи (рис. 2).

Об'єкт В – засіб КЗІ. Об'єкт А – оператор (забезпечення автентифікації).

Структура маркера *TokenBA*, що відсилається пред'явником В об'єкту А, має вигляд:

$$\begin{aligned} &TokenBA = R_A \| R_B \| A \| \\ &\| Text3 \| \\ &\| sS_A\left(R_A \| R_B \| A \| Text2\right) \end{aligned} \quad (1)$$

Протокол виконується у такому порядку:

1. Об'єкт А генерує випадкове число R_A та відсилає його об'єкту В, також об'єкту В може бути відіслано текстове поле *Text*1 (необов'язково).

2. Об'єкт В формує та відсилає маркер *TokenBA* об'єкту А , а також свій сертифікат.

3. Після отримання повідомлення, що містить *TokenBA*, об'єкт А виконує такі кроки:

а) впевнюється в тому, що володіє дійсним відкритим ключем об'єкта у вигляді сертифіката або будь-яких інших засобів;

б) перевіряє коректність маркера *TokenBA* шляхом:

– перевірки електронного цифрового підпису, що міститься у маркері;

– перевірки відповідності випадкового числа R_A, що було надіслано об'єкту В на кроці (1) та випадкового числа, що міститься в підписаних даних маркера *TokenBA*;

– порівняння ідентичності значення поля ідентифікатора А в підписаних даних маркера *TokenBA* та розпізнавального ідентифікатора об'єкта А (тобто самого оператора).

Рекомендується включення розпізнавального ідентифікатора А в маркер *TokenBA* (стандартом це не вимагається). Його включення не дозволить зловмиснику звузити обсяг можливих повідомлень для підпису.

Крім перевірок передбачених протоколом, об'єкт А також може перевіряти чи використовувалося число R_B до поточного сеансу (хоча б перевірка попередніх декількох сеансів).

Стійкість наведеного протоколу випливає зі стійкості цифрового підпису (асиметричних криптоперетворень). В ході використання затверджених криптоалгоритмів з доведеною стійкістю зловмисник не зможе отримати особистий ключ криптомодуля за допомогою криптоаналізу. Під час виконання вимог, що наведені в попередніх пунктах, зловмисник також не зможе фізично зчитати цей ключ. Завдяки використанню в підписі R_A, ідентифікатора А (якщо можливих операторів декілька, тобто неможливо з 100% ймовірністю спрогнозувати цей ідентифікатор), а також можливого обмеження на повторне використання R_B, зловмисник не зможе записати в свій криптографічний модуль масив підписів на прогнозовані записи. Створення та збереження повної таблиці відповідності повідомлення-підпис (чи геш-підпис) для сучасних алгоритмів не можливо через великий її розмір і обмеження, що накладаються апаратною реалізацією криптографічного модуля (наприклад, при використанні нового стандарту гешування ДСТУ 7564:2014 [7] в режимі «Купина-256» знадобиться 2^{256} записів підпису і, якщо кожен запис займатиме хоча б 32 байти, то знадобиться $2^{261} \approx 10^{78,3}$ байт).

Розглянемо вимоги до реалізації наведеного протоколу для автентифікації криптографічного модуля з метою протидії атаки підміни:

1) для перевірка підпису (3-й крок) оператор не повинен використовувати сам криптомодуль, бо можлива зловмисником реалізація, при якій завжди такий підпис вважатиметься правильним;

2) сертифікат криптографічного модуля не має бути самопідписаним. Бо наша задача не тільки довести, що в засобі КЗІ зберігається певний ключ, а й довести, що цей засіб виготовлений виробником. Якщо ж сертифікат модуля буде самопідписаний, то при заміні самого модуля порушник може замінити й сертифікат на сер-

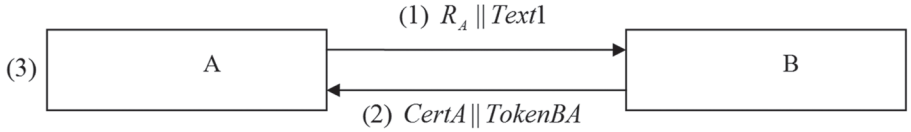


Рис. 2. Схема однобічної двопрохідної автентифікації

тифікат відкритого ключа, що відповідає ключу записаному в підробленому криптографічному модулі;

3) сертифікат засобу КЗІ має бути підписаний на ключі Виробника. Сертифікат виробника може бути самопідписаний чи вести далі до ЦСК, якому довіряє користувач (до ЦЗО включно);

3*а*) якщо сертифікат Виробника самопідписаний, то:

– він повинен постачатися окремо від засобу КЗІ (інакше можлива заміна на етапах 2, 3);

– даний сертифікат не має бути в межах прямого доступу оператора (щоб унеможливити заміну на етапах 4, 5), він має надаватися зовнішнім сервісом;

– якщо в даній ІТС не можливо використання зовнішнього сервісу, то сертифікат виробника має бути захищений від несанкціонованого перезапису.

ВИСНОВКИ

В ході побудови моделі загрози, коли виконана атака модифікації чи підміни засобу КЗІ, виявилося, що дана атака несе в собі більші загрози, ніж звичайна спроба атаки з фізичним доступом. Тому особлива увага приділялася розкриттю можливих дій зловмисника та способів протидії їм:

1) зловмисник за своїми можливостями та цілями розділений на два типи: виконуючий атаку модифікації та виконуючий атаку підробки;

2) при захисті від модифікації програмної складової засобу КЗІ особлива увага приділяється політиці безпеки та управління; розподіленню ролей; реалізації процесів завантаження, тестування та виконання програм; а також самотестування криптографічного модуля;

3) при захисті від атак на апаратну складову криптографічного модуля особлива увага приділялася на виконання криптомодуля; забезпечення рівня захищеності; механізм виявлення та реагування на вторгнення, що є основним компонентом системи захисту від атак з фізичним доступом (складової частини атаки модифікації засобу КЗІ);

4) вимоги стосовно попередніх питань забезпечені міжнародними стандартами, але на жаль не всі ці стандарти мають національні аналоги (чи плануються хоча б у вигляді приняття методом «обкладинки»). Відсутні аналоги вимог до фізичного захисту (ISO/IEC TS 30104:2015), тестуванню (ISO/IEC 18367), захисту від неінвазивних атак (ISO/IEC 17825) та перевірку і підвердження реалізації криптографічних протоколів у засобах КЗІ (ISO/IEC 29128:2011);

5) розглянута проблема захисту від підробка. Для її вирішення запропоновано використання протоколу автентифікації на основі цифрового підпису. Висунуто вимоги та рекомендації до реалізації цього протоколи з ціллю протидії атаці підробки.

Література

[1] *ISO/IEC 19790:2012.* Information technology – Security techniques – Security requirements for cryptographic modules. – International standard ISO/IEC, 2012. – 72 p.

[2] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Теорія. Практика. Застосування: монографія. – Харків: «Форт», 2012. – 870 с.

[3] *Шевчук О.* Актуальність атаки на зв'язанних ключах для апаратних реалізацій засобів КЗІ // Радиотехника. – 2011. – № 166. – С. 70–75.

[4] *Горбенко, Ю. І.* Аналіз атак з фізичним доступом та обґрунтування вимог до захисту криптографічних модулів від них [Текст]/ Ю. І. Горбенко, В. А. Пономар // Радіотехніка : Всеукр. міжвід. наук.-техн. зб. – 2015. – №180. – С. 144–151.

[5] *Горбенко, Ю. І.* Обґрунтування вимог, розробка та дослідження скриптової мови безпечного управління в інформаційно-телекомунікаційних системах [Текст] / Ю. І. Горбенко, В. А. Бобух, В. А. Пономар // Вісник НТУ «ХПІ». Серія: Нові рішення в сучасних технологіях. – Харків: НТУ «ХПІ», – 2014. – №26 (1069). – С. 73–88.

[6] *ISO/IEC 9798-3:1998.* Information technology –Security techniques – Entity authentication – Part 3: Mechanisms using digital signature techniques. – International standard ISO/IEC, 1998. – 6 p.

[7] ДСТУ 7624:2014 "Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення".

Надійшла до редколегії 17.11.2015

Пономар Володимир Андрійович, аспірант факультету комп'ютерних наук, кафедри безпеки інформаційних систем і технологій Харківського національного університету імені В.Н. Каразіна. Наукові інтереси: криптографічні перетворення, безпечне програмування, методи багатофакторної автентифікації та їх застосування з метою захисту інформації, захист криптографічних засобів інформації.

УДК 004.056.5

Построение модели нарушителя для угрозы подмены или модификации аппаратного средства КЗИ и анализ способов противодействия / В.А. Пономарь // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 390–396.

В работе рассматривается модель нарушителя для угрозы подмены или модификации аппаратного средства КЗИ. Проводится анализ способов защиты от такой угрозы.

Ключевые слова: средства КЗИ, физическая защита, аутентификация средств КЗИ.

Табл. 1. Рис.: 2. Библиогр.: 7 назв.

UDC 004.056.5

Construction of the model of an attacker for threatening substitution and modification of hardware cryptographic module and analysis of protection ways / V.A. Ponomar // Applied Radio Electronics: Sci. Journ. – 2015. – Vol. 14. – № 4. – P. 390–396.

The paper deals with the model of an attacker for threatening substitution and modification of cryptographic module. Analyzing ways of protection against this threat is performed.

Keywords: cryptographic modules, physical protection module, authentication of cryptographic modules.

Tab.: 1. Fig. 2. Ref.: 7 items.

УДК 681.3.06

КОМПЛЕКСНИЙ МЕТОД ПОРІВНЯННЯ, ОЦІНКИ ТА ВИБОРУ МЕХАНІЗМІВ УПРАВЛІННЯ КЛЮЧОВИМИ ДАНИМИ КОРИСТУВАЧІВ У ХМАРНИХ ІТС

І.Ф. АУЛОВ

Запропоновано методику, що дозволяє провести комплексне порівняння механізмів управління ключовими даними користувачів у хмарних ІТС за сукупністю критеріїв та показників. Наводиться обґрунтування критеріїв та показників оцінки механізмів управління ключовими даними користувачів у хмарі.

Ключові слова: механізми управління, ключові дані, хмарні ІТС, порівняльний аналіз.

ВСТУП

Сьогодні в ході використання хмарної інфраструктури користувачами, особливо при переході на її з власних інфраструктурних рішень виникає проблема управління ключовими даними користувачів. У першу чергу ця проблема пов'язана з необхідністю довіри користувача до постачальника хмарних послуг.

Аналогічна проблема існує і у випадку використання приватної хмари. Кінцевим користувачам необхідно надавати зручний доступ та можливості гнучкого управління їх ключовими даними безпосередньо в середовищі хмари при цьому забезпечуючи гарантії, щодо неможливості отримання несанкціонованого доступу до їх ключових даних іншими користувачами, в першу чергу обслуговуючим персоналом хмари [1–3].

На сучасному етапі розвитку хмарних технологій пропонується декілька механізмів з управління ключовими даними користувачів у середовищі хмарних обчислень. Метою дослідження є розробка методики, яка дозволить проводити комплексну оцінку та порівняння існуючих та перспективних механізмів управління ключовими даними користувачів у хмарі. Для досягнення мети було вирішено такі задачі: формалізація задачі управління ключовими даними користувача в хмарі, обґрунтування та вибір критеріїв та показників, вибір та обґрунтування методики порівняння, визначення обмежень до застосування методики.

1. СУЧАСНИЙ СТАН ДОСЛІДЖЕНЬ

Аналіз зарубіжних публікацій та патентів в області хмарних обчислень показав, що для управління ключами користувача існує 6 базових механізмів [4–13]:

– механізм управління ключами з використанням сертифікатів відкритих ключів;

– механізм управління ключами на основі паролів;

– механізм управління ключами з використанням HSM хмарного ЦОД;

– механізм управління ключами з використанням HSM користувача;

– механізм управління ключами з використанням криптографічного сервісу та захищеного сховища ключів;

– комбінований механізм управління ключами.

В першу чергу дослідники механізмів управління ключами приділяють увагу розробці нових, а також адаптації до вимог хмари та впровадженню існуючих рішень для управління ключами в середовищі хмарних обчислень.

Також суттєва увага дослідників приділяється питанням захисту та управління ключами в хмарі. До них відносяться такі [1–3]:

1. Встановлення таємних чи особистих ключів користувачів на об'єктах хмари з забезпеченням їх конфіденційності, цілісності, справжності, доступності, неспростовності та надійності.

2. Розповсюдження та встановлення чи отримання доступу до відкритих ключів користувачів з забезпеченням їх цілісності, справжності, доступності, неспростовності та надійності.

3. Оперативне виявлення фактів компрометації таємних та особистих ключів користувачів та виведення їх з дії з подальшим захищеним відновлення.

4. Дистанційне блокування чи знищення таємних та особистих ключів користувачами чи уповноваженими особами.

5. Захист таємних та особистих ключів користувачів від несанкціонованого доступу від внутрішніх та зовнішніх порушників, включаючи спеціалістів служб безпеки.

6. Створення, впровадження та застосування інфраструктур відкритого ключа (ІВК) в частині виготовлення та обслуговування сертифікатів відкритих ключів для надання електронних довірчих транскордонних послуг.

7. Забезпечення відносно особистих та відкритих ключів ІВК різних періодів чинності, наприклад чинність відкритих ключів має бути не менше періоду зберігання захищеної інформації архівів тощо.

Аналіз показав, що наведені вище проблемні питання відносяться і до захисту ключових даних безпосередньо хмарних обчислень та службових ключових даних посадових осіб тощо.

2. ВИМОГИ ДО МЕХАНІЗМІВ УПРАВЛІННЯ КЛЮЧОВИМИ ДАНИМИ В ХМАРІ

Безумовною вимогою для гарантування належного рівня безпеки в хмарі є застосування загальноприйнятих та міжнародних практик з використання та управління ключами. Надання обов'язкових для хмари послуг, таких як самообслуговування на вимогу, забезпечення універсального доступу, об'єднання ресурсів, швидка еластичність та облік споживання, а також відсутність контролю з боку користувача над хмарним середовищем, одночасна обробка декількох потоків інформації з різним рівнем безпеки, неоднорідність системи та її розгалуженість висуває нові вимоги до механізмів управління ключовими даними.

Аналогічно існуючим ІТС, в середовищі хмарних обчислень висуваються вимоги з забезпечення конфіденційності, цілісності, доступності, спостережливості, неспростовності, криптоживучості та надійності ключів. Виконання цих вимог може бути досягнуто за рахунок технічних, організаційних та криптографічних методів захисту.

В середовищі хмарних обчислень існують значні проблеми в забезпеченні виконання цих вимог. Аналіз еталонної моделі хмари NIST США показав, що в процесі використання хмарних сервісів користувачам можуть надаватися три типи послуг – IaaS, PaaS та SaaS [1], а також необхідні засоби забезпечення безпеки та архітектурні рішення, що забезпечують реалізацію цих послуг.

Із [1–3] випливає, що управління ресурсами, які пов'язані з хмарними сервісами, є найважливішим аспектом хмарних обчислень та потребують забезпечення безпеки. При чому, на рівні з організаційними та технічними механізмами та методами забезпечення безпеки, криптографічні механізми та методи є обов'язковими для забезпечення безпечного управління ресурсами хмари. Окрім необхідності забезпечення безпеки управління хмарою та ресурсами, розглядаючи згідно з моделлю NIST роль користувача в хмарі, необхідно зазначити, що криптографічні перетворення необхідні також для безпечної взаємодії користувача хмари з різними сервісами хмари, а також для зберігання даних, що були генеровані та/або оброблені цими сервісами.

Таким чином система управління ключами, що необхідна для підтримки криптографічних перетворень для зазначених вище функцій, може бути складною, зважаючи на відмінності відносно контролю над рівнями інфраструктури хмари, в яких знаходяться KMS та ресурси, що захищаються.

Наприклад, хоча споживач і має право власності на дані, що знаходяться та обробляються в хмарі, дані фізично зберігаються на ресурсах, що контролюються провайдером хмарних послуг, і в багатьох випадках KMS, яка необхідна для управління ключами для захисту даних також буде розгорнута на ресурсах постачальника хмарних послуг. Це породжує проблему безпеки і довіри хмарних користувачів до криптографічних операцій, що виконуються в хмарі.

Хоча є деякі розбіжності в послугах, що надаються хмарними провайдерами, але можна визначити набір функцій для основних моделей надання послуг (IaaS, PaaS та SaaS). Грунтуючись на цьому наборі функції визначаються можливості щодо реалізації функцій захисту та архітектурних рішень для досягнення цих можливостей безпеки. Слід зазначити, що не залежно від реалізованих функцій безпеки та архітектурного рішення, у випадках, коли криптографічні ключі зберігаються в хмарі є обмеження на ступінь забезпечення безпеки, на яку може розраховувати споживач хмари при будь-якій моделі надання послуг, в зв'язку з тим, що логічні і фізичні ресурси повністю під контролем хмарного провайдера.

Проведений аналіз та враховуючи рекомендації [1] дозволяє визначити три основні групи вимог, що висуваються до механізмів управління ключами в хмарі: орагізаційні та правові, технічні та функціональні, інформаційної безпеки.

До організаційних та правових вимог відносятся:

– відповідність міжнародним та державним стандартам та нормативним документам;

– проходження обов'язкової міжнародної та\або державної сертифікації;

– проведення аудиту впроваджених механізмів захисту;

– моніторинг, тестування та оновлення програмного та апаратного забезпечення;

– укладення договорів між користувачами та постачальниками;

– створення комплексної системи захисту для сервісу управління ключовими даними;

– забезпечення розмежування доступу та контролем над обслуговуючим персоналом;

– забезпечення безперебійної роботи, резервування та захист від відмов механізму управління ключами з причини стихійного лиха, збоїв в роботі сервісів хмари тощо;

– можливість зміни постачальника послуг.

До складу технічних і функціональних вимог, яким має відповідати механізм в першу чергу відносяться базові вимоги, що висуваються до хмари, а саме забезпечення:

– самообслуговування користувача на вимогу: функції пов'язані з підключенням та відключенням послуг сервісу управління ключами, безпосереднім управлінням ключами (генерацією, використанням, блокуванням, знищенням) мають надаватися автоматизовано;

– доступу до сервісу управління ключами за рахунок використання стандартних протоколів мережі та клієнтського програмного забезпечення для різних платформ;

– об'єднання ресурсів сервісу управління ключами в спільний пул та надаватися користувачам з нього залежно від їх запитів;

– надання та звільнення наданих ресурсів сервісу управління ключами має виконуватися автоматично за запитом від користувача з найбільшою швидкістю;

– вимірювання та контроль за обліком споживання ресурсів як користувачем, так і хмарним провайдером.

Окрім зазначених вище обов'язкових вимог існують такі, що можуть виконуватися частково залежно від потреб користувача:

– забезпечення функціонування та надання послуг сервісу управління ключами для моделей надання послуг IaaS, PaaS, SaaS;

– забезпечення функціонування та надання послуг сервісу управління ключами для різних моделей розгортання хмари;

– вимоги до технічних показників сервісу управління ключами, наприклад, такі як максимальна кількість клієнтів, що може одночасно обслуговуватися, середній час обробки запиту від клієнта, протоколи, алгоритми та формати ключових даних, що підтримуються, тощо.

Для задовільнення вимог інформаційної безпеки механізм має забезпечувати:

– автентифікацію сторін та перевірку дозволу при доступі до функцій управління ключовими даними окремого ключа;

– захист від підміни (автентифікація джерела має здійснюватися до виконання команд) та модифікації (забезпечення цілісності) команд управління ключами, результатів їх виконання, даних, що передаються;

– захист секретних та особистих ключів від несанкціонованого доступу;

– захист ключів та їх ключових даних від підміни;

– захист ключів та ключових даних від неявних та несанкціонованих модифікацій (забезпечення цілісності);

– стійкість криптографічних перетворень, що використовуються для захисту ключів не меншу, ніж стійкість ключів, що захищаються;

– захищений канал передачі (у випадках передачі ключів та ключових даних) з стійкістю не меншою, ніж стійкість ключів, що захищаються;

– виконання криптографічних перетворень, що застосовують ключі в довіреному середовищі.

3. КРИТЕРІЇ ТА ПОКАЗНИКИ МЕХАНІЗМІВ УПРАВЛІННЯ КЛЮЧОВИМИ ДАНИМИ В ХМАРІ

Окрім зазначених вище вимог, які висуваються до механізмів управління ключовими даними в хмарі під час проведення порівняння та вибору механізму серед інших пропонується використовувати групи критеріїв, що складаються з показників продуктивності (ПП), технічних показників (ТП), показників можливості розширення (ПМР), економічних показники (ЕП).

Як показники продуктивності пропонується використовувати:

– максимальна кількість сесій користувачів (ПП-1);

– щвидкість надання доступу до управління ключами, оп/с (ПП-2);

– швидкість криптографічних операцій за показником (ПП-3);

– швидкість генерації ключів, кл/с (ПП-4);

– швидкість захисту ключів, кл/с (ПП-5).

До технічних показників відносяться:

– максимальна кількість ключів, що можуть зберігатися (ТП-1);

– клас захищеності (ТП-2);

– захищеність сховища ключів (ТП-3);

– функції управління ключами в хмарі (ТП-4);

– моделі розгортання хмари, що підтримуються (ТП-5);

– моделі надання послуг у хмарі, що підтримуються (ТП-6);

– криптографічні алгоритми, що підтримуються (ТП-7);

– механізм автентифікації користувача (ТП-8).

До показників можливості розширення відносяться:

– функції криптографічного сервісу (ПМР-1);

– функції реєстрації подій (ПМР-2);

– функції ідентифікації та автентифікації (ПМР-3);

– функції сервісу управління ключами (ПМР-4);

– підтримка розподілених вузлів (ПМР-5);

– підтримка архівування, зберігання та відновлення ключів (ПМР-6);

– максимальна кількість розподілених вузлів (ПМР-7);

– гнучкість, щодо розширення та оновлення (ПМР-8).

До економічних показників відносяться:

– вартість рішення, у.о. (ЕП-1);

– вартість супроводження, у.о./рік (ЕП-2);

– вартість навчання відповідального персоналу, у.о./рік (ЕП-3).

4. АНАЛІЗ ЕФЕКТИВНОСТІ МЕХАНІЗМІВ УПРАВЛІННЯ КЛЮЧАМИ НА ОСНОВІ ЕКОНОМІЧНОГО ПОКАЗНИКА

Під час проведення аналізу механізмів та методів забезпечення безпеки в АС на практиці прийнято використовувати формалізовані моделі та методики аналізу ефективності систем захисту. Загальноприйнятим є твердження, що при синтезі механізмів захисту в першу чергу необхідно користуватися принципом розумної достатності [17], що передбачає застосовування формального критерію відповідності побудова-

ного механізму цьому принципу. В літературі як критерій ефективності захисту прийнято використовувати «комплексу оцінку захищеності», яка є сукупністю оцінок з ефективністю застосування програного-апаратних та організаційно-правових засобів захисту [17].

Вирішення задачі такого роду засновано на застосуванні теоретико-графового підходу у вигляді моделі системи з повним перекриттям загроз безпеці [17]. Модель системи захисту для механізмів управління ключами розглядається у вигляді дольного графа (рис. 1).

Кожне ребро графа G(P, O, Z, E, H) визначає дію конкретної загрози на конкретний об'єкт або нейтралізацію певним механізмом захисту загрози безпеки.

Отримання кількісного показника граф G(P, O, Z, E, H) зважується і еквівалентно представляється такою сукупністю векторів і матриць:

– вектор $P(p_1, p_2, \ldots p_N)$, де p_i – ймовірність реалізації загрози;

– вектор $O(o_1, o_2, \ldots o_L)$, де o_i – вартість об'єкту захисту;

– матриця $E\{e_{ij}\}$ розмірністю $N\times L$, де $e_{ij}=1$ якщо реалізується i-та загроза на j-й об'єкт, та $e_{ij}=0$, в іншому випадку;

– вектор $Z(z_1, z_2, \ldots z_M)$, де z_i – вартість механізму або засобу захисту;

– матриця $H\{h_{ij}\}$ розмірністю $N\times M$, де h_{ij} ймовірність усунення i-ї загрози від застосування j-го механізму або засобу захисту.

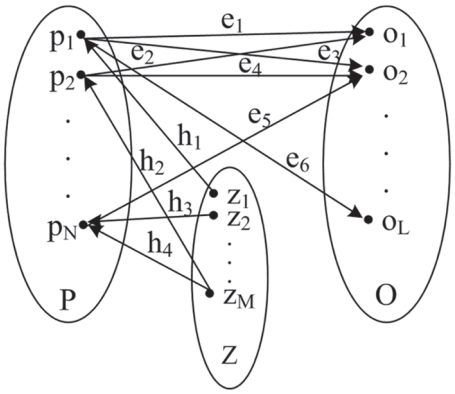


Рис. 1. Модель системи захисту системи управління ключами у вигляді тридольного графа

З використанням розглянутої моделі, можна провести аналіз ефективності системи захисту системи управління ключами на основі кількісного критерію, що відображає вплив застосованих методів і засобів захисту на зміну величини збитків від загроз безпеці. Як такий критерій зокрема, можна використовувати наступний показник економічної ефективності застосування механізму захисту:

$$T = \frac{U - U`}{\sum_{i=1}^{L} o_i + \sum_{k=1}^{M} z_k}, \qquad (1)$$

де U – оцінка величини збитку від загроз безпеки за відсутності засобів і механізмів захисту; $U`$ – оцінка величини залишкового збитку в ході застосування засобів і механізмів захисту.

Значення U та U` може бути розраховано таким чином:

$$U = \sum_{i=1}^{L} o_i (1 - \prod_{j=1}^{N} e_{ij}(1 - p_j)), \qquad (2)$$

$$U` = \sum_{i=1}^{L} o_i (1 - \prod_{j=1}^{N} e_{ij}(1 - p_j(\prod_{k=1}^{M}(1 - h_{jk})))). \qquad (3)$$

Залишкова ймовірність реалізації всіх загроз у системі може бути обчислена таким чином:

$$P`_{\text{залиш.}} = 1 - \prod_{i=1}^{N}(1 - p_i(\prod_{k=1}^{M}(1 - h_{ik}))). \qquad (4)$$

Практичне застосування наведених співвідношень потребує застосування експертних методів оцінки ймовірностей реалізації загроз безпеці та значення h_{ij}.

Реалізація системного підходу щодо аналізу ймовірності загроз, рівню збитків, що вони завдають, а також ефективності застосованих механізмів управління ключами потребує долучення компетентних експертів, які добре знайомі з предметом експертизи, а також володіють достатнім досвідом і здатні виносити об'єктивні судження. Для зменшення впливу суб'єктивного фактора, що існує при долученні експертів до аналізу, пропонується використовувати підхід до нормування отриманих оцінок від експертів за рахунок коефіцієнта їх компетентності – $K_j^{(l)}$ [16]. Оцінка рівня компетентності експертів може бути проведена з використанням документального методу, який передбачає оцінку компетентності на основі таких документальних даних, як число наукових праць в області оцінюваної проблеми, науковий ступінь, стаж практичної роботи, займана посада, тестовий метод – оцінка компетентності формується на підставі вирішення експертами тестових задач, у яких відбита специфіка предмета експертизи, а також метод взаємної і самооцінки експертів – сутність якого полягає в оцінці компетентності експертом кожного з своїх колег, а також своєї власної компетентності [16].

5. ПОРЯДОК ЗАСТОСУВАННЯ МЕТОДИКИ

Запропонована методика направлена на отримання комплексної оцінки механізмів управління ключами в середовищі хмари та складається з таких кроків:

– збір та визначення вхідних даних;

– визначення переліку необов'язкових вимог на основі вхідних даних;

– визначити модель загроз та модель порушника для ключів та ключових даних у системі хмарних обчислень;

– з урахуванням наявних статистичних даних та оцінок експертів визначити ймовірності реалізації загроз;

– визначити можливі збитки та їх вартість;

– визначення відповідності механізмів висунутим обов'язковим та визначеним необов'язковим вимогам;

– визначити ефективність застосування механізмів управління ключами;

– критерії та показники механізмів управління ключами та їх значущість;

– враховуючи отримані результати обрати найбільш ефективний механізм управління ключовими даними.

З метою вирішення задачі оцінки та обрання механізму управління ключами, скористаємося системним підходом.

Вхідними даними під час проведення аналізу є такі:

– об'єкт де розташовані потужності, що забезпечують функціонування та розгортання механізму управління ключовими даними;

– модель розгортання хмари (приватна, публічна, гібридна) та модель надання послуг (SaaS, PaaS, IaaS);

– характеристики механізмів управління ключами, а саме показники продуктивності (ПП), технічні показники (ТП), показники можливості розширення (ПМР), економічні показники (ЕП);

– вартість ресурсу, що захищається.

Якщо модель розгортання хмари не задана, рекомендується розглядати публічну хмару, яка має найбільшу кількість та ймовірність реалізації загроз, порівняно з іншими існуючими моделями, до яких можна перейти, зменшивши кількість загроз, що існує для публічної хмари. Аналогічно, якщо не задана модель надання послуг, слід розглядати можливості з надання послуги на різних рівнях.

Характеристики окремих елементів, що входять до складу механізмів управління ключами, мають бути обрані з урахуванням існуючих рішень в галузі захисту інформації.

Вартість ресурсу, що захищається, визначається на основі даних, отриманих від користувача хмарних послуг. Якщо в системі існує декілька користувачів, вартість ресурсів, що захищається, задається як набір значень вартості ресурсів. При цьому, якщо вартість ресурсу набагато вища, ніж сукупна вартість хмарного сервісу, користувачу необхідно розрахувати ефективність створення приватної хмари, що дозволяє суттєво знизити ризики виникнення загроз.

Ефективність механізмів управління ключами пропонується розраховувати за економічними показниками.

Як методи для вирішення задачі обрання механізму управління ключами, пропонується використовувати метод аналізу ієрархій [15] за рахунок використання кількісних та якісних показників, а також із застосуванням методу визначення вагових коефіцієнтів на основі функцій втрат ефективності системи з використанням тільки кількісних показників [16]. При подальшому прийнятті рішення з використання механізму управління ключами особа, що приймає рішення залежно від поставленої задачі має застосовуватись один з критеріїв прийняття рішень [16].

ВИСНОВКИ

Аналіз публікацій, стандартів та патентів в області хмарних обчислень, а також проведені дослідження показали, що для управління ключами зі сторони користувача можна застосовувати механізми управління ключами з використанням: сертифікатів відкритих ключів; паролів; HSM хмарного ЦОД; HSM користувача; криптографічного сервісу та захищеного сховища ключів.

Реалізація системного підходу до аналізу ймовірності загроз, які дорівнюють рівню збитків, що вони завдають, а також ефективності застосованих механізмів управління ключами потребує долучення компетентних експертів, які добре знайомі з предметом експертизи, а також володіють достатнім досвідом і здатні виносити об'єктивні судження. Для зменшення впливу суб'єктивного фактора, що існує при долученні експертів до аналізу, слід використовувати підхід, щодо нормування отриманих оцінок від експертів за рахунок коефіцієнта їх компетентності.

Ефективність механізмів управління ключами пропонується розраховувати за економічними показниками. Як методи для вирішення задачі обрання механізму управління ключами, пропонується використовувати метод аналізу ієрархій та метод на основі функцій втрат ефективності системи.

Література

[1] *Chandramouli, R.* NIST Cryptographic Key Management Issues & Challenges in Cloud Services. / R. Chandramoli, S. Chokhani, M. Iorga, // Режим доступу: http://nvlpubs.nist.gov/nistpubs/ir/2013/NIST.IR.7956.pdf. http://dx.doi.org/10.6028/NIST.IR.7956

[2] *Damgård I.* Secure Key Management in the Cloud / I. Damgård, T. P. Jakobsen, J. B. Nielsen, J. I. Pagter // Cryptography and Coding Lecture Notes, Springer, v. 8306. – P. 270-289. – 2013.

[3] *Choo, K.* A Cloud Security Risk-Management Strategy. / K. Choo // Cloud Computing, IEEE, – 2014. – Vol. 1 (2). – P. 52–56.

[4] *Tysowski, P.K.* Cloud-hosted key sharing towards secure and scalable mobile applications in clouds / P.K Tysowski, M.A. Hasan // Networking and Communications, – 449-455 pp. – 2013.

[5] Tysowski, P.K. Hybrid Attribute- and Re-Encryption-Based Key Management for Secure and Scalable Mobile Applications in Clouds, / P.K Tysowski, M.A. Hasan // IEEE Transactions on , vol. 1, – 172-186pp., – 2013.

[6] Cloud key management system: United States Patent Application Publication. / Pub. No.: US 2014/0050317 A1 / Authors: Jason Allen Sabin, Lehi, UT (US). Режим доступу: https://www.google.com/patents/US20140050317

[7] Cloud key management system: United States Patent Application Publication. / Pub. No.: US 20140019753

A1 / Authors: J John Houston Lowry, Jonathan A. Rubin. Режим доступу: https://www.google.com/patents/US20140019753

[8] Cloud-based hardware security modules: United States Patent Application Publication. / US 20130179676 A1 / Authors: Laurence Hamid. Режим доступу: https://www.google.com/patents/US20130179676

[9] Cloud key management: United States Patent Application Publication. / US 20140019753 A1/ Authors: John Houston Lowry, Jonathan A. Rubin. Режим доступу: http://www.google.com/patents/US20140019753

[10] Remote key management in a cloud-based environment: United States Patent Application Publication. / US 20140270178 A1 / Authors: Andy Kiang, Chris Byron, Jeffrey Queisser. Режим доступу: http://www.google.com/patents/US20140270178

[11] Secure Cloud Storage and Encryption Management System: United States Patent Application Publication. / US 20140281477 A1 / Authors: Alex Nayshtut, Edward Jimison, Omer Ben-Shalom, Michael Raziel. Режим доступу: http://www.google.com/patents/US20140281477

[12] Virtual key management and isolation of data deployments in multi-tenant environments: United States Patent Application Publication. / US 20140283010 A1 / Authors: Matthew Francis Rutkowski, Ronald W. Bassett, Thomas Alexander. Bellwood. Режим доступу: http://www.google.com/patents/US20140283010

[13] Cloud-based key management: United States Patent Application Publication. / US 20150172260 A1 / Authors: Stephan Brenner. Режим доступу: http://www.google.com/patents/US20150172260

[14] *Аулов І.Ф.* Дослідження моделі загроз ключових систем хмари та пропозиції захисту від них / І.Ф. Аулов // Восточно-Европейский журнал передовых технологий. – 2015. – № 5/2 (77). – С. 4–13.

[15] *Саати Т.* Принятие решений. Метод анализа иерархий. – М: Радио и связь, 1999. – 341 с.

[16] *Медіченко М.П., Потій О.В.* Основи теорії систем та системного аналізу. – Харків: ХВУ, 2002. – 258 с.

[17] *Гайдамакин, Николай Александрович*. Учебно-методический комплекс дисциплины «Теоретические основы компьютерной безопасности» [Электронный ресурс] / Н. А. Гайдамакин ; Федер. агентство по образованию, Урал. гос. ун-т им. А. М. Горького, ИОНЦ «Информационная безопасность» [и др.]., Екатеринбург. – 2008.

Надійшла до редколегії 25.11.2015

Аулов Іван Федорович, молодший науковий співробітник кафедри БІТ ХНУРЕ. Наукові інтереси: дослідження безпеки хмарних обчислень.

УДК 681.3.06

Комплексный метод сравнения, оценки и выбора механизмов управления ключевыми данными пользователя в облачной ИТС / И.Ф. Аулов // Прикладная радиоэлектроника: научн.-техн. журнал. – 2015. – Том 14. – № 4. – С. 397–402.

Предложено методику, которая позволяет провести комплексное сравнение механизмов управления ключевыми данными пользователей в облачных ИТС по совокупности критериев и показателей. Приводится обоснование критериев и показателей оценки механизмов управления ключевыми данными пользователей в облаке.

Ключевые слова: механизмы управления, ключевые данные, облачные ИТС, сравнительный анализ.

Рис.: 1. Библиогр.:17 назв.

UDC 681.3.06

Complex method of comparison, assessment and selection of key data management mechanisms in a cloud ITS / I.F. Aulov // Applied Radio Electronics: Sci. Journ. – 2015. – Vol. 14. – № 4. – P. 397–402.

The paper proposes a technique, which allows to make complex comparative analysis of user key management mechanisms in a cloud ITS using a collection of criteria and indicators. The substantiation of criteria and indicators for assessing user key data management mechanisms in the cloud is given.

Keywords: control mechanisms, key data, cloud ITS, comparative analysis.

Fig.: 1. Ref.: 17 items.

К 70-летию со дня рождения Ивана Дмитриевича ГОРБЕНКО

21 января 2016 года исполняется 70 лет со дня рождения доктора технических наук, профессора, академика Академии наук прикладной радиоэлектроники, профессора кафедры безопасности информационных систем и технологий Харьковского национального университета им. В.Н. Каразина **Горбенко Ивана Дмитриевича.**

С 1967 года судьба Ивана Дмитриевича связана с Харьковским высшим командно-инженерным училищем им. маршала Советского Союза Н.И. Крылова, в котором он учился, а затем учил других, передавая им свои знания и опыт. Настойчиво работал над собой, совершенствуя свою научную и педагогическую квалификацию, стал одним из лучших методистов ХВКИУ. Участвовал в постановке и чтении ряда фундаментальных курсов, является автором признанных учебников и учебных пособий, своим примером оказывал большое влияние на формирование не одного поколения офицеров-ракетчиков. Здесь стал кандидатом наук, доцентом, а затем доктором наук и профессором, возглавил одну из лучших кафедр «Радиосистем боевого управления и безопасности информации».

После увольнения из Вооружённых Сил с 1992 года И.Д. Горбенко успешно вел научно-педагогическую деятельность в Харьковском национальном университете радиоэлектроники. Благодаря высокому научному авторитету и личной инициативе Ивана Дмитриевича в университете была создана новая кафедра «Безопасность информационных технологий», которую он успешно возглавлял более 15 лет. За это время удалось создать дружный и трудолюбивый коллектив коллег-единомышленников, внести огромный вклад в подготовку и воспитание ученых в области криптографической защиты информации. За короткий срок кафедра вошла в число лучших кафедр университета.

Как замечательный педагог и методист высшей квалификации Иван Дмитриевич отличается исключительным трудолюбием, целеустремленностью, умением быстро и глубоко разбираться в новейших достижениях в области науки и техники и с большим талантом донести до обучаемых глубинную суть изучаемых вопросов. Трудно найти место в странах бывшего Союза, где бы не трудились его ученики или последователи.

У научной общественности Украины Иван Дмитриевич пользуется большим авторитетом. Награжден нагрудным знаком «Відмінник освіти України», является лауреатом государственной премии Украины в области науки и техники, членом двух специализированных докторских советов, экспертом секции «Национальная безопасность» и методической комиссии Министерства образования и науки Украины, членом редколлегии 5 научно-технических журналов и сборников.

С 1995 года Горбенко Иван Дмитриевич является Председателем правления, главным конструктором Закрытого акционерного общества «Институт информационных технологий». Под его непосредственным руководством созданы и внедрены более 40 систем и средств защиты информации, налажено плодотворное сотрудничество ИИТ с профилирующей кафедрой и с АН ПРЭ. Как ученый и специалист в области защиты информации и информационной безопасности он пользуется огромным авторитетом в Украине, является разработчиком важнейших нормативных документов.

Мы гордимся тем, что действительным членом Академии наук прикладной радиоэлектроники и членом её Президиума является такой разносторонний учёный, автор более 150 научных трудов, из которых 10 монографий, 10 учебников и учебных пособий, более 100 авторских свидетельств и патентов на изобретения, создавший научную школу, в которой подготовлены 6 докторов и более 30 кандидатов наук.

Как руководитель ИИТ, который является коллективным членом АН ПРЭ, И.Д. Горбенко оказывает регулярную финансовую помощь Академии. Тематические выпуски журнала «Прикладная радиоэлектроника», которые организованы им, играют большую роль и пользуются большой популярностью среди преподавателей, научных работников и специалистов.

Убежден, что опыт, знания и конкретные действия Ивана Дмитриевича принесут в дальнейшем большую пользу Академии, росту ее авторитета не только в Украине, но и за ее пределами.

Члены редакционного совета журнала «Прикладная радиоэлектроника» вместе со всей научной общественностью сердечно поздравляют Вас, дорогой Иван Дмитриевич, с 70-летним юбилеем, желают здоровья, семейного благополучия, дальнейших творческих успехов.

Президиум АН ПРЭ

ПРИКЛАДНАЯ РАДИОЭЛЕКТРОНИКА

Научно-технический журнал

Ответственный секретарь

Е. Б. Исаева

Корректор

Б. П. Косиковская

Перевод на английский язык

К. Т. Умяров

Компьютерный дизайн и верстка

Е. Б. Исаева

Рекомендовано засіданням Бюро Президії Академії наук прикладної радіоелектроніки
(протокол № 4 від 29.12.2015 р.).

Рекомендовано Вченою радою Харківського національного університету радіоелектроніки
(протокол № 10 від 25.12.2015 р.)

Свідоцтво про державну реєстрацію КВ № 6037 від 09.04.2002 р.

Журнал включений до списку фахових видань ВАК України
з технічних наук
(постанова президії ВАК України № 1-05/2 від 10.03.2010),
з фізико-математичних наук (фізика)
(постанова президії ВАК України № 1-05/5 від 1.07.2010)

Підписано до друку 29.12.2015. Формат 60 × 84 $^{1}/_{8}$.
Папір офсет. Друк офсет. Умов.-друк. арк. 15,8. Облік.-вид. арк. 15,2.
Тираж 300 прим. Ціна договірна.

Віддруковано в ТОВ «ДРУКАРНЯ МАДРИД»
61024, м. Харків, вул. Максимільянівська, 11. Тел.: (057) 756-53-25
www.madrid.in.ua, e-mail: info@madrid.in.ua