

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ Функция хэширования

Information technology.
Cryptographic data security.
Hash function

Дата введения - 20__ - __ - __

1 Область применения

Настоящий стандарт определяет алгоритм и процедуру вычисления хэш-функции для любой последовательности двоичных символов, которые применяются в криптографических методах обработки и защиты информации, в том числе для реализации процедур обеспечения целостности, аутентичности, электронной цифровой подписи (ЭЦП) при передаче, обработке и хранении информации в автоматизированных системах.

Определенная в настоящем стандарте функция хэширования используется при реализации систем электронной цифровой подписи на базе ассиметричного криптографического алгоритма по ГОСТ Р 34.10-20__ (проект).

Стандарт рекомендуется использовать в новых системах обработки информации различного назначения, а также при модернизации действующих систем.

2 Нормативные ссылки

В настоящем стандарте использованы ссылки на следующие стандарты:

ГОСТ 28147-89 Системы обработки информации. Защита криптографическая. Алгоритмы криптографического преобразования

ГОСТ Р 34.10-20__ (проект) Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи

П р и м е ч а н и е – При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования – на официальном сайте Федерального агентства Российской Федерации по техническому регулированию и метрологии в сети Интернет или по ежегодно издаваемому информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по соответствующим ежемесячно издаваемым информационным указателям, опубликованным в текущем году. Если ссылочный стандарт заменен (изменен), то при пользовании настоящим стандартом, следует руководствоваться замененным (измененным) стандартом. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, применяется в части, не затрагивающей эту ссылку.

3 Определения и обозначения

3.1 Определения

В настоящем стандарте применены следующие термины с соответствующими определениями:

3.1.1

дополнение (appendix): Строка бит, формируемая из цифровой подписи и произвольного текстового поля. [ИСО/МЭК 14888-1].

3.1.2

заполнение (padding): Приписывание дополнительных бит к строке бит. [ИСО/МЭК 10118-1]

3.1.3

инициализационный вектор (initializing value): Вектор, определенный как начальная точка работы функции хэширования. [ИСО/МЭК 10118-1]

3.1.4

последовательность псевдослучайных чисел (pseudo-random number sequence): Последовательность чисел, полученная в результате выполнения некоторого арифметического (вычислительного) процесса, используемая в конкретном случае вместо последовательности случайных чисел. [ИСО 2382-2]

3.1.5

последовательность случайных чисел (random number sequence): Последовательность чисел, каждое из которых не может быть предсказано (вычислено) только на основе знания предшествующих ему чисел данной последовательности. [ИСО 2382-2].

3.1.6

случайное число (random number): Число, выбранное из определенного набора чисел таким образом, что каждое число из данного набора может быть выбрано с одинаковой вероятностью. [ИСО 2382-2].

3.1.7

сообщение (message): Строка бит ограниченной длины [ИСО/МЭК 9796].

3.1.8

функция сжатия (round function): Итеративно используемая функция, преобразующая строку бит длины L_1 и полученную на предыдущем шаге строку бит длины L_2 в строку бит длины L_2 . [ИСО/МЭК 10118-1]

3.1.9

хэш-код (hash-code): Строка бит, являющаяся выходным результатом хэш-функции. [ИСО/МЭК 10118-1].

3.1.10

хэш-функция (hash-function): Функция, отображающая строки бит в строки бит фиксированной длины и удовлетворяющая следующим свойствам:
1 по данному значению функции сложно вычислить исходные данные, отображенные в это значение;
2 для заданных исходных данных трудно найти другие исходные данные, отображаемые с тем же результатом;
3 трудно найти какую-либо пару исходных данных с одинаковым значением хэш-функции.
[ИСО/МЭК 10118-1]

П р и м е ч а н и е – Применительно к области электронной цифровой подписи свойство 1 подразумевает, что по известной электронной цифровой подписью невозможно восстановить исходное сообщение; свойство 2 подразумевает, что для заданного подписанного сообщения трудно подобрать другое (фальсифицированное) сообщение, имеющее ту же электронную цифровую подпись, свойство 3 подразумевает, что трудно подобрать какую-либо пару сообщений, имеющих одну и ту же подпись.

3.1.11

[электронная] цифровая подпись (digital signature); ЭЦП: Строка бит, полученная в результате процесса формирования подписи. Данная строка имеет внутреннюю структуру, зависящую от конкретного механизма формирования подписи. [ИСО/МЭК 14888-1]

П р и м е ч а н и е – В настоящем стандарте в целях сохранения терминологической преемственности с действующими отечественными нормативными документами и опубликованными научно-техническими изданиями установлено, что термины «цифровая подпись» и «электронная цифровая подпись (ЭЦП)» являются синонимами.

3.2 Обозначения

V^*	множество всех двоичных векторов-строк конечной размерности (далее – векторов), включая пустую строку;
$ A $	размерность (число компонент) вектора $A \in V^*$ (если A – пустая строка, то $ A = 0$);
V_n	множество всех n -мерных двоичных векторов, где n – целое неотрицательное число; нумерация подвекторов и компонент вектора осуществляется справа налево начиная с нуля;
\oplus	операция покомпонентного сложения по модулю 2 двух двоичных векторов одинаковой размерности;
$A\ B$	конкатенация векторов $A, B \in V^*$, т.е. вектор из $V_{ A + B }$, в котором левый подвектор из $V_{ A }$ совпадает с вектором A , а правый подвектор из $V_{ B }$ совпадает с вектором B ;
A^n	конкатенация n экземпляров вектора A ;
\mathbb{Z}_2^n	кольцо вычетов по модулю 2^n ;
\boxplus	операция сложения в кольце \mathbb{Z}_2^n ;
$\text{Vec}_n : \mathbb{Z}_2^n \rightarrow V_n$	биективное отображение, сопоставляющее целому числу из \mathbb{Z}_2^n его двоичное представление, т. е. для любого элемента $z = z_0 + 2 \cdot z_1 + \dots + 2^{n-1} \cdot z_{n-1}$ кольца \mathbb{Z}_2^n , где $z_j \in \{0, 1\}, j = 0, \dots, n - 1$, выполнено равенство $\text{Vec}_n(z) = z_{n-1}\ \dots \ z_1\ z_0$;
$\text{Int}_n : V_n \rightarrow \mathbb{Z}_2^n$	отображение, обратное к отображению Vec_n , т.е. $\text{Int}_n = \text{Vec}_n^{-1}$;
$\text{MSB}_n : V^* \rightarrow V_n$	отображение, ставящее в соответствие вектору $z_{k-1}\ \dots \ z_1\ z_0$, $k \geq n$, вектор $z_{k-1}\ \dots \ z_{k-n+1}\ z_{k-n}$
$a := b$	операция присваивания переменной a значения b ;
$\Phi\Psi$	произведение отображений, при котором отображение Ψ действует первым;
M	двоичный вектор, подлежащий хэшированию, $M \in V^*, M < 2^{512}$;
$H : V^* \rightarrow V_{512}$	функция хэширования, отображающая вектор (сообщение) M в вектор (хэш-код) $H(M)$;
IV	инициализационный вектор функции хэширования, $IV \in V_{512}$.

4 Общие положения

Под хэш-функцией H понимается зависящее от инициализационного вектора IV отображение

$$H: V^* \rightarrow V_n,$$

сопоставляющее строке бит произвольной длины хэш-код фиксированной длины n .

Для определения хэш-функции необходимы описание процедуры инициализации, алгоритма вычисления функции сжатия и итеративной процедуры вычисления значения хэш-функции.

Данный стандарт определяет две функции хэширования – «Стрибог-256» и «Стрибог-512» – с длинами хэш-кода 256 и 512 бит соответственно.

5 Значения параметров

5.1 Инициализационные векторы

Значение инициализационного вектора IV для хэш-функции «Стрибог-512» полагаем равным 0^{512} . Значение инициализационного вектора IV для хэш-функции «Стрибог-256» полагаем равным $(00000001)^{64}$.

5.2 Нелинейное биективное преобразование множества двоичных векторов

В качестве нелинейного биективного преобразования множества двоичных векторов V_8 выступает подстановка $\pi = \text{Vec}_8 \pi' \text{Int}_8 : V_8 \rightarrow V_8$, где $\pi' : \mathbb{Z}_2^8 \rightarrow \mathbb{Z}_2^8$. Значения подстановки π' записаны ниже в виде массива $\pi' = (\pi'(0), \pi'(1), \dots, \pi'(255))$:

$\pi' = (252, 238, 221, 17, 207, 110, 49, 22, 251, 196, 250, 218, 35, 197, 4, 77, 233, 119, 240, 219, 147, 46, 153, 186, 23, 54, 241, 187, 20, 205, 95, 193, 249, 24, 101, 90, 226, 92, 239, 33, 129, 28, 60, 66, 139, 1, 142, 79, 5, 132, 2, 174, 227, 106, 143, 160, 6, 11, 237, 152, 127, 212, 211, 31, 235, 52, 44, 81, 234, 200, 72, 171, 242, 42, 104, 162, 253, 58, 206, 204, 181, 112, 14, 86, 8, 12, 118, 18, 191, 114, 19, 71, 156, 183, 93, 135, 21, 161, 150, 41, 16, 123, 154, 199, 243, 145, 120, 111, 157, 158, 178, 177, 50, 117, 25, 61, 255, 53, 138, 126, 109, 84, 198, 128, 195, 189, 13, 87, 223, 245, 36, 169, 62, 168, 67, 201, 215, 121, 214, 246, 124, 34, 185, 3, 224, 15, 236, 222, 122, 148, 176, 188, 220, 232, 40, 80, 78, 51, 10, 74, 167, 151, 96, 115, 30, 0, 98, 68, 26, 184, 56, 130, 100, 159, 38, 65, 173, 69, 70, 146, 39, 94, 85, 47, 140, 163, 165, 125, 105, 213, 149, 59, 7, 88, 179, 64, 134, 172, 29, 247, 48, 55, 107, 228, 136, 217, 231, 137, 225, 27, 131, 73, 76, 63, 248, 254, 141, 83, 170, 144, 202, 216, 133, 97, 32, 113, 103, 164, 45, 43, 9, 91, 203, 155, 37, 208, 190, 229, 108, 82, 89, 166, 116, 210, 230, 244, 180, 192, 209, 102, 175, 194, 57, 75, 99, 182).$

5.3 Перестановка байт

Значения перестановки $\tau \in S_{64}$ записаны ниже в виде массива $\tau = (\tau(0), \tau(1), \dots, \tau(63))$:

$\tau = (0, 8, 16, 24, 32, 40, 48, 56, 1, 9, 17, 25, 33, 41, 49, 57, 2, 10, 18, 26, 34, 42, 50, 58, 3, 11, 19, 27, 35, 43, 51, 59, 4, 12, 20, 28, 36, 44, 52, 60, 5, 13, 21, 29, 37, 45, 53, 61, 6, 14, 22, 30, 38, 46, 54, 62, 7, 15, 23, 31, 39, 47, 55, 63).$

5.4 Линейное преобразование множества двоичных векторов

Линейное преобразование l множества двоичных векторов V_{64} задается умножением справа на матрицу A над полем $GF(2)$, строки которой записаны ниже последовательно в шестнадцатеричном виде. То есть, строка матрицы с номером j , $j = 0, \dots, 63$, записанная в виде $a_{j,15} \dots a_{j,0}$, где $a_{j,i} \in \mathbb{Z}_{16}$, $i = 0, \dots, 15$, есть $\text{Vec}_4(a_{j,15}) \parallel \dots \parallel \text{Vec}_4(a_{j,0})$.

Здесь в одной строке записано 4 строки матрицы A , при этом в строке с номером i , $i = 0, \dots, 15$, записаны строки матрицы A с номерами $4 \cdot i + j$, $j = 0, \dots, 3$, в следующем порядке (слева направо) $4 \cdot i + 0$, $4 \cdot i + 1$, $4 \cdot i + 2$, $4 \cdot i + 3$:

8e20faa72ba0b470	47107ddd9b505a38	ad08b0e0c3282d1c	d8045870ef14980e
6c022c38f90a4c07	3601161cf205268d	1b8e0b0e798c13c8	83478b07b2468764
a011d380818e8f40	5086e740ce47c920	2843fd2067adea10	14aff010bdd87508
0ad97808d06cb404	05e23c0468365a02	8c711e02341b2d01	46b60f011a83988e
90dab52a387ae76f	486dd4151c3dfdb9	24b86a840e90f0d2	125c354207487869
092e94218d243cba	8a174a9ec8121e5d	4585254f64090fa0	accc9ca9328a8950
9d4df05d5f661451	c0a878a0a1330aa6	60543c50de970553	302a1e286fc58ca7
18150f14b9ec46dd	0c84890ad27623e0	0642ca05693b9f70	0321658cba93c138
86275df09ce8aaa8	439da0784e745554	afc0503c273aa42a	d960281e9d1d5215
e230140fc0802984	71180a8960409a42	b60c05ca30204d21	5b068c651810a89e
456c34887a3805b9	ac361a443d1c8cd2	561b0d22900e4669	2b838811480723ba
9bcf4486248d9f5d	c3e9224312c8c1a0	effa11af0964ee50	f97d86d98a327728
e4fa2054a80b329c	727d102a548b194e	39b008152acb8227	9258048415eb419d
492c024284fbaec0	aa16012142f35760	550b8e9e21f7a530	a48b474f9ef5dc18
70a6a56e2440598e	3853dc371220a247	1ca76e95091051ad	0edd37c48a08a6d8
07e095624504536c	8d70c431ac02a736	c83862965601dd1b	641c314b2b8ee083

Результат умножения вектора $b = b_{63} \dots b_0 \in V_{64}$ на матрицу A есть вектор $c \in V_{64}$:

$$c = b_{63} \cdot (\text{Vec}_4(a_{0,15}) \parallel \dots \parallel \text{Vec}_4(a_{0,0})) \oplus \dots \oplus b_0 \cdot (\text{Vec}_4(a_{63,15}) \parallel \dots \parallel \text{Vec}_4(a_{63,0}))$$

где

$$b_i \cdot (\text{Vec}_4(a_{63-i,15}) \parallel \dots \parallel \text{Vec}_4(a_{63-i,0})) = \begin{cases} 0^{64}, & \text{если } b_i = 0, \\ (\text{Vec}_4(a_{63-i,15}) \parallel \dots \parallel \text{Vec}_4(a_{63-i,0})) & \text{если } b_i = 1, \end{cases}$$

для всех $i = 0, \dots, 63$.

5.5 Итерационные константы

Итерационные константы записаны в шестнадцатеричном виде. То есть значение константы, записанное в виде $a_{127} \dots a_0$, где $a_i \in \mathbb{Z}_{16}$, $i = 0, \dots, 127$, есть $\text{Vec}_4(a_{127}) \parallel \dots \parallel \text{Vec}_4(a_0)$:

$C_1 = \text{b1085bda1ecadae9ebcb2f81c0657c1f2f6a76432e45d016714eb88d7585c4fc4b7ce09192676901a2422a08a460d31505767436cc744d23dd806559f2a64507};$
 $C_2 = \text{6fa3b58aa99d2f1a4fe39d460f70b5d7f3feea720a232b9861d55e0f16b501319ab5176b12d699585cb561c2db0aa7ca55dda21bd7cbcd56e679047021b19bb7};$
 $C_3 = \text{f574dcac2bce2fc70a39fc286a3d843506f15e5f529c1f8bf2ea7514b1297b7bd3e20fe490359eb1c1c93a376062db09c2b6f443867adb31991e96f50aba0ab2};$
 $C_4 = \text{ef1fdfb3e81566d2f948e1a05d71e4dd488e857e335c3c7d9d721cad685e353fa9d72c82ed03d675d8b71333935203be3453eaa193e837f1220cbebc84e3d12e};$
 $C_5 = \text{4bea6bacad4747999a3f410c6ca923637f151c1f1686104a359e35d7800fffbdbfcd1747253af5a3dfff00b723271a167a56a27ea9ea63f5601758fd7c6cfe57};$
 $C_6 = \text{ae4faeae1d3ad3d96fa4c33b7a3039c02d66c4f95142a46c187f9ab49af08ec6cffa6b71c9ab7b40af21f66c2bec6b6bf71c57236904f35fa68407a46647d6e};$
 $C_7 = \text{f4c70e16eeaac5ec51ac86feb240954399ec6c7e6bf87c9d3473e33197a93c90992abc52d822c3706476983284a05043517454ca23c4af38886564d3a14d493};$
 $C_8 = \text{9b1f5b424d93c9a703e7aa020c6e41414eb7f8719c36de1e89b4443b4ddbc49af4892bcb929b069069d18d2bd1a5c42f36acc2355951a8d9a47f0dd4bf02e71e};$
 $C_9 = \text{378f5a541631229b944c9ad8ec165fde3a7d3a1b258942243cd955b7e00d0984800a440bdbb2ceb17b2b8a9aa6079c540e38dc92cb1f2a607261445183235adb};$
 $C_{10} = \text{abbedea680056f52382ae548b2e4f3f38941e71cff8a78db1fffe18a1b3361039fe76702af69334b7a1e6c303b7652f43698fad1153bb6c374b4c7fb98459ced};$
 $C_{11} = \text{7bcd9ed0efc889fb3002c6cd635afe94d8fa6bbbebab076120018021148466798a1d71fe48b9caefbacd1d7d476e98dea2594ac06fd85d6bcaa4cd81f32d1b};$
 $C_{12} = \text{378ee767f11631bad21380b00449b17acda43c32bcdf1d77f82012d430219f9b5d80ef9d1891cc86e71da4aa88e12852faf417d5d9b21b9948bc924af11bd720}.$

6 Преобразования

При вычислении хэш-кода $H(M)$ сообщения $M \in V^*$ используются следующие преобразования:

$$X[k] : V_{512} \rightarrow V_{512}, \quad X[k](a) = k \oplus a, \quad k, a \in V_{512};$$

$$S : V_{512} \rightarrow V_{512}, \quad S(a) = S(a_{63} \| \dots \| a_0) = \pi(a_{63}) \| \dots \| \pi(a_0), \quad \text{где } a = a_{63} \| \dots \| a_0 \in V_{512},$$

$$a_i \in V_8, \quad i = 0, \dots, 63;$$

$$P : V_{512} \rightarrow V_{512}, \quad P(a) = P(a_{63} \| \dots \| a_0) = a_{\tau(63)} \| \dots \| a_{\tau(0)}, \quad \text{где } a = a_{63} \| \dots \| a_0 \in V_{512},$$

$$a_i \in V_8, \quad i = 0, \dots, 63;$$

$$L : V_{512} \rightarrow V_{512}, \quad L(a) = L(a_7 \| \dots \| a_0) = l(a_7) \| \dots \| l(a_0), \quad \text{где } a = a_7 \| \dots \| a_0 \in V_{512},$$

$$a_i \in V_{64}, \quad i = 0, \dots, 7;$$

7 Функция сжатия

Значение хэш-кода сообщения $M \in V^*$ вычисляется с использованием итерационной процедуры. При этом на каждой итерации вычисления хэш-кода используется функция сжатия:

$$g_N : V_{512} \times V_{512} \rightarrow V_{512}, N \in V_{512},$$

значение которой вычисляется следующим образом:

$$g_N(h, m) = E(LPS(h \oplus N), m) \oplus h \oplus m,$$

где

$$E(K, m) = X[K_{13}]LPSX[K_{12}] \dots LPSX[K_2]LPSX[K_1](m),$$

вычисление значений $K_i \in V_{512}$, $i = 1, \dots, 13$, осуществляется следующим образом:

$$K_1 = K;$$

$$K_i = LPS(K_{i-1} \oplus C_{i-1}), \quad i = 2, \dots, 13.$$

8 Процедура вычисления хэш-функции

Исходными данными для процедуры вычисления хэш-кода $H(M)$ является подлежащее хэшированию сообщение $M \in V^*$ и $IV \in V_{512}$ – инициализационный вектор хэширования.

Алгоритм вычисления функции H состоит из следующих этапов:

8.1 Этап 1.

Присвоить начальные значения текущих величин:

- 1.1. $h := IV$;
- 1.2. $N := 0^{512} \in V_{512}$;
- 1.3. $\Sigma := 0^{512} \in V_{512}$;
- 1.4. Перейти к этапу 2.

8.2 Этап 2.

- 2.1. Проверить условие $|M| < 512$.

При положительном исходе перейти к этапу 3.

В противном случае выполнить последовательность вычислений:

- 2.2. Вычислить подвектор $m \in V_{512}$ сообщения M : $M = M' \parallel m$. Далее выполнить последовательность вычислений:

- 2.3. $h := g_N(h, m)$;
- 2.4. $N := \text{Vec}_{512}(\text{Int}_{512}(N) \boxplus 512)$;

2.5. $\Sigma := \text{Vec}_{512} (\text{Int}_{512}(\Sigma) \boxplus \text{Int}_{512}(m))$;

2.6. $M := M'$

2.7. Перейти к шагу 2.1.

8.3 Этап 3.

3.1. $m := 0^{511 - |M|} \parallel 1 \parallel M$;

3.2. $h := g_N(h, m)$;

3.3. $N := \text{Vec}_{512} (\text{Int}_{512}(N) \boxplus |M|)$;

3.4. $\Sigma := \text{Vec}_{512} (\text{Int}_{512}(\Sigma) \boxplus \text{Int}_{512}(m))$;

3.5. $h := g_0(h, N)$;

3.6. $h := \begin{cases} g_0(h, \Sigma), & \text{для хэш – функции «Стрибог – 512»} \\ MSB_{256}(g_0(h, \Sigma)), & \text{для хэш – функции «Стрибог – 256»} \end{cases}$;

3.7. Конец работы алгоритма.

Значение величины h , полученное на шаге 3.6, является значением функции хэширования $H(M)$.

$K_1 =$ b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574
b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574.

$X[K_1](m) =$ b2b1cd1ef7ec924286b7cf1cffe49c4c84b5c91afde694448abbcb18fbe09646
82b3c516f9e2904080b1cd1ef7ec924286b7cf1cffe49c4c84b5c91afde69444

$SX[K_1](m) =$ 4645d95fc0beec2c432f8914b62d4efd3e5e37f14b097aead67de417c220b048
2492ac996667e0ebdf45d95fc0beec2c432f8914b62d4efd3e5e37f14b097aea

$PSX[K_1](m) =$ 46433ed624df433e452f5e7d92452f5ed98937e4acd989375f14f117995f14f1
c0b64bc266c0b64bbe2d092067be2d09ec4e7ab0e0ec4e7a2cfdea48eb2cfdea

$LPSX[K_1](m) =$ e60059d4d8e0758024c73f6f3183653f56579189602ae4c21e7953ebc0e212a0
ce78a8df475c2fd4fc43fc4b71c01e35be465fb20dad2cf690cdf65028121bb9

$K_1 \oplus C_1 =$ 028ba7f4d01e7f9d5848d3af0eb1d96b9ce98a6de0917562c2cd44a3bb516188
f8ff1cbf5cb3cc7511c1d6266ab47661b6f5881802a0e8576e0399773c72e073

$S(K_1 \oplus C_1) =$ ddf644e6e15f5733bff249410445536f4e9bd69e200f3596b3d9ea737d70a1d7
d1b6143b9c9288357758f8ef78278aa155f4d717dda7cb12b211e87e7f19203d

$PS(K_1 \oplus C_1) =$ ddbf4eb3d17755b2f6f29bd9b658f4114449d6ea14f8d7e8e6419e733bef177e
e104207d9c78dd7f5f450f709227a719575335a1888acb20336f96d735a1123d

$LPS(K_1 \oplus C_1) =$ d0b00807642fd78f13f2c3ebc774e80de0e902d23aef2ee9a73d010807dae9c1
88be14f0b2da27973569cd2ba051301036f728bd1d7eec33f4d18af70c46cf1e

Итерация 2

$K_2 =$ d0b00807642fd78f13f2c3ebc774e80de0e902d23aef2ee9a73d010807dae9c1
88be14f0b2da27973569cd2ba051301036f728bd1d7eec33f4d18af70c46cf1e

$LPSX[K_2]LPSX[K_1](m) =$

18e77571e703d19548075c574ce5e50e480c9c5b9f21d45611ab86cf32e352a
d91854ea7df8f863d46333673f62ff2d3efae1cd966f8e2a74ce49902799aad4

Итерация 3

$K_3 =$ 9d4475c7899f2d0bb0e8b7dac6ef6e6b44ecf66716d3a0f16681105e2d13712a
1a9387ecc257930e2d61014a1b5c9fc9e24e7d636eb1607e816dbaf927b8fca9

$LPSX[K_3] \dots LPSX[K_1](m) =$
3dc0a9c64d42543ccdb62960d58c17eb5b805d08a07406ece679d5f82b70fe
a22a7ea56e21814619e8749b308214575489d4d465539852cd4b0cd3829bef39

Итерация 4

$K_4 =$ 5c283daba5ec1f233b8c833c48e1c670dae2e40cc4c3219c73e58856bd96a72f
df9f8055ffe3c004c8cde3b8bf78f95f3370d0a3d6194ac5782487defd83ca0f

$LPSX[K_4] \dots LPSX[K_1](m) =$
dbee312ea7301b0d6d13e43855e85db81608c780c43675bc93cfd82c1b4933b3
898a35b13e1878abe119e4dff9de4889738ca74d064cd9eb732078c1fb25e04

Итерация 5

$K_5 =$ 109f33262731f9bd569cbc9317baa551d4d2964fa18d42c41fab4e37225292ec
2fd97d7493784779046388469ae195c436fa7cba93f8239ceb5ffc818826470c

$LPSX[K_5] \dots LPSX[K_1](m) =$
7fb3f15718d90e889f9fb7c38f527bec861c298afb9186934a93c9d96ade20df
109379bb9c1a1ffd0ad81fce7b45ccd54501e7d127e32874b5d7927b032de7a1

Итерация 6

$K_6 =$ b32c9b02667911cf8f8a0877be9a170757e25026ccf41e67c6b5da70b1b87474
3e1135cfbefe244237555c676c153d99459bc382573aee2d85d30d99f286c5e7

$LPSX[K_6] \dots LPSX[K_1](m) =$
95efa4e104f235824bae5030fe2d0f170a38de3c9b8fc6d8fa1a9adc2945c413
389a121501fa71a65067916b0c06f6b87ce18de1a2a98e0a64670985f47d73f1

Итерация 7

$K_7 =$ 8a13c1b195fd0886ac49989e7d84b08bc7b00e4f3f62765ece6050fcbabdc234
6c8207594714e8e9c9c7aad694edc922d6b01e17285eb7e61502e634559e32f1

$LPSX[K_7] \dots LPSX[K_1](m) =$
7ea4385f7e5e40103bfb25c67e404c7524eec43e33b1d06557469c6049854304
32b43d941b77ffd476103338e9bd5145d9c1e18b1f262b58a81dcefff6fc6535

Итерация 8

$K_8 = 52cec3b11448bb8617d0ddfb9c926f2e88730cb9179d6decea5acbffd323ec376$
 $4c47f7a9e13bb1db56c342034773023d617ff01cc546728e71dff8de5d128cac$

$LPSX[K_8] \dots LPSX[K_1](m) =$
 $b2426da0e58d5cfe898c36e797993f902531579d8ecc59f8dd8a60802241a456$
 $1f290cf992eb398894424bf681636968c167e870967b1dd9047293331956daba$

Итерация 9

$K_9 = f38c5b7947e7736d502007a05ea64a4eb9c243cb82154aa138b963bbb7f28e74$
 $d4d710445389671291d70103f48fd4d4c01fc415e3fb7dc61c6088afa1a1e735$

$LPSX[K_9] \dots LPSX[K_1](m) =$
 $5e0c9978670b25912dd1ede5bdd1cf18ed094d14c6d973b731d50570d0a9bca2$
 $15415a15031fd20ddefb5bc61b96671d6902f49df4d2fd346ceebda9431cb075$

Итерация 10

$K_{10} = 0740b3faa03ed39b257dd6e3db7c1bf56b6e18e40cdaabd30617cecbaddd618e$
 $a5e61bb4654599581dd30c24c1ab877ad0687948286cfefaa7eef99f6068b315$

$LPSX[K_{10}] \dots LPSX[K_1](m) =$
 $c1ddd840fe491393a5d460440e03bf451794e792c0c629e49ab0c1001782dd37$
 $691cb6896f3e00b87f71d37a584c35b9cd8789fad55a46887e5b60e124b51a61$

Итерация 11

$K_{11} = 185811cf3c2633aec8cfdca9dbb29347011bf92b95910a3ad71e5fca678e45$
 $e374f088f2e5c29496e9695ce8957837107bb3aa56441af11a82164893313116$

$LPSX[K_{11}] \dots LPSX[K_1](m) =$
 $3f75beaf2911c35d575088e30542b689c85b6b1607f8b800405941f5ab704284$
 $7b9b08b58b4fbdd6154ed7b366fd3ee778ce647726ddb3c7d48c8ce8866a8435$

Итерация 12

$K_{12} = 9d46bf66234a7ed06c3b2120d2a3f15e0fedd87189b75b3cd2f206906b5ee00d$
 $c9a1eab800fb8cc5760b251f4db5cdef427052fa345613fd076451901279ee4c$

$$h := IV = (00000001)^{64}$$

$$N := 0^{512}$$

$$\Sigma := 0^{512}$$

Длина сообщения $|M_1| = 504 < 512$, поэтому происходит заполнение неполного блока.

$$m := 0132313039383736353433323130393837363534333231303938373635343332$$

$$3130393837363534333231303938373635343332313039383736353433323130$$

Вычисляется значение $K := LPS(h \oplus N) = LPS((00000001)^{64})$.

После преобразования S :

$$S(h \oplus N) = \text{ee}$$

$$\text{ee}$$

после преобразования P :

$$PS(h \oplus N) = \text{ee}$$

$$\text{ee}$$

после преобразования L :

$$K = LPS(h \oplus N) = 23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15$$

$$23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15.$$

Затем выполняется преобразование $E(K, m)$:

Итерация 1

$$K_1 = 23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15$$

$$23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15$$

$$X[K_1](m) = 22f7df708943682316f1dd72814b662d14f3db7483496e251afdd976854f6c27$$

$$12f5d778874d6a2110f7df708943682316f1dd72814b662d14f3db7483496e25$$

$$SX[K_1](m) = 65c061327951f35a99a6d819f5a29a0193d290ffa92ab25cf14b538aa8cc9d21$$

$$f0f4fe6dc93a7818e9c061327951f35a99a6d819f5a29a0193d290ffa92ab25c$$

$$PSX[K_1](m) = 5a015c21185a015cf39ab29d78f39ab251a22acc3a51a22a79f5a9a8c979f5a9$$

$$3219ff8a6d3219ff61d89053fe61d890c0a6d24bf4c0a6d2659993f1f0e99993$$

$$LPSX[K_1](m) = e549368917a0a2611d5e08c9c2fd5b3c563f18c0f68c410d84ae9d5fbdfb9340$$

$$55650121b7aa6d7b3e7d09d46ac4358adaa6ae44fa3b0402c4166d2c3eb2ef02$$

$$K_1 \oplus C_1 = 92cdb59aaeb185fcc80ec1c1701e230a0caf98039e3e8f03528b56cdc5fe9be9$$

$$68b90ed1221c36148187c448141b8c0026b39a767c0f1236fe458b1942dd1a12$$

$S(K_1 \oplus C_1) = \text{ecd95e282645a83930045858325f5afa2341dc110ad303110ef676d9ac63509b}$
 $\text{f3a341b65148f93f5c986f293bb7cfcef92288ac34df08f63c8f6362cd8f1f0}$

$PS(K_1 \oplus C_1) = \text{39fa119b93fc8ff0a85a03508f7cf0f1455fd36314bb4dd826320aac6593c32c}$
 $\text{285811d91bf28a365e58dc76048628f6d90441f6a3c992c8ec30230ef3f5ef63}$

$LPS(K_1 \oplus C_1) = \text{18ee8f3176b2ebea3bd6cb8233694cea349769df88be26bf451cfab6a904a549}$
 $\text{da22de93a66a66b19c7e6b5eea633511e611d68c8401bfcd0c7d0cc39d4a5eb9}$

Итерация 2

$K_2 = \text{18ee8f3176b2ebea3bd6cb8233694cea349769df88be26bf451cfab6a904a549}$
 $\text{da22de93a66a66b19c7e6b5eea633511e611d68c8401bfcd0c7d0cc39d4a5eb9}$

$LPSX[K_2]LPSX[K_1](m) =$
 $\text{c502dab7e79eb94013fcd1ba64def3b916f18b63855d43d22b77fca1452f9866}$
 $\text{c2b45089c62e9d82edf1ef45230db9a23c9e1c521113376628a5f6a5dbc041b2}$

Итерация 3

$K_3 = \text{aaa4cf31a265959157aec8ce91e7fd46bf27dee21164c5e3940bba1a519e9d1f}$
 $\text{ce0913f1253e7757915000cd674be12cc7f68e73ba26fb00fd74af4101805f2d}$

$LPSX[K_3] \dots LPSX[K_1](m) =$
 $\text{8e5a4fe41fc790af29944f027aa2f10105d65cf60a66e442832bb9ab5020dc54}$
 $\text{772e36b03d4b9aa471037212cde93375226552392ef4d83010a007e1117a07b5}$

Итерация 4

$K_4 = \text{61fe0a65cc177af50235e2afadded326a5329a2236747bf8a54228aeca9c4585}$
 $\text{cd801ea9dd743a0d98d01ef0602b0e332067fb5ddd6ac1568200311920839286}$

$LPSX[K_4] \dots LPSX[K_1](m) =$
 $\text{dee0b40df69997afef726f03bdc13cb6ba9287698201296f2fd8284f06d33ea4}$
 $\text{a850a0ff48026dd47c1e88ec813ed2eb1186059d842d8d17f0bfa259e56655b1}$

Итерация 5

$K_5 = \text{9983685f4fd3636f1fd5abb75fbf26a8e2934314aa2ecb3ee4693c86c06c7d4e}$
 $\text{169bd540af75e1610a546acd63d960bad595394cc199bf6999a5d5309fe73d5a}$

$LPSX[K_5] \dots LPSX[K_1](m) =$
675ea894d326432e1af7b201bc369f8ab021f6fa58da09678ffc08ef30db43a3
7f1f7347cb77da0f6ba30c85848896c3bac240ab14144283518b89a33d0caf07

Итерация 6

$K_6 =$ f05772ae2ce7f025156c9a7fbcc6b8fdf1e735d613946e32922994e52820ffea
62615d907eb0551ad170990a86602088af98c83c22cdb0e2be297c13c0f7a156

$LPSX[K_6] \dots LPSX[K_1](m) =$
1bc204bf9506ee9b86bbcf82d254a112aea6910b6db3805e399cb718d1b33199
64459516967cee4e648e8cfbf81f56dc8da6811c469091be5123e6a1d5e28c73

Итерация 7

$K_7 =$ 5ad144c362546e4e46b3e7688829fbb77453e9c3211974330b2b8d0e6be2b5ac
c89eb6b35167f159b7b005a43e5959a651a9b18cfc8e4098fcf03d9b81cfbb8d

$LPSX[K_7] \dots LPSX[K_1](m) =$
f30d791ed78bdee819022a3d78182242124efcdd54e203f23fb2dc7f94338ff9
55a5afc15ffef03165263c4fdb36933aa982016471fbac9419f892551e9e568b

Итерация 8

$K_8 =$ 6a6cec9a1ba20a8db64fa840b934352b518c638ed530122a83332fe0b8efdac9
018287e5a9f509c78d6c746adcd5426fb0a0ad5790dfb73fc1f191a539016daa

$LPSX[K_8] \dots LPSX[K_1](m) =$
1fc20f1e91a1801a4293d3f3aa9e91560fcc3810bb15f3ee9741c9b87452519f
67cb9145519884a24de6db736a5cb1430da7458e5e51b80be5204ba5b2600177

Итерация 9

$K_9 =$ 99217036737aa9b38a8d6643f705bd51f351531f948f0fc5e35fa35fee9dd8bd
bb4c9d580a224e9cd82e0e2069fc49ed367d5f94374435382b8fb6a8f5dd0409

$LPSX[K_9] \dots LPSX[K_1](m) =$
1a52f09d1e81515a36171e0b1a2809c50359bed90f2e78cbd89b7d4afa6d0466
55c96bdae6ee97055cc7e857267c2ccf28c8f5dd95ed58a9a68c12663bb28967

Итерация 10

$K_{10} = 906763c0fc89fa1ae69288d8ec9e9dda9a7630e8bfd6c3fed703c35d2e62aeaf$
 $f0b35d80a7317a7f76f83022f2526791ca8fdf678fcb337bd74fe5393ccb05d2$

$LPSX[K_{10}] \dots LPSX[K_1](m) =$

$764043744a0a93687e65aba8cfc25ec8714fb8e1bdc9ae2271e7205eaaa577c1$
 $b3b83e7325e50a19bd2d56b061b5de39235c9c9fd95e071a1a291a5f24e8c774$

Итерация 11

$K_{11} = 88ce996c63618e6404a5c8e03ee433854e2ae3eee68991bbbff3c29d38dadb6e$
 $d6a1dae9a6dc6ddf52ce34af272f96d3159c8c624c3fe6e13d695c0bfc89add5$

$LPSX[K_{11}] \dots LPSX[K_1](m) =$

$9b1ce8ff26b445cb288c0aeccf84658eea91dbdf14828bf70110a5c9bd146cd9$
 $646350cff4e90e7b63c5cc325e9b441081935f282d4648d9584f71860538f03b$

Итерация 12

$K_{12} = 3e0a281ea9bd46063eec550100576f3a506aa168cf82915776b978fcca32f38$
 $b55f30c79982ca45628e8365d8798477e75a49c68199112a1d7b5a0f7655f2db$

$LPSX[K_{12}] \dots LPSX[K_1](m) =$

$133aeeccede251eb81914b8ba48dcbc0b8a6fc63a292cc49043c3d3346b3f0829$
 $a9cb71ecff25ed2a91bdcf8f649907c110cb76ff2e43100cdd4ba8a147a572f5$

Итерация 13

$K_{13} = f0b273409eb31aebe432fbae1867212262c848422b6a92f93f6cbab54ed18b83$
 $14b21cffc51e3fa319ff433e76ef6adb0ef9f5e03c907fa1fcf9eca06500bf03$

$LPSX[K_{13}] \dots LPSX[K_1](m) =$

$e3889d8e40960453fd26431450bb9d29e8a78e78024656697caf698125ee83aa$
 $14b21cffc51e3fa319ff433e76ef6adb0ef9f5e03c907fa1fcf9eca06500bf03$

Результат выполнения преобразования $g_N(h, m)$:

$h = e3bbadb7f8af3264c9137127608aa510de90ba4d3075665844965fb611dbb199$
 $8d48552a0c0ce6bcba71bc802a4f5b2d2a07b12c22e25794178570341096fdc7$

$K_2 = d0b00807642fd78f13f2c3ebc774e80de0e902d23aef2ee9a73d010807dae9c188be14f0b2da27973569cd2ba051301036f728bd1d7eec33f4d18af70c46cf1e$

$LPSX[K_2] \dots LPSX[K_1](m) =$
 $301aadd761d13df0b473055b14a2f74a45f408022aecadd4d5f19cab8228883a021ac0b62600a495950c628354ffce1161c68b7be7e0c58af090ce6b45e49f16$

Итерация 3

$K_3 = 9d4475c7899f2d0bb0e8b7dac6ef6e6b44ecf66716d3a0f16681105e2d13712a1a9387ecc257930e2d61014a1b5c9fc9e24e7d636eb1607e816dbaf927b8fca9$

$LPSX[K_3] \dots LPSX[K_1](m) =$
 $9b83492b9860a93cbca1c0d8e0ce59db04e10500a6ac85d4103304974e78d32259ceff03fbb353147a9c948786582df78a34c9bde3f72b3ca41b9179c2cceeef3$

Итерация 4

$K_4 = 5c283daba5ec1f233b8c833c48e1c670dae2e40cc4c3219c73e58856bd96a72fd9f8055ffe3c004c8cde3b8bf78f95f3370d0a3d6194ac5782487defd83ca0f$

$LPSX[K_4] \dots LPSX[K_1](m) =$
 $e638e0a1677cdea107ec3402f70698a4038450dab44ac7a447e10155aa33ef1bdaf8f49da7b66f3e05815045fbd39c991cb0dc536e09505fd62d3c2cd00b0f57$

Итерация 5

$K_5 = 109f33262731f9bd569cbc9317baa551d4d2964fa18d42c41fab4e37225292ec2fd97d7493784779046388469ae195c436fa7cba93f8239ceb5ffc818826470c$

$LPSX[K_5] \dots LPSX[K_1](m) =$
 $1c7c8e19b2bf443eb3adc0c787a52a173821a97bc5a8efea58fb8b27861829f6dd5ff9c97865e08c1ac66f47392b578e21266e323a0aacedeec3ef0314f517c6$

Итерация 6

$K_6 = b32c9b02667911cf8f8a0877be9a170757e25026ccf41e67c6b5da70b1b874743e1135cfbefe244237555c676c153d99459bc382573aee2d85d30d99f286c5e7$

$LPSX[K_6] \dots LPSX[K_1](m) =$
 $48fecfc5b3eb77998fb39bfcccd128cd42fccb714221be1e675a1c6fdde7e31198b318622412af7e999a3eff45e6d61609a7f2ae5c2ff1ab7ff3b37be7011ba2$

Итерация 7

$K_7 = 8a13c1b195fd0886ac49989e7d84b08bc7b00e4f3f62765ece6050fcbabdc234$
 $6c8207594714e8e9c9c7aad694edc922d6b01e17285eb7e61502e634559e32f1$

$LPSX[K_7] \dots LPSX[K_1](m) =$

$a48f8d781c2c5be417ae644cc2e15a9f01fceed3232e5bd53f18a5ab875cce1b$
 $8a1a400cf48521c7ce27fb1e94452fb54de23118f53b364ee633170a62f5a8a9$

Итерация 8

$K_8 = 52cec3b11448bb8617d0ddfb9c926f2e88730cb9179d6decea5acbffd323ec376$
 $4c47f7a9e13bb1db56c342034773023d617ff01cc546728e71dff8de5d128cac$

$LPSX[K_8] \dots LPSX[K_1](m) =$

$e8a31b2e34bd2ae21b0ecf29cc4c37c75c4d11d9b82852517515c23e81e906a4$
 $51b72779c3087141f1a15ab57f96d7da6c7ee38ed25befbdef631216356ff59c$

Итерация 9

$K_9 = f38c5b7947e7736d502007a05ea64a4eb9c243cb82154aa138b963bbb7f28e74$
 $d4d710445389671291d70103f48fd4d4c01fc415e3fb7dc61c6088afa1a1e735$

$LPSX[K_9] \dots LPSX[K_1](m) =$

$34392ed32ea3756e32979cb0a2247c3918e0b38d6455ca88183356bf8e5877e5$
 $5d542278a696523a8036af0f1c2902e9cbc585de803ee4d26649c9e1f00bda31$

Итерация 10

$K_{10} = 0740b3faa03ed39b257dd6e3db7c1bf56b6e18e40cdaabd30617cecbadddd618e$
 $a5e61bb4654599581dd30c24c1ab877ad0687948286cfefaa7eef99f6068b315$

$LPSX[K_{10}] \dots LPSX[K_1](m) =$

$6a82436950177fea74cce6d507a5a64e54e8a3181458e3bdfbdbc6180c9787de$
 $7ccb676dd809e7cb1eb2c9ebd016561570801a4e9ce17a438b85212f4409bb5e$

Итерация 11

$K_{11} = 185811cf3c2633aec8cfdca9dbb29347011bf92b95910a3ad71e5fca678e45$
 $e374f088f2e5c29496e9695ce8957837107bb3aa56441af11a82164893313116$

$LPSX[K_{11}] \dots LPSX[K_1](m) =$
7b97603135e2842189b0c9667596e96bd70472ccbc73ae89da7d1599c72860c2
85f5771088f1fb0f943d949f22f1413c991eafb51ab8e5ad8644770037765aec

Итерация 12

$K_{12} =$ 9d46bf66234a7ed06c3b2120d2a3f15e0fedd87189b75b3cd2f206906b5ee00d
c9a1eab800fb8cc5760b251f4db5cdef427052fa345613fd076451901279ee4c

$LPSX[K_{12}] \dots LPSX[K_1](m) =$
39ec8a88db635b46c4321adf41fd9527a39a67f6d7510db5044f05efaf721db5
cf976a726ef33dc4dfcda94033e741a463770861a5b25fefcb07281eed629c0e

Итерация 13

$K_{13} =$ 0f79104026b900d8d768b6e223484c9761e3c585b3a405a6d2d8565ada926c3f
7782ef127cd6b98290bf612558b4b60aa3cbc28fd94f95460d76b621cb45be70

$LPSX[K_{13}] \dots LPSX[K_1](m) =$
36959ac8fdda5b9e135aac3d62b5d9b0c279a27364f50813d69753b575e0718a
b8158560122584464f72c8656b53f7aec0bccae7cfdcaa9c6719e3f2627227e

Результат выполнения преобразования $g_N(h, m)$:

$h =$ cd7f602312faa465e3bb4ccd9795395de2914e938f10f8e127b7ac459b0c517b
98ef779ef7c7a46aa7843b8889731f482e5d221e8e2cea852e816cdac407c7af

Изменяются значения переменных N и Σ :

$N =$ 00
00200

$\Sigma =$ fbeafaebef20ffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ce8f0f2e5e220e5d1,

Длина оставшейся части сообщения меньше 512, поэтому происходит заполнение неполного блока.

$m :=$ 00
001fbe2e5f0eee3c820

Результат выполнения преобразования $g_N(h, m)$:

$$LPSX[K_2]LPSX[K_1](m) =$$

9f50697b1d9ce23680db1f4d35629778864c55780727aa79eb7bb7d648829cba
8674afdac5c62ca352d77556145ca7bc758679fbe1fbd32313ca8268a4a603f1

Итерация 3

$K_3 =$ aaa4cf31a265959157aec8ce91e7fd46bf27dee21164c5e3940bba1a519e9d1f
ce0913f1253e7757915000cd674be12cc7f68e73ba26fb00fd74af4101805f2d

$$LPSX[K_3] \dots LPSX[K_1](m) =$$

4183027975b257e9bc239b75c977ecc52ddad82c091e694243c9143a945b4d85
3116eae14fd81b14bb47f2c06fd283cb6c5e61924edfaf971b78d771858d5310

Итерация 4

$K_4 =$ 61fe0a65cc177af50235e2afadded326a5329a2236747bf8a54228aeca9c4585
cd801ea9dd743a0d98d01ef0602b0e332067fb5ddd6ac1568200311920839286

$$LPSX[K_4] \dots LPSX[K_1](m) =$$

368c884fcee489207b5b97a133ce39a1ebfe5a3ae3cccb3241de1e7ad72857e
76811d324f01fd7a75e0b669e8a22a4d056ce6af3e876453a9c3c47c767e5712

Итерация 5

$K_5 =$ 9983685f4fd3636f1fd5abb75fbf26a8e2934314aa2ecb3ee4693c86c06c7d4e
169bd540af75e1610a546acd63d960bad595394cc199bf6999a5d5309fe73d5a

$$LPSX[K_5] \dots LPSX[K_1](m) =$$

c31433ceb8061e46440144e65553976512e5a9806ac9a2c771d5932d5f6508c5
b78e406c4efab98ac5529be0021b4d58fa26f01621eb10b43de4c4c47b63f615

Итерация 6

$K_6 =$ f05772ae2ce7f025156c9a7fbcc6b8fdf1e735d613946e32922994e52820ffea
62615d907eb0551ad170990a86602088af98c83c22cdb0e2be297c13c0f7a156

$$LPSX[K_6] \dots LPSX[K_1](m) =$$

5d0ae97f252ad04534503fe5f52e9bd07f483ee3b3d206beadc6e736c6e754bb
713f97ea7339927893eacf2b474a482cadd9ac2e58f09bcb440cf36c2d14a9b6

Итерация 7

$K_7 = 5ad144c362546e4e46b3e7688829fbb77453e9c3211974330b2b8d0e6be2b5ac$
 $c89eb6b35167f159b7b005a43e5959a651a9b18cfc8e4098fcf03d9b81cfbb8d$

$LPSX[K_7] \dots LPSX[K_1](m) =$
 $a59aa21e6ad3e330deedb9ab9912205c355b1c479fdfd89a7696d7de66fbf7d3$
 $cec25879f7f1a8cca4c793d5f2888407aecb188bda375eae586a8cfd0245c317$

Итерация 8

$K_8 = 6a6cec9a1ba20a8db64fa840b934352b518c638ed530122a83332fe0b8efdac9$
 $018287e5a9f509c78d6c746adcd5426fb0a0ad5790dfb73fc1f191a539016daa$

$LPSX[K_8] \dots LPSX[K_1](m) =$
 $9903145a39d5a8c83d28f70fa1fbd88f31b82dc7cfe17b54b50e276cb2c4ac68$
 $2b4434163f214cf7ce6164a75731bcea5819e6a6a6fea99da9222951d2a28e01$

Итерация 9

$K_9 = 99217036737aa9b38a8d6643f705bd51f351531f948f0fc5e35fa35fee9dd8bd$
 $bb4c9d580a224e9cd82e0e2069fc49ed367d5f94374435382b8fb6a8f5dd0409$

$LPSX[K_9] \dots LPSX[K_1](m) =$
 $330e6cb1d04961826aa263f2328f15b4f3370175a6a9fd6505b286efed2d8505$
 $f71823337ef71513e57a700eb1672a685578e45dad298ee2223d4cb3fda8262f$

Итерация 10

$K_{10} = 906763c0fc89fa1ae69288d8ec9e9dda9a7630e8bfd6c3fed703c35d2e62aeaf$
 $f0b35d80a7317a7f76f83022f2526791ca8fdf678fcb337bd74fe5393ccb05d2$

$LPSX[K_{10}] \dots LPSX[K_1](m) =$
 $ad347608443ab9c9bbb64f633a5749ab85c45d4174bfd78f6bc79fc4f4ce9ad1$
 $dd71cb2195b1cfab8dcaaf6f3a65c8bb0079847a0800e4427d3a0a815f40a644$

Итерация 11

$K_{11} = 88ce996c63618e644a5c8e03ee433854e2ae3eee68991bbbff3c29d38dad6e$
 $d6a1dae9a6dc6ddf52ce34af272f96d3159c8c624c3fe6e13d695c0bfc89add5$

$$LPSX[K_{11}] \dots LPSX[K_1](m) =$$

a065c55e2168c31576a756c7ecc1a9129cd3d207f8f43073076c30e111fd5f11
9095ca396e9fb78a2bf4781c44e845e447b8fc75b788284aae27582212ec23ee

Итерация 12

$K_{12} = 3e0a281ea9bd46063ecc550100576f3a506aa168cf82915776b978fcca32f38$
 $b55f30c79982ca45628e8365d8798477e75a49c68199112a1d7b5a0f7655f2db$

$$LPSX[K_{12}] \dots LPSX[K_1](m) =$$

2a6549f7a5cd2eb4a271a7c71762c8683e7a3a906985d60f8fc86f64e35908b2
9f83b1fe3c704f3c116bdfe660704f3b9c8a1d0531baaffaa3940ae9090a33ab

Итерация 13

$K_{13} = f0b273409eb31aebe432fbae1867212262c848422b6a92f93f6cbab54ed18b83$
 $14b21cffc51e3fa319ff433e76ef6adb0ef9f5e03c907fa1fcf9eca06500bf03$

$$LPSX[K_{13}] \dots LPSX[K_1](m) =$$

dad73ab73b7e345f46435c690f05e94a5cb272d242ef44f6b0a4d5d1ad888331
8b31ad01f96e709f08949cd8169f25e09273e8e50d2ad05b5f6de6496c0a8ca8

Результат выполнения преобразования $g_N(h, m)$:

$h = 203cc15dd55fcaa5b7a3bd98fb2408a67d5b9f33a80bb50540852b204265a2c1$
 $aa5a5efe1d8d51b2e1636e34f5becc077d930114fefaf176b69c15ad8f2b6878$

Изменяются значения переменных N и Σ :

$N = 00$
 00200

$\Sigma = fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0e0e020f1$
 $20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ce8f0f2e5e220e5d1,$

Длина оставшейся части сообщения меньше 512, поэтому происходит заполнение неполного блока:

$m = 00$
 $0001f$
 $be2e5f0eee3c820$

Результат выполнения преобразования $g_N(h, m)$:

$h = \text{a69049e7bd076ab775bc2873af26f098c538b17e39a5c027d532f0a2b3b56426}$
 $\text{c96b285fa297b9d39ae6afd8b9001d97bb718a65fcc53c41b4ebf4991a617227}$

Изменяются значения переменных N и Σ :

$N = \text{00}$
 00240

$\Sigma = \text{fbeafaebef20ffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1}$
 $\text{20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ee4d3d8d6d104adf1}$

Результат выполнения преобразования $g_0(h, N)$:

$h = \text{aee3bd55ea6f387bcf28c6dcdbbbfb3ddacc67dcc13dbd8d548c6bf808111d4b}$
 $\text{75b8e74d2afae960835ae6a5f03575559c9fd839783ffcd5cf99bd61566b4818}$

Результат выполнения преобразования $g_0(h, \Sigma)$:

$h = \text{508f7e553c06501d749a66fc28c6cac0b005746d97537fa85d9e40904efed29d}$
 $\text{c345e53d7f84875d5068e4eb743f0793d673f09741f9578471fb2598cb35c230}$.

Хэш-кодом сообщения M_2 является значение

$H(M_2) = \text{508f7e553c06501d749a66fc28c6cac0b005746d97537fa85d9e40904efed29d}$.

Библиография¹

- [1] ИСО 2382-2:1976 Системы обработки информации. Словарь. Часть 2. Арифметические и логические операции
- [2] ИСО/МЭК 9796-2:2010 Информационные технологии. Методы обеспечения безопасности. Схемы цифровой подписи, обеспечивающие восстановление сообщений. Часть 2. Механизмы на основе целочисленной факторизации
- [3] ИСО/МЭК 9796-3:2006 Информационные технологии. Методы обеспечения безопасности. Схемы цифровой подписи, обеспечивающие восстановление сообщений. Часть 3. Механизмы на основе дискретного логарифма
- [4] ИСО/МЭК 14888-1:2008 Информационные технологии. Методы защиты. Цифровые подписи с приложением. Часть 1. Общие положения
- [5] ИСО/МЭК 14888-2:2008 Информационные технологии. Методы защиты. Цифровые подписи с приложением. Часть 2. Механизмы, основанные на разложении на множители
- [6] ИСО/МЭК 14888-3:2006 Информационные технологии. Методы защиты. Цифровые подписи с приложением. Часть 3. Механизмы на основе дискретного логарифма
- [7] ИСО/МЭК 14888-3:2006/Amd 1:2010 Методы защиты. Цифровые подписи с приложением. Часть 3. Механизмы на основе дискретного логарифма. Изменение 1. Алгоритм русской цифровой подписи эллиптической кривой, алгоритм цифровой подписи Шнора, алгоритм цифровой подписи Шнора для эллиптической кривой, и полный алгоритм цифровой подписи Шнора для эллиптической кривой
- [8] ИСО/МЭК 10118-1:2000 Информационные технологии. Методы защиты информации. Хэш-функции. Часть 1. Общие положения
- [9] ИСО/МЭК 10118-2:2010 Информационные технологии. Методы защиты информации. Хэш-функции. Часть 2. Хэш-функции с использованием алгоритма шифрования n-битными блоками
- [10] ИСО/МЭК 10118-3:2004 Информационные технологии. Методы защиты информации. Хэш-функции. Часть 3. Выделенные хэш-функции
- [11] ИСО/МЭК 10118-4:1998 Информационные технологии. Методы защиты информации. Хэш-функции. Часть 4. Хэш-функции с применением арифметики в остаточных классах

¹ Оригиналы международных стандартов ИСО/МЭК – в ФГУП «Стандартинформ» Федерального агентства по техническому регулированию и метрологии

Ключевые слова: информационная технология, криптографическая защита информации, функция хэширования, хэш-функция, электронная цифровая подпись, асимметричный криптографический алгоритм, системы обработки информации, защита сообщений, подтверждение подписи
