FEDERAL AGENCY
ON TECHNICAL REGULATION AND METROLOGY

NATIONAL
STANDARD OF THE
RUSSIAN
FEDERATION

GOST R 34.11-2012

Information technology

CRYPTOGRAPHIC DATA SECURITY

Hash-function

Official Publication
English Version

Moscow
Standartinform
2012

Foreword

Standardization purposes and principles in the Russian Federation are established by Federal Law On Technical Regulation No. 184-FZ dated December 27, 2002, and the application rules for the national standards of the Russian Federation are regulated by GOST R 1.0–2004 “Standardization in the Russian Federation. Basic provisions”

Information on the Standard

1 DEVELOPED by the Center for Information Protection and Special Communications of the Federal Security Service of the Russian Federation with participation of the Open Joint-Stock Company “Information Technologies and Communication Systems” (JSC “InfoTeCS”)

2 SUBMITTED by the Technical Committee for Standardization TC 26 “Cryptography and security mechanisms”

3 APPROVED AND PUT INTO EFFECT under Decree No. 216-st of Federal Agency on Technical Regulation and Metrology dated August 7, 2012

4 REPLACES GOST R 34.11-94

Information on amendments to this Standard is published in National Standards annual information index. The text of revisions and amendments is published in National Standards monthly information indices. In case of revision (replacement) or cancellation of this Standard, a corresponding notification is published in National Standards monthly information index. Any corresponding information, notifications and texts are also available in the public informational system — on the official web-site of the Federal Agency on Technical Regulation and Metrology

Contents

Introduction

This Standard specifies the hash-function algorithm and the hash-function calculation procedure for any sequence of binary symbols. The algorithm and the calculation procedure are used in cryptographic methods of information protection, including the processes of producing and verifying digital signatures.

This Standard replaces GOST R 34.11–94. The need for the development of this Standard was determined by the demand for a hash-function to meet modern requirements for cryptographic strength and the requirements of GOST R 34.10–2012 for digital signatures.

The terms and notions of this Standard comply with international standards ISO 2382-2 [1], ISO/IEC 9796 [2 – 3], series of standards ISO/IEC 14888 [4 – 7], and series of standards ISO/IEC 10118 [8 – 11].

N o t e – The main part of this Standard is supplemented with one Annex:
Annex A (Informative) Examples.

NATIONAL STANDARD OF THE RUSSIAN FEDERATION

Information technology

CRYPTOGRAPHIC DATA SECURITY

Hash-function

Effective date – January 1, 2013

1 Scope

This Standard specifies the hash-function algorithm and calculation procedure for any sequence of binary symbols used in cryptographic methods of information processing and protection, including the provision of integrity, authentication and non-repudiation services, during information transmission, processing and storage in computer-aided systems.

The hash-function specified in this Standard is used to implement digital signature solutions based on the asymmetric cryptographic algorithm in compliance with GOST R 34.10–2012.

This Standard applies to creation, operation, and modernization of information systems of different purpose.

2 Standard References

The following standards are referred to in this Standard:

GOST R 34.10–2012 Information technology. Cryptographic data security. Formation and verification processes of [electronic] digital signature

Official Publication

N o t e – The user of this Standard may need to check the validity of the referenced standards on the official web-site of the Federal Agency on Technical Regulation and Metrology, in the National Standards annual informational index published as on January the 1st of the current year, and in corresponding monthly indices published during the current year. If any referenced standard is replaced (amended), then the replacing (amending) standard shall be used. If any referenced standard is cancelled without replacement, then the statement containing the reference can be used only in the part not concerning the cancelled standard.

3 Terms, Definitions, and Symbols

In this Standard, the following terms with corresponding definitions are used.

3.1 Terms and definitions

3.1.1

padding
appending extra bits to a data string

[ISO/IEC 10118-1, clause 3.9]

3.1.2

initializing value
a value used in defining the starting point of a hash-function

[ISO/IEC 10118-1, clause 3.7]

3.1.3

message
a string of bits of any length

[ISO/IEC 14888-1 clause 3.10]

3.1.4

round-function
a function that transforms two binary strings of lengths L_1 and L_2 to a binary string of length L_2, it is used iteratively as part of a hash-function, where it combines a data string of length L_1 with the previous output of length L_2

[ISO/IEC 10118-1, clause 3.10]

N O T E – In this Standard, the notions “string of bits of length L” and “binary row vector of length L” are identical.

3.1.5

hash-code
the string of bits which is the output of a hash-function

[ISO/IEC 14888-1, clause 3.6]

3.1.6

collision-resistant hash-function

a function which maps strings of bits to fixed-length strings of bits satisfying the following properties:

1) for a given output, it is computationally infeasible to find an input which maps to this output;

2) for a given input, it is computationally infeasible to find a second input which maps to the same output;

3) it is computationally infeasible to find any two distinct inputs which map to the same output

[ISO/IEC 14888-1, clause 3.2, 3.7]

N O T E – In this Standard (to provide terminological compatibility with the current national standard documentation and with the published scientific and technical works), it is established that the terms “hash-function” and “cryptographic hash-function” are synonyms.

3.1.7

[digital] signature

one or more data elements resulting from the signature process

[ISO/IEC 14888-1, clause 3.12]

N O T E – In this Standard (to provide terminological compatibility with the current national standard documentation and with the published scientific and technical works), it is established that the terms “digital signature”, “electronic signature”, and “electronic digital signature” are synonyms.

3.2 Symbols

The following symbols are used in this Standard:

V^* the set of all binary row vectors of a finite length (hereinafter referred to as “vectors”), including an empty string;

$|A|$ the length (number of components) of the vector $A \in V^*$ (if A is the empty string, then $|A| = 0$);

V_n the set of all binary vectors of length n, where n is a non-negative integer; subvectors and vector components are enumerated from right to left starting from zero;

$\oplus$ the exclusive OR operation on two binary vectors of the same length;

$A\|B$ concatenation of the vectors $A,B \in V^*$, i.e. a vector from $V_{|A|+|B|}$, where the left subvector from $V_{|A|}$ is equal to the vector A and the right subvector from $V_{|B|}$ is equal to the vector B;

A^n concatenation of n instances of the vector A;

$\mathbb{Z}_{2^n}$	the integer residue ring modulo 2^n;
$\boxplus$	the addition operation in the ring $\mathbb{Z}_{2^n}$;
$\mathrm{Vec}_n : \mathbb{Z}_{2^n} \to V_n$	a bijective mapping, which for an integer from $\mathbb{Z}_{2^n}$ puts into correspondence its binary representation, i.e. for any integer $z = z_0 + 2z_1 + \ldots + 2^{n-1}z_{n-1}$ of the ring $\mathbb{Z}_{2^n}$, where $z_j \in \{0,1\}$, $j = 0,\ldots,n$-1, the equality $\mathrm{Vec}_n(z) = z_{n-1}\|\ldots\|z_1\|z_0$ holds;
$\mathrm{Int}_n : V_n \to \mathbb{Z}_{2^n}$	the mapping inverse to the mapping Vec_n, i.e. $\mathrm{Int}_n = \mathrm{Vec}_n^{-1}$;
$\mathrm{MSB}_n : V^* \to V_n$	the mapping associating the vector $z_{k-1}\|\ldots\|z_1\|z_0$, $k \geq n$ with the vector $z_{k-1}\|\ldots\|z_{k-n+1}\|z_{k-n}$;
$a := b$	the operation of assigning the value b to the variable a;
$\Phi\Psi$	a composition of mappings, where the mapping Ψ applies first;
M	a binary vector to be hashed, $M \in V^*$, $\|M\| < 2^{512}$;
H: $V^* \to V_n$	a hash-function mapping a vector (message) M into the vector (hash-code) $H(M)$;
IV	a hash-function initializing value, $IV \in V_{512}$.

4 General provisions

This Standard specifies two hash-functions H: $V^* \to V_n$ with the hash-code lengths n = 512 bits and n = 256 bits.

5 Parameter values

5.1 Initializing values

The initializing value IV for the hash-function with the hash-code length of 512 bits is 0^{512}. The initializing value IV for the hash-function with the hash-code length of 256 bits is $(00000001)^{64}$.

5.2 Nonlinear bijections of binary-vector sets

A nonlinear bijection of the binary-vector set V_8 is defined by the following substitution

$$\pi = \mathrm{Vec}_8\, \pi' \mathrm{Int}_8 \colon V_8 \to V_8, \tag{1}$$

where π': $\mathbb{Z}_{2^8} \to \mathbb{Z}_{2^8}$.

The values of the substitution π' are given by the array π' = (π'(0), π'(1),..., π' (255)):

π' = (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 Byte permutation

The values of the permutation τ, defined over the set {0,…,63}, are given by the array τ = (τ(0), τ(1),..., τ(63)):

τ = (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 Linear transformations of binary-vector sets

A linear transformation l of the binary-vector set V_{64} is specified by the right multiplication by matrix A over the field $GF(2)$. The matrix rows are expressed sequentially in hexadecimal notation. The row with number j, j = 0,...,63 (specified in the form $a_{j,15} \ldots a_{j,0}$, where $a_{j,i} \in \mathbb{Z}_{16}$, i = 0,...,15), is $\mathrm{Vec}_4(a_{j,15}) \| \ldots \| \mathrm{Vec}_4(a_{j,0})$.

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

Here one string contains four rows of the matrix A. So, the string with number i, $i = 0,...,15$, specifies rows of the matrix A with the numbers $4\,i + j$, $j = 0,...,3$, in the following left-to-right order: $4\,i + 0$, $4\,i + 1$, $4\,i + 2$, $4\,i + 3$.

The product of the vector $b = b_{63} \ldots b_0 \in V_{64}$ and the matrix A is a vector $c \in V_{64}$:

$$c = b_{63}(\mathrm{Vec}_4(a_{0,15})\| \ldots \|\mathrm{Vec}_4(a_{0,0})) \oplus \ldots \oplus b_0(\mathrm{Vec}_4(a_{63,15})\| \ldots \|\mathrm{Vec}_4(a_{63,0})), \quad (2)$$

where $b_i\ (\mathrm{Vec}_4(a_{63-i,15})\| \ldots \|\mathrm{Vec}_4(a_{63-i,0})) = \begin{cases} 0^{64}, & \text{if } b_i = 0, \\ (\mathrm{Vec}_4(a_{63-i,15})\| \ldots \|\mathrm{Vec}_4(a_{63-i,0})) & \text{if } b_i = 1, \end{cases}$

for all $i = 0,...,63$.

5.5 Iteration constants

Iteration constants are expressed in hexadecimal notation. The constant value specified in the form $a_{127} \ldots a_0$ (where $a_i \in \mathbb{Z}_{16}$, $i = 0,...,127$) is $\mathrm{Vec}_4(a_{127})\| \ldots \|\mathrm{Vec}_4(a_0)$:

C_1 = b1085bda1ecadae9ebcb2f81c0657c1f2f6a76432e45d016714eb88d7585c4fc
4b7ce09192676901a2422a08a460d31505767436cc744d23dd806559f2a64507;

C_2 = 6fa3b58aa99d2f1a4fe39d460f70b5d7f3feea720a232b9861d55e0f16b50131
9ab5176b12d699585cb561c2db0aa7ca55dda21bd7cbcd56e679047021b19bb7;

C_3 = f574dcac2bce2fc70a39fc286a3d843506f15e5f529c1f8bf2ea7514b1297b7b
d3e20fe490359eb1c1c93a376062db09c2b6f443867adb31991e96f50aba0ab2;

C_4 = ef1fdfb3e81566d2f948e1a05d71e4dd488e857e335c3c7d9d721cad685e353f
a9d72c82ed03d675d8b71333935203be3453eaa193e837f1220cbebc84e3d12e;

C_5 = 4bea6bacad4747999a3f410c6ca923637f151c1f1686104a359e35d7800fffbd
bfcd1747253af5a3dfff00b723271a167a56a27ea9ea63f5601758fd7c6cfe57;

C_6 = ae4faeae1d3ad3d96fa4c33b7a3039c02d66c4f95142a46c187f9ab49af08ec6
cffaa6b71c9ab7b40af21f66c2bec6b6bf71c57236904f35fa68407a46647d6e;

C_7 = f4c70e16eeaac5ec51ac86febf240954399ec6c7e6bf87c9d3473e33197a93c9
0992abc52d822c3706476983284a05043517454ca23c4af38886564d3a14d493;

C_8 = 9b1f5b424d93c9a703e7aa020c6e41414eb7f8719c36de1e89b4443b4ddbc49a
f4892bcb929b069069d18d2bd1a5c42f36acc2355951a8d9a47f0dd4bf02e71e;

C_9 = 378f5a541631229b944c9ad8ec165fde3a7d3a1b258942243cd955b7e00d0984
800a440bdbb2ceb17b2b8a9aa6079c540e38dc92cb1f2a607261445183235adb;

C_{10} = abbedea680056f52382ae548b2e4f3f38941e71cff8a78db1fffe18a1b336103
9fe76702af69334b7a1e6c303b7652f43698fad1153bb6c374b4c7fb98459ced;

C_{11} = 7bcd9ed0efc889fb3002c6cd635afe94d8fa6bbbebab076120018021148466798a1d71efea48b9caefbacd1d7d476e98dea2594ac06fd85d6bcaa4cd81f32d1b;

C_{12} = 378ee767f11631bad21380b00449b17acda43c32bcdf1d77f82012d430219f9b
5d80ef9d1891cc86e71da4aa88e12852faf417d5d9b21b9948bc924af11bd720.

6 Transformations

For calculating the hash-code $H(M)$ of the message $M \in V^*$, the following transformations are used:

$$X[k]: V_{512} \to V_{512}, \quad X[k](a) = k \oplus a, \; k, a \in V_{512}; \tag{3}$$

$$S: V_{512} \to V_{512}, \quad S(a) = S(a_{63} \| \ldots \| a_0) = \pi(a_{63}) \| \ldots \| \pi(a_0), \tag{4}$$

where $a = a_{63} \| \ldots \| a_0 \in V_{512}$, $a_i \in V_8$, $i = 0, \ldots, 63$;

$$P: V_{512} \to V_{512}, \quad P(a) = P(a_{63} \| \ldots \| a_0) = a_{\tau(63)} \| \ldots \| a_{\tau(0)}, \tag{5}$$

where $a = a_{63} \| \ldots \| a_0 \in V_{512}$, $a_i \in V_8$, $i = 0, \ldots, 63$;

$$L: V_{512} \to V_{512}, \quad L(a) = L(a_7 \| \ldots \| a_0) = l(a_7) \| \ldots \| l(a_0), \tag{6}$$

where $a = a_7 \| \ldots \| a_0 \in V_{512}$, $a_i \in V_{64}$, $i = 0, \ldots, 7$.

7 Round-functions

The hash-code value of the message $M \in V^*$ is calculated using the iterative procedure. Each iteration is performed using the round-function:

$$g_N: V_{512} \times V_{512} \rightarrow V_{512}, N \in V_{512}, \tag{7}$$

calculated as

$$g_N(h, m) = E(LPS(h \oplus N), m) \oplus h \oplus m, \tag{8}$$

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

The values $K_i \in V_{512}$, $i = 1, \ldots, 13$, are calculated as follows:

$$K_1 = K; \tag{9}$$

$$K_i = LPS(K_{i-1} \oplus C_{i-1}),\ i = 2, \ldots, 13. \tag{10}$$

For brevity, instead of $g_{0^{512}}$ we use the notation g_0.

8 Hash-function calculation procedure

The input for calculating the hash-code *H*(*M*) is the message $M \in V^*$ (to be hashed) and the initializing value $IV \in V_{512}$.

The algorithm for calculating the function *H* consists of the following stages.

8.1 Stage 1

Assign initial values to the following variables:

1.1. $h := IV$;

1.2. $N := 0^{512} \in V_{512}$;

1.3. $\sum := 0^{512} \in V_{512}$;

1.4. Go to stage 2.

8.2 Stage 2

2.1. Check the condition $|M| < 512$.

If it is true, then go to stage 3.

Else, perform the following calculations:

2.2. Calculate the subvector $m \in V_{512}$ of the message *M*: $M = M'\|m$. Then perform the following calculations:

2.3. $h := g_N(h, m)$.

2.4. $N := \mathrm{Vec}_{512}\,(\mathrm{Int}_{512}(N) \boxplus 512)$.

2.5. $\sum := \mathrm{Vec}_{512}\,(\mathrm{Int}_{512}(\Sigma) \boxplus \mathrm{Int}_{512}(m))$.

2.6. $M := M'$.

2.7. Go to step 2.1.

8.3 Stage 3

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

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

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

3.4. $\sum := \mathrm{Vec}_{512}\,(\mathrm{Int}_{512}(\Sigma) \boxplus \mathrm{Int}_{512}(m))$.

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

3.6. $h := \begin{cases} g_0(h, \Sigma), & \text{for the hash} - \text{function with } 512 - \text{bit hash} - \text{code;} \\ \mathrm{MSB}_{256}\big(g_0(h, \Sigma)\big), & \text{for the hash} - \text{function with } 256 - \text{bit hash} - \text{code.} \end{cases}$

3.7. End of the algorithm.

The value of the variable *h* (obtained in step 3.6) is the value of the hash-function *H*(*M*).

Annex A

(Informative)

Examples

This Annex is for information only and is not a normative part of this Standard.

The vectors from V^* are expressed in hexadecimal notation. The vector $A \in V_{4n}$, given in the form $a_{n-1} \dots a_0$, where $a_i \in \mathbb{Z}_{16}$, $i = 0, \dots, n\text{-}1$, is $\mathrm{Vec}_4(a_{n-1}) \| \dots \| \mathrm{Vec}_4(a_0)$.

A.1 Example 1

Let us calculate the hash-code of the following message:

M_1 = 32313039383736353433323130393837363534333231303938373635343332
3130393837363534333231303938373635343332313039383736353433323130.

A.1.1 For hash-functions with 512-bit hash-codes

Assign the following values to the variables:

$$h := IV = 0^{512};$$

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

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

The length of the message is $|M_1|$ = 504 < 512, so the incomplete block is padded:

m : = 0132313039383736353433323130393837363534333231303938373635343332
3130393837363534333231303938373635343332313039383736353433323130.

Calculate $K := LPS(h \oplus N) = LPS(0^{512})$.

After the transformation S:

$S(h \oplus N)$ = fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc
fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc,

after the transformation P:

$PS(h \oplus N)$ = fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc
fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc,

after the transformation L:

$$K := LPS(h \oplus N) =$$

b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574
b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574.

Then the transformation $E(K, m)$ is performed:

Iteration 1

K_1 = b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574
b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574,

$X[K_1](m)$ = 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.

Iteration 2

K_2 = d0b00807642fd78f13f2c3ebc774e80de0e902d23aef2ee9a73d010807dae9c1
88be14f0b2da27973569cd2ba051301036f728bd1d7eec33f4d18af70c46cf1e,

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

18e77571e703d19548075c574ce5e50e0480c9c5b9f21d45611ab86cf32e352a
d91854ea7df8f863d46333673f62ff2d3efae1cd966f8e2a74ce49902799aad4.

Iteration 3

K_3 = 9d4475c7899f2d0bb0e8b7dac6ef6e6b44ecf66716d3a0f16681105e2d13712a
1a9387ecc257930e2d61014a1b5c9fc9e24e7d636eb1607e816dbaf927b8fca9,

$LPSX[K_3]$ ... $LPSX[K_1](m)$ =

03dc0a9c64d42543ccdb62960d58c17e0b5b805d08a07406ece679d5f82b70fe
a22a7ea56e21814619e8749b308214575489d4d465539852cd4b0cd3829bef39.

Iteration 4

K_4 = 5c283daba5ec1f233b8c833c48e1c670dae2e40cc4c3219c73e58856bd96a72f
df9f8055ffe3c004c8cde3b8bf78f95f3370d0a3d6194ac5782487defd83ca0f,

$LPSX[K_4]$ ... $LPSX[K_1](m)$ =

dbee312ea7301b0d6d13e43855e85db81608c780c43675bc93cfd82c1b4933b3
898a35b13e1878abe119e4dffb9de4889738ca74d064cd9eb732078c1fb25e04.

Iteration 5

K_5 = 109f33262731f9bd569cbc9317baa551d4d2964fa18d42c41fab4e37225292ec
2fd97d74937847790463884 69ae195c436fa7cba93f8239ceb5ffc818826470c,

$LPSX[K_5]$ ... $LPSX[K_1](m)$ =

7fb3f15718d90e889f9fb7c38f527bec861c298afb9186934a93c9d96ade20df
109379bb9c1a1ffd0ad81fce7b45ccd54501e7d127e32874b5d7927b032de7a1.

Iteration 6

K_6 = b32c9b02667911cf8f8a0877be9a170757e25026ccf41e67c6b5da70b1b87474
3e1135cfbefe244237555c676c153d99459bc382573aee2d85d30d99f286c5e7,

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

95efa4e104f235824bae5030fe2d0f170a38de3c9b8fc6d8fa1a9adc2945c413
389a121501fa71a65067916b0c06f6b87ce18de1a2a98e0a64670985f47d73f1.

Iteration 7

K_7 = 8a13c1b195fd0886ac49989e7d84b08bc7b00e4f3f62765ece6050fcbabdc234
6c8207594714e8e9c9c7aad694edc922d6b01e17285eb7e61502e634559e32f1,

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

7ea4385f7e5e40103bfb25c67e404c7524eec43e33b1d06557469c6049854304
32b43d941b77ffd476103338e9bd5145d9c1e18b1f262b58a81dcefff6fc6535.

Iteration 8

K_8 =52cec3b11448bb8617d0ddfbc926f2e88730cb9179d6decea5acbffd323ec376
4c47f7a9e13bb1db56c342034773023d617ff01cc546728e71dff8de5d128cac,

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

b2426da0e58d5cfe898c36e797993f902531579d8ecc59f8dd8a60802241a456
1f290cf992eb398894424bf681636968c167e870967b1dd9047293331956daba.

Iteration 9

K_9 = f38c5b7947e7736d502007a05ea64a4eb9c243cb82154aa138b963bbb7f28e74
d4d710445389671291d70103f48fd4d4c01fc415e3fb7dc61c6088afa1a1e735,

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

5e0c9978670b25912dd1ede5bdd1cf18ed094d14c6d973b731d50570d0a9bca2
15415a15031fd20ddefb5bc61b96671d6902f49df4d2fd346ceebda9431cb075.

Iteration 10

K_{10} = 0740b3faa03ed39b257dd6e3db7c1bf56b6e18e40cdaabd30617cecbaddd618e
a5e61bb4654599581dd30c24c1ab877ad0687948286cfefaa7eef99f6068b315,

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

c1ddd840fe491393a5d460440e03bf451794e792c0c629e49ab0c1001782dd37
691cb6896f3e00b87f71d37a584c35b9cd8789fad55a46887e5b60e124b51a61.

Iteration 11

K_{11} = 185811cf3c2633aec8cfdfcae9dbb29347011bf92b95910a3ad71e5fca678e45
e374f088f2e5c29496e9695ce8957837107bb3aa56441af11a82164893313116,

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

3f75beaf2911c35d575088e30542b689c85b6b1607f8b800405941f5ab704284
7b9b08b58b4fbdd6154ed7b366fd3ee778ce647726ddb3c7d48c8ce8866a8435.

Iteration 12

K_{12} = 9d46bf66234a7ed06c3b2120d2a3f15e0fedd87189b75b3cd2f206906b5ee00d
c9a1eab800fb8cc5760b251f4db5cdef427052fa345613fd076451901279ee4c,

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

f35b0d889eadfcff73b6b17f33413a97417d96f0c4cc9d30cda8ebb7dcd5d1b0
61e620bac75b367370605f474ddc006003bec4c4d7ce59a73fbe6766934c55a2.

Iteration 13

K_{13} = 0f79104026b900d8d768b6e223484c9761e3c585b3a405a6d2d8565ada926c3f
7782ef127cd6b98290bf612558b4b60aa3cbc28fd94f95460d76b621cb45be70,

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

fc221dc8b814fc27a4de079d10097600209e5375776898961f70bded0647bd8f
1664cfa8bb8d8ff1e0df3e621568b66aa075064b0e81cce132c8d1475809ebd2.

The result of the transformation $g_N(h, m)$ is

h = fd102cf8812ccb1191ea34af21394f3817a86641445aa9a626488adb33738ebd
2754f6908cbbbac5d3ed0f522c50815c954135793fb1f5d905fee4736b3bdae2.

The variables N and Σ change their values to:

N = 0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000001f8,

Σ = 0132313039383736353433323130393837363534333231303938373635343332
3130393837363534333231303938373635343332313039383736353433323130.

The result of the transformation $g_0(h, N)$ is

h = 5c881fd924695cf196c2e4fec20d14b642026f2a0b1716ebaabb7067d4d59752
3d2db69d6d3794622147a14f19a66e7f9037e1d662d34501a8901a5de7771d7c.

The result of the transformation $g_0(h,\Sigma)$ is

h = 486f64c1917879417fef082b3381a4e211c324f074654c38823a7b76f830ad00
fa1fbae42b1285c0352f227524bc9ab16254288dd6863dccd5b9f54a1ad0541b.

The hash-code of the message M_1 is the value:

$H(M_1)$ = 486f64c1917879417fef082b3381a4e211c324f074654c38823a7b76f830ad00
fa1fbae42b1285c0352f227524bc9ab16254288dd6863dccd5b9f54a1ad0541b.

A.1.2 For hash-functions with 256-bit hash-codes

Assign the following values to the variables:

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

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

$$\sum := 0^{512}.$$

The length of the message is $|M_1| = 504 < 512$, so the incomplete block is padded:

m : = 0132313039383736353433323130393837363534333231303938373635343332
3130393837363534333231303938373635343332313039383736353433323130.

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

After the transformation S:

$$S(h \oplus N) =$$

eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee,

after the transformation P:

$$PS(h \oplus N) =$$

eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee,

after the transformation L:

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

23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15
23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15.

Then the transformation $E(K, m)$ is performed:

Iteration 1

K_1 = 23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15
23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15,

$X[K_1](m)$ = 22f7df708943682316f1dd72814b662d14f3db7483496e251afdd976854f6c27
12f5d778874d6a2110f7df708943682316f1dd72814b662d14f3db7483496e25,

$SX[K_1](m)$ = 65c061327951f35a99a6d819f5a29a0193d290ffa92ab25cf14b538aa8cc9d21
f0f4fe6dc93a7818e9c061327951f35a99a6d819f5a29a0193d290ffa92ab25c,

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

659993f1f0e99993c0a6d24bf4c0a6d261d89053fe61d8903219ff8a6d3219ff
79f5a9a8c979f5a951a22acc3a51a22af39ab29d78f39ab25a015c21185a015c,

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

e549368917a0a2611d5e08c9c2fd5b3c563f18c0f68c410d84ae9d5fbdfb9340
55650121b7aa6d7b3e7d09d46ac4358adaa6ae44fa3b0402c4166d2c3eb2ef02,

$K_1 \oplus C_1$ = 92cdb59aaeb185fcc80ec1c1701e230a0caf98039e3e8f03528b56cdc5fe9be9
68b90ed1221c36148187c448141b8c0026b39a767c0f1236fe458b1942dd1a12,

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

ecd95e282645a83930045858325f5afa2341dc110ad303110ef676d9ac63509b
f3a3041b65148f93f5c986f293bb7cfcef92288ac34df08f63c8f6362cd8f1f0,

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

ec30230ef3f5ef63d90441f6a3c992c85e58dc76048628f6285811d91bf28a36
26320aac6593c32c455fd36314bb4dd8a85a03508f7cf0f139fa119b93fc8ff0,

$$LPS(K_1 \oplus C_1) =$$

18ee8f3176b2ebea3bd6cb8233694cea349769df88be26bf451cfab6a904a549
da22de93a66a66b19c7e6b5eea633511e611d68c8401bfcd0c7d0cc39d4a5eb9.

Iteration 2

K_2 = 18ee8f3176b2ebea3bd6cb8233694cea349769df88be26bf451cfab6a904a549
da22de93a66a66b19c7e6b5eea633511e611d68c8401bfcd0c7d0cc39d4a5eb9,

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

c502dab7e79eb94013fcd1ba64def3b916f18b63855d43d22b77fca1452f9866
c2b45089c62e9d82edf1ef45230db9a23c9e1c521113376628a5f6a5dbc041b2.

Iteration 3

K_3 = aaa4cf31a265959157aec8ce91e7fd46bf27dee21164c5e3940bba1a519e9d1f
ce0913f1253e7757915000cd674be12cc7f68e73ba26fb00fd74af4101805f2d,

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

8e5a4fe41fc790af29944f027aa2f10105d65cf60a66e442832bb9ab5020dc54
772e36b03d4b9aa471037212cde93375226552392ef4d83010a007e1117a07b5.

Iteration 4

K_4 = 61fe0a65cc177af50235e2afadded326a5329a2236747bf8a54228aeca9c4585
cd801ea9dd743a0d98d01ef0602b0e332067fb5ddd6ac1568200311920839286,

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

dee0b40df69997afef726f03bdc13cb6ba9287698201296f2fd8284f06d33ea4
a850a0ff48026dd47c1e88ec813ed2eb1186059d842d8d17f0bfa259e56655b1.

Iteration 5

K_5 = 9983685f4fd3636f1fd5abb75fbf26a8e2934314aa2ecb3ee4693c86c06c7d4e
169bd540af75e1610a546acd63d960bad595394cc199bf6999a5d5309fe73d5a,

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

675ea894d326432e1af7b201bc369f8ab021f6fa58da09678ffc08ef30db43a3
7f1f7347cb77da0f6ba30c85848896c3bac240ab14144283518b89a33d0caf07.

Iteration 6

K_6 = f05772ae2ce7f025156c9a7fbcc6b8fdf1e735d613946e32922994e52820ffea
62615d907eb0551ad170990a86602088af98c83c22cdb0e2be297c13c0f7a156,

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

1bc204bf9506ee9b86bbcf82d254a112aea6910b6db3805e399cb718d1b33199
64459516967cee4e648e8cfbf81f56dc8da6811c469091be5123e6a1d5e28c73.

Iteration 7

K_7 = 5ad144c362546e4e46b3e7688829fbb77453e9c3211974330b2b8d0e6be2b5ac
c89eb6b35167f159b7b005a43e5959a651a9b18cfc8e4098fcf03d9b81cfbb8d,

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

f30d791ed78bdee819022a3d78182242124efcdd54e203f23fb2dc7f94338ff9
55a5afc15ffef03165263c4fdb36933aa982016471fbac9419f892551e9e568b.

Iteration 8

K_8 = 6a6cec9a1ba20a8db64fa840b934352b518c638ed530122a83332fe0b8efdac9
018287e5a9f509c78d6c746adcd5426fb0a0ad5790dfb73fc1f191a539016daa,

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

1fc20f1e91a1801a4293d3f3aa9e91560fcc3810bb15f3ee9741c9b87452519f
67cb9145519884a24de6db736a5cb1430da7458e5e51b80be5204ba5b2600177.

Iteration 9

K_9 = 99217036737aa9b38a8d6643f705bd51f351531f948f0fc5e35fa35fee9dd8bd
bb4c9d580a224e9cd82e0e2069fc49ed367d5f94374435382b8fb6a8f5dd0409,

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

1a52f09d1e81515a36171e0b1a2809c50359bed90f2e78cbd89b7d4afa6d0466
55c96bdae6ee97055cc7e857267c2ccf28c8f5dd95ed58a9a68c12663bb28967.

Iteration 10

K_{10} = 906763c0fc89fa1ae69288d8ec9e9dda9a7630e8bfd6c3fed703c35d2e62aeaf
f0b35d80a7317a7f76f83022f2526791ca8fdf678fcb337bd74fe5393ccb05d2,

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

764043744a0a93687e65aba8cfc25ec8714fb8e1bdc9ae2271e7205eaaa577c1
b3b83e7325e50a19bd2d56b061b5de39235c9c9fd95e071a1a291a5f24e8c774.

Iteration 11

K_{11} = 88ce996c63618e6404a5c8e03ee433854e2ae3eee68991bbbff3c29d38dadb6e
d6a1dae9a6dc6ddf52ce34af272f96d3159c8c624c3fe6e13d695c0bfc89add5,

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

9b1ce8ff26b445cb288c0aeccf84658eea91dbdf14828bf70110a5c9bd146cd9
646350cff4e90e7b63c5cc325e9b441081935f282d4648d9584f71860538f03b.

Iteration 12

K_{12} = 3e0a281ea9bd46063eec550100576f3a506aa168cf82915776b978fccaa32f38
b55f30c79982ca45628e8365d8798477e75a49c68199112a1d7b5a0f7655f2db,

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

133aeecede251eb81914b8ba48dcbc0b8a6fc63a292cc49043c3d3346b3f0829
a9cb71ecff25ed2a91bdcf8f649907c110cb76ff2e43100cdd4ba8a147a572f5.

Iteration 13

K_{13} = f0b273409eb31aebe432fbae1867212262c848422b6a92f93f6cbab54ed18b83
14b21cffc51e3fa319ff433e76ef6adb0ef9f5e03c907fa1fcf9eca06500bf03,

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

e3889d8e40960453fd26431450bb9d29e8a78e78024656697caf698125ee83aa
bd796d133a3bd28988428cb112766d1a1e32831f12d36fad21b2440122a5cdf6.

The result of the transformation $g_N(h, m)$ is

h = e3bbadbf78af3264c9137127608aa510de90ba4d3075665844965fb611dbb199
8d48552a0c0ce6bcba71bc802a4f5b2d2a07b12c22e25794178570341096fdc7.

The variables N and Σ change their values to:

N = 0000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000001f8,

Σ = 0132313039383736353433323130393837363534333231303938373635343332
3130393837363534333231303938373635343332313039383736353433323130.

The result of the transformation $g_0(h, N)$ is

h = 70f22bada4cfe18a6a56ec4b3f328cd40db8e1bf8a9d5f711d5efab11191279d
715aab7648d07eddbf87dc79c80516e6ffcbcf5678b0ac29ea00fa85c8173cc6.

The result of the transformation $g_0(h, \Sigma)$ is

h = 00557be5e584fd52a449b16b0251d05d27f94ab76cbaa6da890b59d8ef1e159d
2088e482e2acf564e0e9795a51e4dd261f3f667985a2fcc40ac8631faca1709a.

The hash-code of the message M_1 is the value:

$H(M_1)$ = 00557be5e584fd52a449b16b0251d05d27f94ab76cbaa6da890b59d8ef1e159d.

A.2 Example 2

Let us calculate the hash-code of the following message

M_2=fbe2e5f0eee3c820fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ce8f0f2e5e220e5d1.

A.2.1 For hash-functions with 512-bit hash-codes

Assign the following values to the variables:

$$h := IV = 0^{512};$$

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

$$\sum := 0^{512}.$$

The length of the message is $|M_2| = 576 > 512$, so a part of this message is initially processed

m : = fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ce8f0f2e5e220e5d1.

Calculate $K := LPS(h \oplus N) = LPS(0^{512})$.

After the transformation S:

$S(h \oplus N)$ = fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc
fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc,

after the transformation P:

$PS(h \oplus N)$ = fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc
fcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfcfc,

after the transformation L:

$LPS(h \oplus N)$ = b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574
b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574.

Then the transformation $E(K, m)$ is performed:

Iteration 1

K_1 = b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574
b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574b383fc2eced4a574,

$X[K_1](m)$ = 486906c521f45a8f43621cde3bf44599936b10ce2531558642a303de20388585
93790ed02b3685585b750fc32cf44d925d6214de3c0585585b730ecb2cf440a5,

$SX[K_1](m)$ = f29131ac18e613035196148598e6c8e8de6fe9e75c840c432c731185f906a8a8
de5404e1428fa8bf47354d408be63aecb79693857f6ea8bf473d04e48be6eb00,

$PSX[K_1](m)$ =
f251de2cde47b74791966f735435963d3114e911044d9304ac85e785e14085e4
18985cf9428b7f8be6e684068fe66ee613c80ca8a83aa8eb03e843a8bfecbf00,

$LPSX[K_1](m)$ =
909aa733e1f52321a2fe35bfb8f67e92fbc70ef544709d5739d8faaca4acf126
e83e273745c25b7b8f4a83a7436f6353753cbbbe492262cd3a868eace0104af1,

$K_1 \oplus C_1$ = 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.

Iteration 2

K_2 = d0b00807642fd78f13f2c3ebc774e80de0e902d23aef2ee9a73d010807dae9c1
88be14f0b2da27973569cd2ba051301036f728bd1d7eec33f4d18af70c46cf1e,

$LPSX[K_2]...LPSX[K_1](m)$ =
301aadd761d13df0b473055b14a2f74a45f408022aecadd4d5f19cab8228883a
021ac0b62600a495950c628354ffce1161c68b7be7e0c58af090ce6b45e49f16.

Iteration 3

K_3 = 9d4475c7899f2d0bb0e8b7dac6ef6e6b44ecf66716d3a0f16681105e2d13712a
1a9387ecc257930e2d61014a1b5c9fc9e24e7d636eb1607e816dbaf927b8fca9,

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

9b83492b9860a93cbca1c0d8e0ce59db04e10500a6ac85d4103304974e78d322
59ceff03fbb353147a9c948786582df78a34c9bde3f72b3ca41b9179c2cceef3.

Iteration 4

K_4 = 5c283daba5ec1f233b8c833c48e1c670dae2e40cc4c3219c73e58856bd96a72f
df9f8055ffe3c004c8cde3b8bf78f95f3370d0a3d6194ac5782487defd83ca0f,

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

e638e0a1677cdea107ec3402f70698a4038450dab44ac7a447e10155aa33ef1b
daf8f49da7b66f3e05815045fbd39c991cb0dc536e09505fd62d3c2cd00b0f57.

Iteration 5

K_5 = 109f33262731f9bd569cbc9317baa551d4d2964fa18d42c41fab4e37225292ec
2fd97d7493784779046388469ae195c436fa7cba93f8239ceb5ffc818826470c,

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

1c7c8e19b2bf443eb3adc0c787a52a173821a97bc5a8efea58fb8b27861829f6
dd5ff9c97865e08c1ac66f47392b578e21266e323a0aacedeec3ef0314f517c6.

Iteration 6

K_6 = b32c9b02667911cf8f8a0877be9a170757e25026ccf41e67c6b5da70b1b87474
3e1135cfbefe244237555c676c153d99459bc382573aee2d85d30d99f286c5e7,

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

48fecfc5b3eb77998fb39bfcccd128cd42fccb714221be1e675a1c6fdde7e311
98b318622412af7e999a3eff45e6d61609a7f2ae5c2ff1ab7ff3b37be7011ba2.

Iteration 7

K_7 = 8a13c1b195fd0886ac49989e7d84b08bc7b00e4f3f62765ece6050fcbabdc234
6c8207594714e8e9c9c7aad694edc922d6b01e17285eb7e61502e634559e32f1,

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

a48f8d781c2c5be417ae644cc2e15a9f01fcead3232e5bd53f18a5ab875cce1b
8a1a400cf48521c7ce27fb1e94452fb54de23118f53b364ee633170a62f5a8a9.

Iteration 8

K_8 = 52cec3b11448bb8617d0ddfbc926f2e88730cb9179d6decea5acbffd323ec376
4c47f7a9e13bb1db56c342034773023d617ff01cc546728e71dff8de5d128cac,

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

e8a31b2e34bd2ae21b0ecf29cc4c37c75c4d11d9b82852517515c23e81e906a4
51b72779c3087141f1a15ab57f96d7da6c7ee38ed25befbdef631216356ff59c.

Iteration 9

K_9 = f38c5b7947e7736d502007a05ea64a4eb9c243cb82154aa138b963bbb7f28e74
d4d710445389671291d70103f48fd4d4c01fc415e3fb7dc61c6088afa1a1e735,

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

34392ed32ea3756e32979cb0a2247c3918e0b38d6455ca88183356bf8e5877e5
5d542278a696523a8036af0f1c2902e9cbc585de803ee4d26649c9e1f00bda31.

Iteration 10

K_{10} = 0740b3faa03ed39b257dd6e3db7c1bf56b6e18e40cdaabd30617cecbaddd618e
a5e61bb4654599581dd30c24c1ab877ad0687948286cfefaa7eef99f6068b315,

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

6a82436950177fea74cce6d507a5a64e54e8a3181458e3bdfbdbc6180c9787de
7ccb676dd809e7cb1eb2c9ebd016561570801a4e9ce17a438b85212f4409bb5e.

Iteration 11

K_{11} = 185811cf3c2633aec8cfdfcae9dbb29347011bf92b95910a3ad71e5fca678e45
e374f088f2e5c29496e9695ce8957837107bb3aa56441af11a82164893313116,

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

7b97603135e2842189b0c9667596e96bd70472ccbc73ae89da7d1599c72860c2
85f5771088f1fb0f943d949f22f1413c991eafb51ab8e5ad8644770037765aec.

Iteration 12

K_{12} = 9d46bf66234a7ed06c3b2120d2a3f15e0fedd87189b75b3cd2f206906b5ee00d
c9a1eab800fb8cc5760b251f4db5cdef427052fa345613fd076451901279ee4c,

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

39ec8a88db635b46c4321adf41fd9527a39a67f6d7510db5044f05efaf721db5
cf976a726ef33dc4dfcda94033e741a463770861a5b25fefcb07281eed629c0e.

Iteration 13

K_{13} = 0f79104026b900d8d768b6e223484c9761e3c585b3a405a6d2d8565ada926c3f
7782ef127cd6b98290bf612558b4b60aa3cbc28fd94f95460d76b621cb45be70,

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

36959ac8fdda5b9e135aac3d62b5d9b0c279a27364f50813d69753b575e0718a
b8158560122584464f72c8656b53f7aec0bccaee7cfdcaa9c6719e3f2627227e.

The result of the transformation $g_N(h, m)$ is

h = cd7f602312faa465e3bb4ccd9795395de2914e938f10f8e127b7ac459b0c517b
98ef779ef7c7a46aa7843b8889731f482e5d221e8e2cea852e816cdac407c7af.

The variables N and Σ change their values to:

N = 0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000200,

Σ = fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ce8f0f2e5e220e5d1.

The length of the rest of the message is less than 512, so the incomplete block is padded.

m := 0000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000001fbe2e5f0eee3c820.

The result of the transformation $g_N(h, m)$ is

h = c544ae6efdf14404f089c72d5faf8dc6aca1db5e28577fc07818095f1df70661
e8b84d0706811cf92dffb8f96e61493dc382795c6ed7a17b64685902cbdc878e.

The variables N and Σ change their values to:

N = 0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000240,

Σ = fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ee4d3d8d6d104adf1.

The result of the transformation $g_0(h, N)$ is

h = 4deb6649ffa5caf4163d9d3f9967fbbd6eb3da68f916b6a09f41f2518b81292b
703dc5d74e1ace5bcd3458af43bb456e837326088f2b5df14bf83997a0b1ad8d.

The result of the transformation $g_0(h, \Sigma)$ is

h = 28fbc9bada033b1460642bdcddb90c3fb3e56c497ccd0f62b8a2ad4935e85f03
7613966de4ee00531ae60f3b5a47f8dae06915d5f2f194996fcabf2622e6881e.

The hash-code of the message M_2 is the value:

$H(M_2)$ = 28fbc9bada033b1460642bdcddb90c3fb3e56c497ccd0f62b8a2ad4935e85f03
7613966de4ee00531ae60f3b5a47f8dae06915d5f2f194996fcabf2622e6881e.

A.2.2 For hash-functions with 256-bit hash-codes

Assign the following values to the variables:

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

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

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

The length of the message is $|M_2| = 576 > 512$, so a part of this message is initially processed

m : = fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ce8f0f2e5e220e5d1.

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

After the transformation S:

$S(h \oplus N)$ =

eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee,

after the transformation P:

$PS(h \oplus N)$ =

eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee,

after the transformation L:

$K := LPS(h \oplus N)$ =

23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15
23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15.

Then the transformation $E(K, m)$ is performed:

Iteration 1

K_1 = 23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15
23c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f1523c5ee40b07b5f15,

$X[K_1](m)$ = d82f14ab5f5ba0eed3240eb0455bbff8032d02a05b9eafe7d2e511b05e977fe4
033f1cbe55997f39cb331dad525bb7f3cd2406b042aa7f39cb351ca5525bbac4,

$SX[K_1](m)$ =

8d4f93828747a76c49e204adc8473bd11101dda7470a415b832b77ad5dbc572d
111f14950ce8570be4aecd9f0e472fd2d9e231ad2c38570be46a14000e47a586,

$PSX[K_1](m)$ =

8d49118311e4d9e44fe2012b1faee26a9304dd7714cd311482ada7ad959fad00
87c8475d0c0e2c0e47470abce8473847a73b4157572f57a56cd15b2d0bd20b86,

$LPSX[K_1](m) =$

a3a72a2e0fb5e6f812681222fec037b0db972086a395a387a6084508cae13093
aa71d352dcbce288e9a39718a727f6fd4c5da5d0bc10fac3707ccd127fe45475,

$K_1 \oplus C_1$= 92cdb59aaeb185fcc80ec1c1701e230a0caf98039e3e8f03528b56cdc5fe9be9
68b90ed1221c36148187c448141b8c0026b39a767c0f1236fe458b1942dd1a12,

$S(K_1 \oplus C_1) =$

ecd95e282645a83930045858325f5afa2341dc110ad303110ef676d9ac63509b
f3a3041b65148f93f5c986f293bb7cfcef92288ac34df08f63c8f6362cd8f1f0,

$PS(K_1 \oplus C_1) =$

ec30230ef3f5ef63d90441f6a3c992c85e58dc76048628f6285811d91bf28a36
26320aac6593c32c455fd36314bb4dd8a85a03508f7cf0f139fa119b93fc8ff0,

$LPS(K_1 \oplus C_1) =$

18ee8f3176b2ebea3bd6cb8233694cea349769df88be26bf451cfab6a904a549
da22de93a66a66b19c7e6b5eea633511e611d68c8401bfcd0c7d0cc39d4a5eb9.

Iteration 2

K_2 = 18ee8f3176b2ebea3bd6cb8233694cea349769df88be26bf451cfab6a904a549
da22de93a66a66b19c7e6b5eea633511e611d68c8401bfcd0c7d0cc39d4a5eb9,

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

9f50697b1d9ce23680db1f4d35629778864c55780727aa79eb7bb7d648829cba
8674afdac5c62ca352d77556145ca7bc758679fbe1fbd32313ca8268a4a603f1.

Iteration 3

K_3 = aaa4cf31a265959157aec8ce91e7fd46bf27dee21164c5e3940bba1a519e9d1f
ce0913f1253e7757915000cd674be12cc7f68e73ba26fb00fd74af4101805f2d,

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

4183027975b257e9bc239b75c977ecc52ddad82c091e694243c9143a945b4d85
3116eae14fd81b14bb47f2c06fd283cb6c5e61924edfaf971b78d771858d5310.

Iteration 4

K_4 = 61fe0a65cc177af50235e2afadded326a5329a2236747bf8a54228aeca9c4585
cd801ea9dd743a0d98d01ef0602b0e332067fb5ddd6ac1568200311920839286,

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

0368c884fcee489207b5b97a133ce39a1ebfe5a3ae3cccb3241de1e7ad72857e
76811d324f01fd7a75e0b669e8a22a4d056ce6af3e876453a9c3c47c767e5712.

Iteration 5

K_5 = 9983685f4fd3636f1fd5abb75fbf26a8e2934314aa2ecb3ee4693c86c06c7d4e
169bd540af75e1610a546acd63d960bad595394cc199bf6999a5d5309fe73d5a,

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

c31433ceb8061e46440144e65553976512e5a9806ac9a2c771d5932d5f6508c5
b78e406c4efab98ac5529be0021b4d58fa26f01621eb10b43de4c4c47b63f615.

Iteration 6

K_6 = f05772ae2ce7f025156c9a7fbcc6b8fdf1e735d613946e32922994e52820ffea
62615d907eb0551ad170990a86602088af98c83c22cdb0e2be297c13c0f7a156,

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

5d0ae97f252ad04534503fe5f52e9bd07f483ee3b3d206beadc6e736c6e754bb
713f97ea7339927893eacf2b474a482cadd9ac2e58f09bcb440cf36c2d14a9b6.

Iteration 7

K_7 = 5ad144c362546e4e46b3e7688829fbb77453e9c3211974330b2b8d0e6be2b5ac
c89eb6b35167f159b7b005a43e5959a651a9b18cfc8e4098fcf03d9b81cfbb8d,

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

a59aa21e6ad3e330deedb9ab9912205c355b1c479fdfd89a7696d7de66fbf7d3
cec25879f7f1a8cca4c793d5f2888407aecb188bda375eae586a8cfd0245c317.

Iteration 8

K_8 = 6a6cec9a1ba20a8db64fa840b934352b518c638ed530122a83332fe0b8efdac9
018287e5a9f509c78d6c746adcd5426fb0a0ad5790dfb73fc1f191a539016daa,

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

9903145a39d5a8c83d28f70fa1fbd88f31b82dc7cfe17b54b50e276cb2c4ac68
2b4434163f214cf7ce6164a75731bcea5819e6a6a6fea99da9222951d2a28e01.

Iteration 9

K_9 = 99217036737aa9b38a8d6643f705bd51f351531f948f0fc5e35fa35fee9dd8bd
bb4c9d580a224e9cd82e0e2069fc49ed367d5f94374435382b8fb6a8f5dd0409,

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

330e6cb1d04961826aa263f2328f15b4f3370175a6a9fd6505b286efed2d8505
f71823337ef71513e57a700eb1672a685578e45dad298ee2223d4cb3fda8262f.

Iteration 10

K_{10} = 906763c0fc89fa1ae69288d8ec9e9dda9a7630e8bfd6c3fed703c35d2e62aeaf
f0b35d80a7317a7f76f83022f2526791ca8fdf678fcb337bd74fe5393ccb05d2,

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

ad347608443ab9c9bbb64f633a5749ab85c45d4174bfd78f6bc79fc4f4ce9ad1
dd71cb2195b1cfab8dcaaf6f3a65c8bb0079847a0800e4427d3a0a815f40a644.

Iteration 11

K_{11} = 88ce996c63618e6404a5c8e03ee433854e2ae3eee68991bbbff3c29d38dadb6e
d6a1dae9a6dc6ddf52ce34af272f96d3159c8c624c3fe6e13d695c0bfc89add5,

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

a065c55e2168c31576a756c7ecc1a9129cd3d207f8f43073076c30e111fd5f11
9095ca396e9fb78a2bf4781c44e845e447b8fc75b788284aae27582212ec23ee.

Iteration 12

K_{12} = 3e0a281ea9bd46063eec550100576f3a506aa168cf82915776b978fccaa32f38
b55f30c79982ca45628e8365d8798477e75a49c68199112a1d7b5a0f7655f2db,

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

2a6549f7a5cd2eb4a271a7c71762c8683e7a3a906985d60f8fc86f64e35908b2
9f83b1fe3c704f3c116bdfe660704f3b9c8a1d0531baaffaa3940ae9090a33ab.

Iteration 13

K_{13} = f0b273409eb31aebe432fbae1867212262c848422b6a92f93f6cbab54ed18b83
14b21cffc51e3fa319ff433e76ef6adb0ef9f5e03c907fa1fcf9eca06500bf03,

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

dad73ab73b7e345f46435c690f05e94a5cb272d242ef44f6b0a4d5d1ad888331
8b31ad01f96e709f08949cd8169f25e09273e8e50d2ad05b5f6de6496c0a8ca8.

The result of the transformation $g_N(h, m)$ is

h = 203cc15dd55fcaa5b7a3bd98fb2408a67d5b9f33a80bb50540852b204265a2c1
aaca5efe1d8d51b2e1636e34f5becc077d930114fefaf176b69c15ad8f2b6878.

The variables N and Σ changed their values to:

N = 0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000200,

Σ = fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ce8f0f2e5e220e5d1.

The length of the rest of the message is less than 512, so the incomplete block is padded:

m = 0000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000001fbe2e5f0eee3c820.

The result of the transformation $g_N(h, m)$ is

h = a69049e7bd076ab775bc2873af26f098c538b17e39a5c027d532f0a2b3b56426
c96b285fa297b9d39ae6afd8b9001d97bb718a65fcc53c41b4ebf4991a617227.

The variables N and Σ change their values to:

N = 00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000240,

Σ = fbeafaebef20fffbf0e1e0f0f520e0ed20e8ece0ebe5f0f2f120fff0eeec20f1
20faf2fee5e2202ce8f6f3ede220e8e6eee1e8f0f2d1202ee4d3d8d6d104adf1.

The result of the transformation $g_0(h, N)$ is

h = aee3bd55ea6f387bcf28c6dcbdbbfb3ddacc67dcc13dbd8d548c6bf808111d4b
75b8e74d2afae960835ae6a5f03575559c9fd839783ffcd5cf99bd61566b4818.

The result of the transformation $g_0(h, \Sigma)$ is

h = 508f7e553c06501d749a66fc28c6cac0b005746d97537fa85d9e40904efed29d
c345e53d7f84875d5068e4eb743f0793d673f09741f9578471fb2598cb35c230.

The hash-code of the message M_2 is the value:

$H(M_2)$ = 508f7e553c06501d749a66fc28c6cac0b005746d97537fa85d9e40904efed29d.

Bibliography[1]

[1] ISO 2382-2:1976	Data processing – Vocabulary – Part 2: Arithmetic and logic operations
[2] ISO/IEC 9796-2:2010	Information technology – Security techniques – Digital signature schemes giving message recovery – Part 2: Integer factorization based mechanisms
[3] ISO/IEC 9796-3:2006	Information technology – Security techniques – Digital signature schemes giving message recovery – Part 3: Discrete logarithm based mechanisms
[4] ISO/IEC 14888-1:2008	Information technology – Security techniques – Digital signatures with appendix – Part 1: General
[5] ISO/IEC 14888-2:2008	Information technology – Security techniques – Digital signatures with appendix – Part 2: Integer factorization based mechanisms
[6] ISO/IEC 14888-3:2006	Information technology – Security techniques – Digital signatures with appendix – Part 3: Discrete logarithm based mechanisms
[7] ISO/IEC 14888-3:2006/Amd 1:2010	Information technology – Security techniques – Digital signatures with appendix – Part 3: Discrete logarithm based mechanisms. Amendment 1. Elliptic Curve Russian Digital Signature Algorithm, Schnorr Digital Signature Algorithm, Elliptic Curve Schnorr Digital Signature Algorithm, and Elliptic Curve Full Schnorr Digital Signature Algorithm
[8] ISO/IEC 10118-1:2000	Information technology – Security techniques – Hash-functions – Part 1: General
[9] ISO/IEC 10118-2:2010	Information technology – Security techniques – Hash-functions – Part 2: Hash-functions using an n-bit block cipher
[10] ISO/IEC 10118-3:2004	Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions
[11] ISO/IEC 10118-4:1998	Information technology -- Security techniques – Hash-functions -- Part 4: Hash-functions using modular arithmetic

[1] These International ISO/IEC standards are available at the FSUE “Standartinform“ of the Federal Agency on Technical Regulation and Metrology

УДК 681.3.06:006.354 ОКС 35. 040 ОКСТУ 5002 П 85

Keywords: information technology, cryptographic data security, hash-function, [electronic] digital signature, asymmetric cryptographic algorithm, information processing systems, message protection, signature verification.