GM/T 0001.3

ZUC Stream Cipher Algorithm

Part 3: Integrity Algorithm

Cryptography Standardization

Technical Committee of China

Issued on 2012-03-21 Translated on 2024-10-30

Contents

Foreword

GM/T 0001 " ZUC Stream Cipher Algorithm" consists of three parts:

Part 1: Algorithm Description;

Part 2: Confidentiality Algorithm;

Part 3: Integrity Algorithm.

This part is Part 3 of GM/T 0001.

1 Scope

This part of GM/T 0001 describes the integrity algorithm based on the ZUC stream cipher algorithm. It is applicable to the development, testing, and use of products related to the integrity algorithm based on the ZUC stream cipher algorithm.

2 Normative References

The following documents are essential for the application of this document. For dated references, only the dated version applies to this document. For undated references, the latest version (including all amendments) applies to this document.

- GM/T 0001.1 ZUC Stream Cipher Algorithm Part 1: Algorithm Description

3 Terms and Definitions

The terms and definitions defined in GM/T 0001.1 apply to this document.

4 Symbols and Abbreviations

4.1 Symbols

The following symbols apply to this document:

$\oplus$	Bitwise XOR operation
||	String or byte string concatenation
$\lceil x \rceil$	Smallest integer no less than x
$\ll k$	Left shift by k bits

4.2 Abbreviations

The following abbreviations apply to this document:

IK: Integrity Key

IV: Initial Vector

LTE: Long Term Evolution

MAC: Message Authentication Code

5 Algorithm Description

5.1 Algorithm Inputs and Outputs

The input parameters of this algorithm are shown in Table 1, and the output parameter is shown in Table 2.

Table 1 The inputs

Parameter	Size(bits)	Remark
IK	128	Integrity key
IV	128	Initialization Vector
LENGTH	32	Bit length of the input message stream
M	LENGTH	Input message stream with length LENGTH

Table 2 The output

Parameter	Size(bits)	Remark
MAC	32	The MAC

5.2 Algorithm Workflow

5.2.1 Key Stream Generation

Let $L = \lceil \text{LENGTH}/32 \rceil + 2$. Using the integrity key (IK) and the initial vector (IV), and L as input parameters and generate a key stream according to the method described in section 5.6 of GM/T 0001.1. Represent the generated key stream as a bit string $k[0],k[1],\ldots,k[32\times L-1]$, where $k[0]$ is the highest bit of the first key word and $k[31]$ is the lowest bit of the first key word, continuing in this manner.

For $i=0,1,2,\ldots,32\times(L-1)$, let

$$k_i=k[i] \,||\, k[i+1] \,||\, \ldots \,||\, k[i+31],$$

where k_i is a 32-bit word.

For 3GPP LTE application scenarios, refer to Appendix A for the initialization method of the initial vector (IV).

5.2.2 MAC Calculation

Let T be a 32-bit variable. Initialize T=0.

For i=0,1,…,LENGTH−1, If M[i]=1, then

$$T=T\oplus k_i$$

Calculate

$$T=T\oplus k_{\mathrm{LENGTH}}$$

Finally, calculate the MAC

$$\mathrm{MAC}=T\oplus k_{32\times(\mathrm{L}-1)}$$

For 3GPP LTE application scenarios, refer to Appendix B for algorithm calculation examples.

Appendix A
(Informative Appendix)
Parameter Initialization in 3GPP LTE

A.1 Input and Output Parameters in 3GPP LTE

The assignment of input and output parameters in 3GPP LTE is specified in Tables A.1 and A.2.

Table A.1 The inputs

Parameter	Size (bits)	Remark
COUNT	32	The counter
BEARER	5	The bearer identity
DIRECTION	1	The direction of transmission
IK	128	The integrity key
LENGTH	32	The length of the input message
M	LENGTH	The input message

Table A.2 The output

Parameter	Size(bits)	Remark
MAC	32	The MAC

A.2 Parameter Initialization

The initialization process constructs the initial vector (IV) based on the counter (COUNT), bearer identifier (BEARER), and transmission direction (DIRECTION) as specified in Table A.1.

Define the counter as

$$\text{COUNT}=\text{COUNT}[0] \,||\, \text{COUNT}[1] \,||\, \text{COUNT}[2] \,||\, \text{COUNT}[3]$$

where COUNT[i] (for i=0,1,2,3) is an 8-bit byte.

Set the initial vector (IV) as:

$$\text{IV}=\text{IV}[0] \,||\, \text{IV}[1] \,||\, \text{IV}[2] \,||\, \ldots \,||\, \text{IV}[15]$$

where IV[i] (for $0 \le i < 16$) is an 8-bit byte.

Calculate the initial vector (IV) as follows:

IV[0] = COUNT[0], IV[1] = COUNT[1],

IV[2] = COUNT[2], IV[3] = COUNT[3],

IV[4] = BEARER ‖ 000_2, IV[5] = 00000000_2,

IV[6] = 00000000_2, IV[7] = 00000000_2,

IV[8] = IV[0] $\oplus$ (DIRECTION $\ll$ 7), IV[9] = IV[1],

IV[10] = IV[2], IV[11] = IV[3],

IV[12] = IV[4], IV[13] = IV[5],

IV[14] = IV[6] $\oplus$ (DIRECTION $\ll$ 7), IV[15] = IV[7].

For algorithm calculation examples in 3GPP LTE, refer to Appendix B.

Appendix B

(Informative Appendix)

Algorithm Calculation Example in 3GPP LTE

The following is an example of the algorithm calculation in 3GPP LTE. The data is represented in hexadecimal.

Example 1: First Calculation Instance:

IK = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
COUNT = 0
BEARER = 0
DIRECTION = 0
LENGTH = 1
M: 00000000
MAC: c8a9595e

Example 2: The Second Calculation Instance:

IK = c9 e6 ce c4 60 7c 72 db 00 0a ef a8 83 85 ab 0a
COUNT = a94059da
BEARER = a
DIRECTION = 1
LENGTH = 241
M: 983b41d4 7d780c9e 1ad11d7e b70391b1 de0b35da 2dc62f83 e7b78d63 06ca0ea0 7e941b7b e91348f9 fcb170e2 217fecd9 7f9f68ad b16e5d7d 21e569d2 80ed775c ebde3f40 93c53881 00000000
MAC: fae8ff0b

Example 3: The Third Calculation Instance:

IK = 6b 8b 08 ee 79 e0 b5 98 2d 6d 12 8e a9 f2 20 cb
COUNT = 561eb2dd
BEARER = 1c
DIRECTION = 0
LENGTH = 1626
M: 5bad7247 10ba1c56 d5a315f8 d40f6e09 3780be8e 8de07b69 92432018 e08ed96a 5734af8b ad8a575d 3a1f162f 85045cc7 70925571 d9f5b94e 454a77c1 6e72936b f016ae15 7499f054 3b5d52ca a6dbeab6 97d2bb73 e41b8075 dce79b4b 86044f66 1d4485a5 43dd7860 6e0419e8 059859d3 cb2b67ce 0977603f 81ff839e 33185954 4cfbc8d0 0fef1a4c 8510fb54 7d6b06c6 11ef44f1 bce107cf a45a06aa b360152b 28dc1ebe 6f7fe09b 0516f9a5 b02a1bd8 4bb0181e

2e89e19b d8125930 d178682f 3862dc51 b636f04e 720c47c3 ce51ad70 d94b9b22 55fbae90
6549f499 f8c6d399 47ed5e5d f8e2def1 13253e7b 08d0a76b 6bfc68c8 12f375c7 9b8fe5fd
85976aa6 d46b4a23 39d8ae51 47f680fb e70f978b 38effd7b 2f7866a2 2554e193 a94e98a6
8b74bd25 bb2b3f5f b0a5fd59 887f9ab6 8159b717 8d5b7b67 7cb546bf 41eadca2 16fc1085
0128f8bd ef5c8d89 f96afa4f a8b54885 565ed838 a950fee5 f1c3b0a4 f6fb71e5 4dfd169e
82cecc72 66c850e6 7c5ef0ba 960f5214 060e71eb 172a75fc 1486835c bea65344 65b055c9
6a72e410 52241823 25d83041 4b40214d aa8091d2 e0fb010a e15c6de9 0850973b df1e423b
e148a237 b87a0c9f 34d4b476 05b803d7 43a86a90 399a4af3 96d3a120 0a62f3d9 507962e8
e5bee6d3 da2bb3f7 237664ac 7a292823 900bc635 03b29e80 d63f6067 bf8e1716 ac25beba
350deb62 a99fe031 85eb4f69 937ecd38 7941fda5 44ba67db 09117749 38b01827 bcc69c92
b3f772a9 d2859ef0 03398b1f 6bbad7b5 74f7989a 1d10b2df 798e0dbf 30d65874 64d24878
cd00c0ea ee8a1a0c c753a279 79e11b41 db1de3d5 038afaf4 9f5c682c 3748d8a3 a9ec54e6
a371275f 1683510f 8e4f9093 8f9ab6e1 34c2cfdf 4841cba8 8e0cff2b 0bcc8e6a dcb71109
b5198fec f1bb7e5c 531aca50 a56a8a3b 6de59862 d41fa113 d9cd9578 08f08571 d9a4bb79
2af271f6 cc6dbb8d c7ec36e3 6be1ed30 8164c31c 7c0afc54 1c000000

MAC: 0ca1279

References

[1] ETSI/SAGE TS 35.221. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 1: 128-EEA3 and 128-EIA3 Specification.

[2] ETSI/SAGE TS 35.222. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 2: ZUC Specification.

[3] ETSI/SAGE TS 35.223. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 3: Implementor's Test Data.

[4] ETSI/SAGE TR 35.924. Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3. Document 4: Design and Evaluation Report.