ETSI/SAGE Specification

Version: 1.1
Date: 4th Jan. 2011

Specification of the 3GPP Confidentiality and Integrity Algorithms 128-EEA3 & 128-EIA3

Document 3: Implementor's Test Data

Blank Page

PREFACE

This specification has been prepared by the 3GPP Task Force, and gives detailed test data for implementors of the algorithm set. It provides visibility of the internal state of the ZUC algorithm to aid in its realization, as well as black box input-output test data. The test data has been selected to give a high degree of confidence that the implementation is correct. However, no claim is made that conformance with this test data guarantees a correct implementation. This document is the third of three, which between them form the entire specification of the 3GPP Confidentiality and Integrity Algorithms:

- Specification of the 3GPP Confidentiality and Integrity Algorithms ***128-EEA3*** & ***128-EIA3***.
  Document 1: ***128-EEA3*** and ***128-EIA3*** Specifications.

- Specification of the 3GPP Confidentiality and Integrity Algorithms ***128-EEA3*** & ***128-EIA3***.
  Document 2: **ZUC** Specification.

- Specification of the 3GPP Confidentiality and Integrity Algorithms ***128-EEA3*** & ***128-EIA3***.
  Document 3: Implementor's Test Data.

This document is purely informative. The normative part of the specification of the ***128-EEA3*** (confidentiality) and the ***128-EIA3*** (integrity) algorithms is in the main body of document 1. The normative part of the specification of **ZUC** is found in document 2.

Blank Page

TABLE OF CONTENTS

1 OUTLINE OF THE DESIGN CONFORMANCE TEST DATA 8

2 INTRODUCTORY INFORMATION 8

2.1 Introduction 8

2.2 Radix 8

2.3 Bit/Byte ordering 8

2.4 Presentation of input/output data 8

2.5 Coverage 8

3 ZUC 9

3.1 Overview 9

3.2 Format 9

3.3 Test Set 1 9

3.4 Test Set 2 10

3.5 Test Set 3 12

3.6 Test Set 4 13

4 CONFIDENTIALITY ALGORITHM 128-*EEA3* 15

4.1 Overview 15

4.2 Format 15

4.3 Test Set 1 15

4.4 Test Set 2 16

4.5 Test Set 3 17

4.6 Test Set 4 17

4.7 Test Set 5 18

5 INTEGRITY ALORITHM 128-*EIA3* 19

5.1 Overview 19

5.2 Test Set 1 19

5.3 Test Set 2 20

5.4 Test Set 3 20

5.5 Test Set 4 20

5.6 Test Set 5 21

REFERENCES

[1] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security Architecture (3G TS 33.102 version 6.3.0)

[2] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Cryptographic Algorithm Requirements; (3G TS 33.105 version 6.0.0)

[3] Specification of the 3GPP Confidentiality and Integrity Algorithms ***128-EEA3*** & ***128-EIA3***.
Document 1: ***128-EEA3*** and ***128-EIA3*** specifications.

[4] Specification of the 3GPP Confidentiality and Integrity Algorithms ***128-EEA3*** & ***128-EIA3***.
Document 2: **ZUC** specification.

[5] Specification of the 3GPP Confidentiality and Integrity Algorithms ***128-EEA3*** & ***128-EIA3***.
Document 3: Implementor's Test Data.

[6] Specification of the 3GPP Confidentiality and Integrity Algorithms ***128-EEA3*** & ***128-EIA3***.
Document 4: Design and Evaluation Report.

1 OUTLINE OF THE TEST DATA

Section 2 introduces the algorithms and describes the notation used in the subsequent sections.

Section 3 provides test data for the **ZUC** algorithm.

Section 4 provides test data for the confidentiality algorithm ***128-EEA3***.

Section 5 provides test data for the integrity algorithm ***128-EIA3***.

2 INTRODUCTORY INFORMATION

2.1 Introduction

Within the security architecture of the 3GPP system there are two standardised algorithms; a confidentiality algorithm ***128-EEA3***, and an integrity algorithm ***128-EIA3***. These algorithms are specified in a companion document[3].

This document provides sets of input/output test data for 'black box' testing of physical realisations of the **ZUC**, ***128-EEA3*** and ***128-EIA3*** algorithms. It also provides some internal values within the ZUC algorithm, to aid in its implementation.

2.2 Radix

Unless stated otherwise, all test data values presented in this document are in hexadecimal.

2.3 Bit/Byte ordering

All data variables in this specification are presented with the most significant bit (or byte) on the left hand side and the least significant bit (or byte) on the right hand side.

2.4 Presentation of input/output data

The basic data processed by the ***128-EEA3*** and ***128-EIA3*** algorithms are bit streams. In general in this document the data is presented in hexadecimal format as bytes, thus the last byte shown as part of an input or output data stream may include between 0 and 7 bits that are ignored once the **LENGTH** parameter is taken into account. (The least significant bits of the byte are ignored).

2.5 Coverage

For each of the algorithms the test data have been selected such that, provided the entire test set is run:

- Each key bit will have been in both the '1' and the '0' states,
- Each bit of the initialisation fields (COUNT, FRESH, BEARER, DIRECTION) will have been in both the '1' and the '0' states,
- If the test set for ***128-EEA3*** is run every entry in the internal S-boxes S_0 and S_1 will have been accessed.

3 ZUC

3.1 Overview

The test data sets presented here are for the ZUC stream cipher algorithm.

3.2 Format

Each test set starts by showing the input and output data values.

This is followed by a table showing the state of the LFSR at the beginning of the computation.

Then for the first 10 steps of the initialisation the content of X_0, X_1, X_2, X_3, R_1, R_2 is given in a table. Steps are indexed by *t* (for "time").

Then the state of the LFSR and the nonlinear function F at the end of the initialization is given.

For the first 3 steps of keystream generation X_0, X_1, X_2, X_3, R_1, R_2 are given in a table.

3.3 Test Set 1

input:

Key: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

IV: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

output:

z_1: 27bede74

z_2: 018082da

Initialisation Mode

LFSR-state at the beginning:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	0044d700	0026bc00	00626b00	00135e00	00578900	0035e200	00713500	0009af00
8	004d7800	002f1300	006bc400	001af100	005e2600	003c4d00	00789a00	0047ac00

t	X_0	X_1	X_2	X_3	R_1	R_2	W	S_{15}
0	008f9a00	f100005e	af00006b	6b000089	67822141	62a3a55f	008f9a00	4563cb1b
1	8ac7ac00	260000d7	780000e2	5e00004d	474a2e7e	119e94bb	4fe932a0	28652a0f
2	50cacb1b	4d000035	13000013	890000c4	c29687a5	e9b6eb51	291f7a20	7464f744
3	e8c92a0f	9a0000bc	c400009a	e2000026	29c272f3	8cac7f5d	141698fb	3f5644ba

4	7eacf744	ac000078	f100005e	350000af	2c85a655	24259cb0	e41b0514	006a144c
5	00d444ba	cb1b00f1	260000d7	af00006b	cbfbc5c0	44c10b3a	50777f9f	07038b9b
6	0e07144c	2a0f008f	4d000035	780000e2	e083c8d3	7abf7679	0abddcc6	69b90e2b
7	d3728b9b	f7448ac7	9a0000bc	13000013	147e14f4	b669e72d	aeb0b9c1	62a913ea
8	c5520e2b	44ba50ca	ac000078	c400009a	982834a0	f095d694	8796020c	7b591cc0
9	f6b213ea	144ce8c9	cb1b00f1	f100005e	e14727d6	d0225869	5f2ffdde	70e21147

LFSR-state after completion of the initialisation mode:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	7ce15b8b	747ca0c4	6259dd0b	47a94c2b	3a89c82e	32b433fc	231ea13f	31711e42
8	4ccce955	3fb6071e	161d3512	7114b136	5154d452	78c69a74	4f26ba6b	3e1b8d6a

FSM-state after completion of the initialisation mode:

R_1 = 14cfd44c

R_2 = 8c6de800

Keystream mode

t	X_0	X_1	X_2	X_3	R_1	R_2	z	S_{15}
0	7c37ba6b	b1367f6c	1e426568	dd0bf9c2	3512bf50	a0920453	286dafe5	7f08e141
1	fe118d6a	d4522c3a	e955463d	4c2be8f9	c7ee7f13	0c0fa817	27bede74	3d383d04
2	7a70e141	9a74e229	071e62e2	c82ec4b3	dde63da7	b9dd6a41	018082da	13d6d780

3.4 Test Set 2

input:

Key: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff

IV: ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff

output:

z_1: 0657cfa0

z_2: 7096398b

Initialisation Mode

LFSR-state at the beginning:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	7fc4d7ff	7fa6bcff	7fe26bff	7f935eff	7fd789ff	7fb5e2ff	7ff135ff	7f89afff
8	7fcd78ff	7faf13ff	7febc4ff	7f9af1ff	7fde26ff	7fbc4dff	7ff89aff	7fc7acff

t	X_0	X_1	X_2	X_3	R_1	R_2	W	S_{15}
0	ff8f9aff	f1ffff5e	afffff6b	6bffff89	b51c2110	30a3629a	ff8f9aff	76e49a1a
1	edc9acff	26ffffd7	78fffe2	5effff4d	a75b6f4b	1a079628	8978f089	5e2d8983
2	bc5b9a1a	4dffff35	13ffff13	89ffffc4	9810b315	99296735	35088b79	5b9484b8
3	b7298983	9afffbc	c4ffff9a	e2ffff26	4c5bd8eb	2d577790	c862a1cb	2db5c755
4	5b6b84b8	acffff78	f1ffff5e	35ffffaf	a13dcb66	21d0939f	4487d3e3	60579232
5	c0afc755	9a1afff1	26ffffd7	afffff6b	cc5ce260	0c50a8e2	83629fd2	29d4e960
6	53a99232	8983ff8f	4dffff35	78fffe2	dada0730	b516b128	ac461934	5e02d9e5
7	bc05e960	84b8edc9	9afffbc	13ffff13	2bbe53a4	12a8a16e	1bf69f78	7904dddc
8	f209d9e5	c755bc5b	acffff78	c4ffff9a	4a90d661	d9c744b4	ec602baf	0c3c9016
9	1879dddc	9232b729	9a1afff1	f1ffff5e	76bc13d7	a49ea404	2cb05071	0b9d257b

LFSR-state after completion of the initialisation mode:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	09a339ad	1291d190	25554227	36c09187	0697773b	443cf9cd	6a4cd899	49e34bd0
8	56130b14	20e8f24c	7a5b1dcc	0c3cc2d1	1cc082c8	7f5904a2	55b61ce8	1fe46106

FSM-state after completion of the initialisation mode:

R_1 = b8017bd5

R_2 = 9ce2de5c

Keystream mode

t	X_0	X_1	X_2	X_3	R_1	R_2	z	S_{15}
0	3fc81ce8	c2d141d1	4bd08879	42271346	aa131b11	09d7706c	668b56df	13f56dbf
1	27ea6106	82c8f4b6	0b14d499	91872523	251e7804	caac5d66	0657cfa0	0c0fe353
2	181f6dbf	04a21879	f24c93c6	773b4aaa	d94e9228	91d88fba	7096398b	10f1eecf

3.5 Test Set 3

input:

Key: 3d 4c 4b e9 6a 82 fd ae b5 8f 64 1d b1 7b 45 5b

IV: 84 31 9a a8 de 69 15 ca 1f 6b da 6b fb d8 c7 66

output:

z_1: 14f1c272

z_2: 3279c419

Initialisation Mode

LFSR-state at the beginning:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	1ec4d784	2626bc31	25e26b9a	74935ea8	355789de	4135e269	7ef13515	5709afca
8	5acd781f	47af136b	326bc4da	0e9af16b	58de26fb	3dbc4dd8	22f89ac7	2dc7ac66

t	X_0	X_1	X_2	X_3	R_1	R_2	W	S_{15}
0	5b8f9ac7	f16b8f5e	afca826b	6b9a3d89	9c62829f	5df00831	5b8f9ac7	3c7b93c0
1	78f7ac66	26fb64d7	781ffde2	5ea84c4d	3d533f3a	80ff1faf	4285372a	41901ee9
2	832093c0	4dd81d35	136bae13	89de4bc4	2ca57e9d	d1db72f9	3f72cca9	411efa99
3	823d1ee9	9ac7b1bc	c4dab59a	e269e926	0e8dc40f	60921a4f	8073d36d	24b3f49f
4	4967fa99	ac667b78	f16b8f5e	35156aaf	16c81467	da8e7d8a	a87c58e5	74265785
5	e84cf49f	93c045f1	26fb64d7	afca826b	50c9eaa4	3c3b2dfd	d9135e82	481c5b9d
6	90385785	1ee95b8f	4dd81d35	781ffde2	59857b80	be0fbdc1	fd2ceb1e	4b7f87ed
7	96ff5b9d	fa9978f7	9ac7b1bc	136bae13	9528f8ea	bcc7f7eb	8d89ddde	0e633ce7
8	1cc687ed	f49f8320	ac667b78	c4dab59a	c59d2932	e1098a64	46b676f2	643ae5a6
9	c8753ce7	5785823d	93c045f1	f16b8f5e	755ebae8	3f9e6e86	eef1a039	625ac5d7

LFSR-state after completion of the initialisation mode:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	10da5941	5b6acbf6	17060ce1	35368174	5cf4385a	479943df	2753bab2	73775d6a
8	43930a37	77b4af31	15b2e89f	24ff6e20	740c40b9	026a5503	194b2a57	7a9a1cff

FSM-state after completion of the initialisation mode:

R_1 = 860a7dfa

R_2 = bf0e0ffc

Keystream mode

t	X_0	X_1	X_2	X_3	R_1	R_2	z	S_{15}
0	f5342a57	6e20ef69	5d6a8f32	0ce121b4	129d8b39	2d7cdce1	3ead461d	3d4aa9e7
1	7a951cff	40b92b65	0a374ea7	8174b6d5	ab7cf688	c1598aa6	14f1c272	71db1828
2	e3b6a9e7	550349fe	af31e6ee	385a2e0c	3cec1a4a	9053cc0e	3279c419	258937da

3.6 Test Set 4

input:

Key: 4d 32 0b fa d4 c2 85 bf d6 b8 bd 00 f3 9d 8b 41

IV: 52 95 9d ab a0 bf 17 6e ce 2d c3 15 04 9e b5 74

output:

z_1: ed4400e7

z_2: 0633e5c5

……

Z_{2000}: 7a574cdb

Initialisation Mode

LFSR-state at the beginning:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	26c4d752	1926bc95	05e26b9d	7d135eab	6a5789a0	6135e2bf	42f13517	5f89af6e
8	6b4d78ce	5c2f132d	5eebc4c3	001af115	79de2604	4ebc4d9e	45f89ab5	20c7ac74

t	X_0	X_1	X_2	X_3	R_1	R_2	W	S_{15}
0	418f9ab5	f115b85e	af6ec26b	6b9d4d89	8d1ca588	4a6d1a2f	418f9ab5	30605895
1	60c0ac74	2604bdd7	78ce85e2	5eab324d	df86fb65	bf68d8ca	3849242b	0c59d998
2	18b35895	4d9e0035	132dbf13	89a00bc4	89b82f4b	b04ce56f	869e7cba	0368ac2f

3	06d1d998	9ab5f3bc	c4c3d69a	e2bffa26	65351ba4	b802da85	3fb6dc42	624f5dfa
4	c49eac2f	ac749d78	f115b85e	3517d4af	90956c83	9bd47e1e	59ae9210	3484ef0a
5	69095dfa	58958bf1	2604bdd7	af6ec26b	a1d3a6f9	3c10dfa7	9570af97	7317f442
6	e62fef0a	d998418f	4d9e0035	78ce85e2	994aa3cf	34444ca6	840d299a	0b2bd34f
7	1657f442	ac2f60c0	9ab5f3bc	132dbf13	d580216f	02236774	c361a433	56f9675b
8	adf2d34f	5dfa18b3	ac749d78	c4c3d69a	ec8eb4ec	79ba8f6a	7a965994	78752db8
9	f0ea675b	ef0a06d1	58958bf1	f115b85e	541bf315	96f17e15	961f6321	2d0f02fc

LFSR-state after completion of the initialisation mode:

i	S_{0+i}	S_{1+i}	S_{2+i}	S_{3+i}	S_{4+i}	S_{5+i}	S_{6+i}	S_{7+i}
0	1f808882	4fc08639	246a9891	1f77c16f	50f0e1c9	723e8fac	24334616	4471b734
8	7dba1992	25180096	4637117c	2a92aac8	7da8d7b5	58f45afe	42814800	56d7e7d8

FSM-state after completion of the initialisation mode:

R_1 = 52761a25

R_2 = 38f712e1

Keystream mode

t	X_0	X_1	X_2	X_3	R_1	R_2	z	S_{15}
0	adaf4800	aac84a30	b734e47d	98913f01	1c5787fc	fbc03f42	a0415a07	161e714b
1	2c3ce7d8	d7b58c6e	19924866	c16f9f81	43a6b863	cdcfe3e8	ed4400e7	2cc64c79
2	598c714b	5afe5525	009688e3	e1c948d5	ebd7843d	eb24cf11	0633e5c5	5d15229e

4 CONFIDENTIALITY ALGORITHM *128-EEA3*

4.1 Overview

The test data sets presented here are for the ***128-EEA3*** confidentiality algorithm.

4.2 Format

Each test set shows the various inputs to the algorithm including the plain text data stream to be encrypted/decrypted. (The length field is in decimal).

The fields are:

Key	= CK[0] || … || CK[127]
Count	= COUNT-C[0] || … || COUNT-C[31]
Bearer	= BEARER[0] || … || BEARER[4]
Direction	= DIRECTION[0]
Length	= Length of data in decimal
Plaintext	= PT[0] || PT[1] || … || PT[Length-1]

This is followed by the modified input data, i.e. it is the bitwise exclusive-or of the corresponding keystream and the input data to the algorithm.

Ciphertext	= CT[0] || CT[1] || … || CT[Length-1]

As this is a bitwise stream cipher it is purely a matter of context whether the operation is regarded as “encryption” or “decryption”. For the purposes of this document we regard the input as Plaintext and the output as Ciphertext.

The first test set is shown twice, once in binary format, once in hexadecimal format. This is to explicitly show the relationship between the binary data and the hexadecimal presentation.

The remainder of the test sets are presented in hexadecimal format only.

4.3 Test Set 1

```
Key      = (hex) 17 3d 14 ba 50 03 73 1d 7a 60 04 94 70 f0 0a 29

Key      = (bin) 00010111 00111101 00010100 10111010

                 01010000 00000011 01110011 00011101

                 01111010 01100000 00000100 10010100

                 01110000 11110000 00001010 00101001

Count    = (hex) 66035492

Count    = (bin) 01100110 00000011 01010100 10010010

Bearer   = (hex) f

Bearer   = (bin) 1111

Direction = (hex) 0

Direction = (bin) 0
```

```
Length   = 193 bits

Plaintext:

     (hex) 6cf65340 735552ab 0c9752fa 6f9025fe 0bd675d9 005875b2 00000000

     (bin) 01101100 11110110 01010011 01000000

           01110011 01010101 01010010 10101011

           00001100 10010111 01010010 11111010

           01101111 10010000 00100101 11111110

           00001011 11010110 01110101 11011001

           00000000 01011000 01110101 10110010

           0

Ciphertext:

     (hex) a6c85fc6 6afb8533 aafc2518 dfe78494 0ee1e4b0 30238cc8 00000000

     (bin) 10100110 11001000 01011111 11000110

           01101010 11111011 10000101 00110011

           10101010 11111100 00100101 00011000

           11011111 11100111 10000100 10010100

           00001110 11100001 11100100 10110000

           00110000 00100011 10001100 11001000

           0
```

4.4 Test Set 2

```
Key      = e5 bd 3e a0 eb 55 ad e8 66 c6 ac 58 bd 54 30 2a

Count    = 56823

Bearer   = 18

Direction = 1

Length   = 800 bits

Plaintext:

     14a8ef69 3d678507 bbe7270a 7f67ff50 06c3525b 9807e467 c4e56000 ba338f5d
     42955903 67518222 46c80d3b 38f07f4b e2d8ff58 05f51322 29bde93b bbdcaf38
     2bf1ee97 2fbf9977 bada8945 847a2a6c 9ad34a66 7554e04d 1f7fa2c3 3241bd8f
     01ba220d

Ciphertext:

     131d43e0 dea1be5c 5a1bfd97 1d852cbf 712d7b4f 57961fea 3208afa8 bca433f4
     56ad09c7 417e58bc 69cf8866 d1353f74 865e8078 1d202dfb 3ecff7fc bc3b190f
```

e82a204e d0e350fc 0f6f2613 b2f2bca6 df5a473a 57a4a00d 985ebad8 80d6f238
64a07b01

4.5 Test Set 3

Key = d4 55 2a 8f d6 e6 1c c8 1a 20 09 14 1a 29 c1 0b

Count = 76452ec1

Bearer = 2

Direction = 1

Length = 1570 bits

Plaintext:

38f07f4b e2d8ff58 05f51322 29bde93b bbdcaf38 2bf1ee97 2fbf9977 bada8945
847a2a6c 9ad34a66 7554e04d 1f7fa2c3 3241bd8f 01ba220d 3ca4ec41 e074595f
54ae2b45 4fd97143 20436019 65cca85c 2417ed6c bec3bada 84fc8a57 9aea7837
b0271177 242a64dc 0a9de71a 8edee86c a3d47d03 3d6bf539 804eca86 c584a905
2de46ad3 fced6554 3bd90207 372b27af b79234f5 ff43ea87 0820e2c2 b78a8aae
61cce52a 0515e348 d196664a 3456b182 a07c406e 4a207912 71cfeda1 65d535ec
5ea2d4df 40000000

Ciphertext:

8383b022 9fcc0b9d 2295ec41 c977e9c2 bb72e220 378141f9 c8318f3a 270dfbcd
ee6411c2 b3044f17 6dc6e00f 8960f97a facd131a d6a3b49b 16b7babc f2a509eb
b16a75dc ab14ff27 5dbeeea1 a2b155f9 d52c2645 2d0187c3 10a4ee55 beaa78ab
4024615b a9f5d5ad c7728f73 560671f0 13e5e550 085d3291 df7d5fec edded559
641b6c2f 585233bc 71e9602b d2305855 bbd25ffa 7f17ecbc 042daae3 8c1f57ad
8e8ebd37 346f71be fdbb7432 e0e0bb2c fc09bcd9 6570cb0c 0c39df5e 29294e82
703a637f 80000000

4.6 Test Set 4

Key = db 84 b4 fb cc da 56 3b 66 22 7b fe 45 6f 0f 77

Count = e4850fe1

Bearer = 10

Direction = 1

Length = 2798 bits

Plaintext:

e539f3b8 973240da 03f2b8aa 05ee0a00 dbafc0e1 82055dfe 3d7383d9 2cef40e9
2928605d 52d05f4f 9018a1f1 89ae3997 ce19155f b1221db8 bb0951a8 53ad852c
e16cff07 382c93a1 57de00dd b125c753 9fd85045 e4ee07e0 c43f9e9d 6f414fc4
d1c62917 813f74c0 0fc83f3e 2ed7c45b a5835264 b43e0b20 afda6b30 53bfb642
3b7fce25 479ff5f1 39dd9b5b 995558e2 a56be18d d581cd01 7c735e6f 0d0d97c4
ddc1d1da 70c6db4a 12cc9277 8e2fbbd6 f3ba52af 91c9c6b6 4e8da4f7 a2c266d0
2d001753 df089603 93c5d568 88bf49eb 5c16d9a8 0427a416 bcb597df 5bfe6f13
890a07ee 1340e647 6b0d9aa8 f822ab0f d1ab0d20 4f40b7ce 6f2e136e b67485e5
07804d50 4588ad37 ffd81656 8b2dc403 11dfb654 cdead47e 2385c343 6203dd83
6f9c64d9 7462ad5d fa63b5cf e08acb95 32866f5c a787566f ca93e6b1 693ee15c
f6f7a2d6 89d97417 98dc1c23 8e1be650 733b18fb 34ff880e 16bbd21b 47ac0000

Ciphertext:

```
4bbfa91b a25d47db 9a9f190d 962a19ab 323926b3 51fbd39e 351e05da 8b8925e3
0b1cce0d 12211010 95815cc7 cb631950 9ec0d679 40491987 e13f0aff ac332aa6
aa64626d 3e9a1917 519e0b97 b655c6a1 65e44ca9 feac0790 d2a321ad 3d86b79c
5138739f a38d887e c7def449 ce8abdd3 e7f8dc4c a9e7b733 14ad310f 9025e619
46b3a56d c649ec0d a0d63943 dff592cf 962a7efb 2c8524e3 5a2a6e78 79d62604
ef268695 fa400302 7e22e608 30775220 64bd4a5b 906b5f53 1274f235 ed506cff
0154c754 928a0ce5 476f2cb1 020a1222 d32c1455 ecaef1e3 68fb344d 1735bfbe
deb71d0a 33a2a54b 1da5a294 e679144d df11eb1a 3de8cf0c c0619179 74f35c1d
9ca0ac81 807f8fcc e6199a6c 7712da86 5021b04c e0439516 f1a526cc da9fd9ab
bd53c3a6 84f9ae1e 7ee6b11d a138ea82 6c5516b5 aadf1abb e36fa7ff f92e3a11
76064e8d 95f2e488 2b5500b9 3228b219 4a475c1a 27f63f9f fd264989 a1bc0000
```

4.7 Test Set 5

```
Key      = e1 3f ed 21 b4 6e 4e 7e c3 12 53 b2 bb 17 b3 e0

Count    = 2738cdaa

Bearer   = 1a

Direction = 0

Length   = 4019 bits

Plaintext:

8d74e20d 54894e06 d3cb13cb 3933065e 8674be62 adb1c72b 3a646965 ab63cb7b
7854dfdc 27e84929 f49c64b8 72a490b1 3f957b64 827e71f4 1fbd4269 a42c97f8
24537027 f86e9f4a d82d1df4 51690fdd 98b6d03f 3a0ebe3a 312d6b84 0ba5a182
0b2a2c97 09c090d2 45ed267c f845ae41 fa975d33 33ac3009 fd40eba9 eb5b8857
14b768b6 97138baf 21380eca 49f644d4 8689e421 5760b906 739f0d2b 3f091133
ca15d981 cbe401ba f72d05ac e05cccb2 d297f4ef 6a5f58d9 1246cfa7 7215b892
ab441d52 78452795 ccb7f5d7 9057a1c4 f77f80d4 6db2033c b79bedf8 e60551ce
10c667f6 2a97abaf abbcd677 2018df96 a282ea73 7ce2cb33 1211f60d 5354ce78
f9918d9c 206ca042 c9b62387 dd709604 a50af16d 8d35a890 6be484cf 2e74a928
99403643 53249b27 b4c9ae29 eddfc7da 6418791a 4e7baa06 60fa6451 1f2d685c
c3a5ff70 e0d2b742 92e3b8a0 cd6b04b1 c790b8ea d2703708 540dea2f c09c3da7
70f65449 e84d817a 4f551055 e19ab850 18a0028b 71a144d9 6791e9a3 57793350
4eee0060 340c69d2 74e1bf9d 805dcbcc 1a6faa97 6800b6ff 2b671dc4 63652fa8
a33ee509 74c1c21b e01eabb2 16743026 9d72ee51 1c9dde30 797c9a25 d86ce74f
5b961be5 fdfb6807 814039e7 137636bd 1d7fa9e0 9efd2007 505906a5 ac45dfde
ed7757bb ee745749 c2963335 0bee0ea6 f409df45 80160000

Ciphertext:

94eaa4aa 30a57137 ddf09b97 b25618a2 0a13e2f1 0fa5bf81 61a879cc 2ae797a6
b4cf2d9d f31debb9 905ccfec 97de605d 21c61ab8 531b7f3c 9da5f039 31f8a064
2de48211 f5f52ffe a10f392a 04766998 5da454a2 8f080961 a6c2b62d aa17f33c
d60a4971 f48d2d90 9394a55f 48117ace 43d708e6 b77d3dc4 6d8bc017 d4d1abb7
7b7428c0 42b06f2f 99d8d07c 9879d996 00127a31 985f1099 bbd7d6c1 519ede8f
5eeb4a61 0b349ac0 1ea23506 91756bd1 05c974a5 3eddb35d 1d4100b0 12e522ab
41f4c5f2 fde76b59 cb8b96d8 85cfe408 0d1328a0 d636cc0e dc05800b 76acca8f
ef672084 d1f52a8b bd8e0993 320992c7 ffbae17c 408441e0 ee883fc8 a8b05e22
f5ff7f8d 1b48c74c 468c467a 028f09fd 7ce91109 a570a2d5 c4d5f4fa 18c5dd3e
4562afe2 4ef77190 1f59af64 5898acef 088abae0 7e92d52e b2de5504 5bb1b7c4
164ef2d7 a6cac15e eb926d7e a2f08b66 e1f759f3 aee44614 725aa3c7 482b3084
4c143ff8 5b53f1e5 83c50125 7dddd096 b81268da a303f172 34c23335 41f0bb8e
190648c5 807c866d 71932286 09adb948 686f7de2 94a802cc 38f7fe52 08f5ea31
96d0167b 9bdd02f0 d2a5221c a508f893 af5c4b4b b9f4f520 fd84289b 3dbe7e61
497a7e2a 584037ea 637b6981 127174af 57b471df 4b2768fd 79c1540f b3edf2ea
22cb69be c0cf8d93 3d9c6fdd 645e8505 91cca3d6 2c0cc000
```

5 INTEGRITY ALORITHM *128-EIA3*

5.1 Overview

The test data sets presented here are for the ***128-EIA3*** integrity algorithm.

Each test set shows the various inputs to the algorithm including the plain text data stream to be 'MAC'd. The length field is in decimal.

The fields are:

Key	= IK[0] || … || IK[127]
Count	= COUNT-I[0] || … || COUNT-I[31]
Bearer	= BEARER[0] || … || BEARER[4]
Direction	= DIRECTION[0]
Length	= Length of data in decimal
Message	= MESSAGE[0] || … || MESSAGE[Length-1]

This is followed by the calculated value for MAC-I.

Output	= MAC-I[0] || … || MAC-I[31]

The first test set is shown twice, once in binary format, once in hexadecimal format. This is to explicitly show the relationship between the binary data and the hexadecimal presentation.

The remainder of the test sets are presented in hexadecimal format only.

5.2 Test Set 1

```
Key      = (hex) 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Key      = (hex) 00000000 00000000 00000000 00000000
                 00000000 00000000 00000000 00000000
                 00000000 00000000 00000000 00000000
                 00000000 00000000 00000000 00000000

Count    = (hex) 0

Count    = (bin) 0

Bearer   = (hex) 0

Bearer   = (bin) 0

Direction = (hex) 0

Direction = (bin) 0

Length   = 1 bits

Message:

     (hex) 00000000

     (bin) 0
```

```
MAC: (hex) c8a9595e

     (bin) 11001000 10101001 01011001 01011110
```

5.3 Test Set 2

```
Key      = 47 05 41 25 56 1e b2 dd a9 40 59 da 05 09 78 50

Count    = 561eb2dd

Bearer   = 14

Direction = 0

Length   =  90 bits

Message:

     00000000 00000000 00000000

MAC: 6719a088
```

5.4 Test Set 3

```
Key      = c9 e6 ce c4 60 7c 72 db 00 0a ef a8 83 85 ab 0a

Count    = a94059da

Bearer   = a

Direction = 1

Length   =  577 bits

Message:

     983b41d4 7d780c9e 1ad11d7e b70391b1 de0b35da 2dc62f83 e7b78d63 06ca0ea0
     7e941b7b e91348f9 fcb170e2 217fecd9 7f9f68ad b16e5d7d 21e569d2 80ed775c
     ebde3f40 93c53881 00000000

MAC: fae8ff0b
```

5.5 Test Set 4

```
Key      = c8 a4 82 62 d0 c2 e2 ba c4 b9 6e f7 7e 80 ca 59

Count    = 05097850

Bearer   = 10

Direction = 1

Length   =  2079 bits

Message:

     b546430b f87b4f1e e834704c d6951c36 e26f108c f731788f 48dc34f1 678c0522
     1c8fa7ff 2f39f477 e7e49ef6 0a4ec2c3 de24312a 96aa26e1 cfba5756 3838b297
     f47e8510 c779fd66 54b14338 6fa639d3 1edbd6c0 6e47d159 d94362f2 6aeeedee
     0e4f49d9 bf841299 5415bfad 56ee82d1 ca7463ab f085b082 b09904d6 d990d43c
```

```
     f2e062f4 0839d932 48b1eb92 cdfed530 0bc14828 0430b6d0 caa094b6 ec8911ab
     7dc36824 b824dc0a f6682b09 35fde7b4 92a14dc2 f4364803 8da2cf79 170d2d50
     133fd494 16cb6e33 bea90b8b f4559b03 732a01ea 290e6d07 4f79bb83 c10e5800
     15cc1a85 b36b5501 046e9c4b dcae5135 690b8666 bd54b7a7 03ea7b6f 220a5469
     a568027e

MAC: 004ac4d6
```

5.6 Test Set 5

```
Key      = 6b 8b 08 ee 79 e0 b5 98 2d 6d 12 8e a9 f2 20 cb

Count    = 561eb2dd

Bearer   = 1c

Direction = 0

Length   = 5670 bits

Message:

     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: 0ca12792
```

<End of Document>