# Specification of the Stream Cipher WG-16 Based Confidentiality and Integrity Algorithms

Xinxin Fan and Guang Gong

Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, N2L 3G1, CANADA
{x5fan,ggong}@uwaterloo.ca

**Abstract.** The fourth generation of mobile telecommunications system (marketed as 4G-LTE) is being commercially deployed. Security mechanisms are crucial to protect communications of mobile users from potential malicious attacks as well as to ensure revenue for 4G-LTE network operators. The randomness properties of the keystream generated by the current cipher suites in 4G-LTE standard are difficult to analyze and some vulnerabilities in regard to the integrity algorithms have been recently discovered. To address those issues, this technical report gives a detailed specification of a bit-oriented stream cipher WG-16 as well as the corresponding confidentiality and integrity algorithms that can be employed to secure the emerging 4G-LTE networks.

**Key words:** Stream cipher, WG-16, confidentiality, integrity, 4G-LTE.

## 1 Introduction

Mobile telecommunications systems have evolved in a stepwise manner. Since the first deployments of analog mobile telephony in the 1980s, new generations of mobile networks have been appearing on the market about every ten years. The dominant second-generation (2G) system, Global System for Mobile Communications (GSM), was introduced in the early 1990s. The most successful third-generation (3G) system, Universal Mobile Telecommunications System (UMTS), was brought into use in 2002. Since 2010, the Long Term Evolution (LTE) technology has established itself as the unrivalled mobile broadband technology of the fourth generation (4G). The 4G-LTE standard aims at simplifying the architecture of the system, as it transitions from the existing UMTS circuit and packet switching combined network to an all Internet Protocol (IP) flat architecture system. By connecting the latest smartphones to 4G-LTE networks, mobile operators are able to provide subscribers with faster browsing speed and a better message, voice, and video experience.

For each evolution of telecommunication systems, new security features have been enhanced to address learning from its predecessor as well as to accommodate new types of deployment scenarios and applications. Since 4G-LTE networks have much flatter architectures, with fewer network elements, and are entirely

IP-based, the communication security issues have to be addressed in an entirely different way from GSM and 3G. The security architecture of current 4G-LTE networks is basically a re-use of UMTS Authentication and Key Agreement (A-KA) with certain extensions and enhancements to accommodate the changes that 4G-LTE networks represent. The 3GPP-TSG is actively working on revising the specifications of the 3GPP confidentiality and integrity algorithms to address any potential security vulnerabilities. Currently there are three cipher suites in 3GPP UMTS systems, including a block cipher Kasumi [1] and two stream ciphers SNOW 3G (2006, from Europe) [2] and ZUC (2010, from China) [3]. Those cipher suites have migrated into the 4G-LTE standard in which Kasumi is replaced by AES. The main issue of the current cipher suites specified in 4G-LTE standard is that the randomness of keystreams generated by those cryptographic algorithms is difficult to characterize. In addition, a number of attacks to the core stream cipher algorithms [4, 5, 12, 21] and some weaknesses of the integrity algorithms [20] have been recently discovered.

In this technical report, we describe a bit-oriented stream cipher WG-16, which is an efficient variant of the well-known WG stream cipher family [16] as submitted to the eSTREAM project. WG-16 inherits good randomness properties of the WG stream cipher family such as period, balance, ideal two-level autocorrelation, ideal tuple distribution, and exact linear complexity. Moreover, WG-16 is able to resist the most common attacks against stream ciphers including algebraic attack, correlation attack, differential attack, cube attack, distinguish attack, discrete fourier transform attack, and time-memory-data tradeoff attack. Therefore, WG-16 is a viable candidate for protecting communications in emerging 4G-LTE networks. The stream cipher WG-16 takes as inputs a 128-bit key and a 128-bit initial vector (IV) and outputs one keystream bit per clock cycle. The keystream can be used for encrypting/decrypting communications between a mobile phone and a base station in 4G-LTE networks.

The remainder of this report is organized as follows. Section 2 gives notations that will be used throughout this report. Subsequently, in Section 3 we describe the architecture of the stream cipher WG-16 in detail. The cryptanalysis of the WG-16 stream cipher is conducted in Section 4. Section 5 presents the confidentiality and integrity algorithms when using WG-16 for securing 4G-LTE networks. Finally, Section 6 concludes this report.

## 2 Preliminaries

We give some terms and notations that will be used to describe the stream cipher WG-16, its architecture, and the confidentiality and integrity algorithms as well as to characterize randomness and cryptographic properties of WG-16.

- $\mathbb{F}_2 = \{0, 1\}$, the Galois field with two elements 0 and 1.
- $p(x) = x^{16} + x^5 + x^3 + x^2 + 1$, a primitive polynomial of degree 16 over $\mathbb{F}_2$.
- $r(x) = x^{64} + x^4 + x^3 + x + 1$, a primitive polynomial of degree 64 over $\mathbb{F}_2$.

- $\mathbb{F}_{2^{16}}$, the extension field of $\mathbb{F}_2$ defined by the primitive polynomial $p(x)$ with $2^{16}$ elements. Each element in $\mathbb{F}_{2^{16}}$ is represented as a 16-bit binary vector. Let $\omega$ be a primitive element of $\mathbb{F}_{2^{16}}$ such that $p(\omega) = 0$.
- $\mathbb{F}_{2^{64}}$, the extension field of $\mathbb{F}_2$ defined by the primitive polynomial $f(x)$ with $2^{64}$ elements. Each element in $\mathbb{F}_{2^{64}}$ is represented as a 64-bit binary vector.
- $\mathrm{Tr}(x) = x + x^2 + x^{2^2} + \cdots + x^{2^{15}}$, the trace function from $\mathbb{F}_{2^{16}} \mapsto \mathbb{F}_2$.
- $l(x) = x^{32} + x^{31} + x^{22} + x^9 + \omega^{11}$, the feedback polynomial of LFSR (which is also a primitive polynomial over $\mathbb{F}_{2^{16}}$).
- $q(x) = x + x^{2^{11}+1} + x^{2^{11}+2^6+1} + x^{2^6-2^{11}+1} + x^{2^{11}+2^6-1}$, a permutation polynomial over $\mathbb{F}_{2^{16}}$.
- $\mathsf{WGP\text{-}16}(x^d) = q(x^d + 1) + 1$, the WG-16 permutation with decimation $d$ from $\mathbb{F}_{2^{16}} \mapsto \mathbb{F}_{2^{16}}$, where $d$ is coprime to $2^{16} - 1$.
- $\mathsf{WGT\text{-}16}(x^d) = \mathrm{Tr}(\mathsf{WGP\text{-}16}(x^d))$, the WG-16 transformation with decimation $d$ from $\mathbb{F}_{2^{16}} \to \mathbb{F}_2$, where $d$ is coprime to $2^{16} - 1$.
- Polynomial basis (PB) of $\mathbb{F}_{2^{16}}$: A polynomial basis of $\mathbb{F}_{2^{16}}$ over $\mathbb{F}_2$ is a basis of the form $\{1, \omega, \omega^2, \cdots, \omega^{15}\}$.
- Normal basis (NB) of $\mathbb{F}_{2^8}$: A normal basis of $\mathbb{F}_{2^{16}}$ over $\mathbb{F}_2$ is a basis of the form $\{\theta, \theta^2, \cdots, \theta^{2^{15}}\}$, where $\theta = \omega^{11}$ (i.e., a normal element) is used in this work.
- Autocorrelation: The autocorrelation of a binary sequence with period $T$ is defined as the difference between the agreements and disagreements when the symbol 0 maps to 1 and 1 maps to $-1$. If all the out-of-phase autocorrelation is equal to $-1$, then the sequence is said to have *ideal two-level autocorrelation*.
- Linear span (LS): The linear span or linear complexity of a binary sequence is defined as the length of the smallest linear feedback shift register (LFSR) which generates the entire binary sequence.
- Nonlinearity: The nonlinearity of a function $f$ is defined as the minimum distance from $f$ to any affine function with the same number of variables.
- Algebraic immunity (AI): The algebraic immunity of a function $f$ is defined as the minimum degree of an annihilator Boolean function $g$ such that $g$ is equivalent to either $f$ or the complement of $f$ (i.e., $f \cdot g = 0$ or $(f+1) \cdot g = 0$). In the ideal case, the algebraic immunity of a function $f$ is equal to the degree of $f$, thus making it immune to algebraic attacks.
- $\oplus$, the bitwise addition operator (i.e., XOR).
- $\otimes$, the multiplication operator over $\mathbb{F}_{2^{16}}$.

## 3 The Specification of the Stream Cipher WG-16

WG-16 is an efficient variant of the well-known Welch-Gong (WG) stream cipher family with 128-bit secret key and 128-bit initial vector (IV). The stream cipher WG-16 consists of a 32-stage LFSR with the feedback polynomial $l(x)$ followed by a WG-16 transformation module with decimation $d = 1057$. Therefore, it can be regarded as a nonlinear filter generator over finite field $\mathbb{F}_{2^{16}}$. WG-16 operates in two phases, including an initialization phase and a running phase.

### 3.1 Initialization Phase

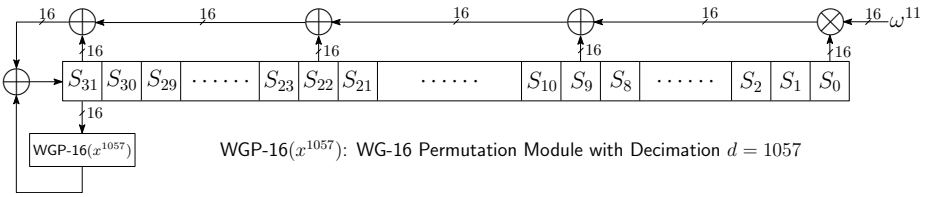


**Fig. 1.** The Initialization Phase of the Stream Cipher WG-16

The key/IV initialization phase of the stream cipher WG-16 is shown in Fig. 1. Let the 128-bit secret key be $K = (K_{127}, \ldots, K_0)_2$, the 128-bit IV be $IV = (IV_{127}, \ldots, IV_0)_2$, and the internal state of the LFSR be $S_0, \ldots, S_{31} \in \mathbb{F}_{2^{16}}$, where $S_i = (S_{i,15}, \ldots, S_{i,0})_2$ for $i = 0, \ldots, 31$. The key/IV initialization process is conducted below:

$$S_i = \begin{cases} (K_{8i+7}, \ldots, K_{8i}, IV_{8i+7}, \ldots, IV_{8i})_2 & \text{for } i = 0, 1, \ldots, 15, \\ S_{i-16} & \text{for } i = 16, 17, \ldots, 31. \end{cases}$$

Once the LFSR is loaded with the key/IV, the apparatus runs for 64 clock cycles. During each clock cycle, the 16-bit internal state $S_{31}$ is sent to the nonlinear WG-16 permutation with decimation $d = 1057$ (i.e., the WGP-16($x^{1057}$) module) and the output is used as the feedback to update the internal state of the LFSR. The LFSR state update procedure follows the recursive relation:

$$S_{k+32} = (\omega^{11} \otimes S_k) \oplus S_{k+9} \oplus S_{k+22} \oplus S_{k+31} \oplus \text{WGP-16}(S_{k+31}^{1057}), \quad 0 \le k < 64.$$

After the key/IV initialization phase, the stream cipher WG-16 goes into the running phase and 1-bit keystream is generated for each clock cycle.

### 3.2 Running Phase

The running phase of the stream cipher WG-16 is illustrated in Fig. 2. During the running phase, the 16-bit internal state $S_{31}$ is sent to the nonlinear WG-16 transformation with decimation $d = 1057$ (i.e., the WGT-16($x^{1057}$) module) and the output is 1-bit keystream. Note that the only feedback in the running phase is within the LFSR and the recursive relation for updating the internal state of LFSR is given below:

$$S_{k+32} = (\omega^{11} \otimes S_k) \oplus S_{k+9} \oplus S_{k+22} \oplus S_{k+31}, \quad k \ge 64.$$

The WG-16 transformation module WGT-16($x^{1057}$) comprises of two sub-modules: a WG-16 permutation module WGP-16($x^{1057}$) followed by a trace computation module $\mathrm{Tr}(\cdot)$. While the WGP-16($x^{1057}$) module permutes elements over $\mathbb{F}_{2^{16}}$, the $\mathrm{Tr}(\cdot)$ module compresses a 16-bit input to a 1-bit keystream.

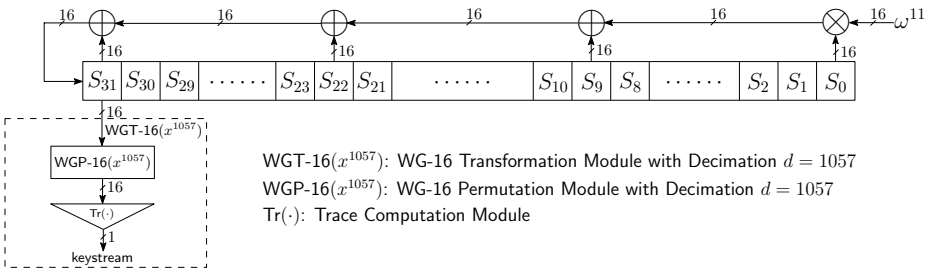


**Fig. 2.** The Running Phase of the Stream Cipher WG-16

### 3.3 Randomness Properties of the WG-16 Keystream

The keystream generated by the stream cipher WG-16 has the following desired randomness properties [7]:

1. The keystream has a period of $2^{512} - 1$.
2. The keystream is balanced, i.e., the number of 0's is only one less than the number of 1's in one period of the keystream.
3. The keystream is an ideal two-level autocorrelation sequence.
4. The keystream has an ideal $t$-tuple ($1 \leq t \leq 32$) distribution, i.e., every possible output $t$-tuple is equally likely to occur in one period of the keystream.
5. The linear span of the keystream can be determined exactly, which is $2^{79.046}$.

## 4 Cryptanalysis of the Stream Cipher WG-16

This section gives an extensive cryptanalysis of the stream cipher WG-16.

### 4.1 Algebraic Attack

The algebraic attack was first proposed by Courtois and Meier [10] to attack LFSR based filtering sequence generators, the goal of which is to form a lower degree multivariate equation by multiplying the filtering function by a low-degree multivariate polynomial. The algebraic attack gives an overdefined system of nonlinear equations for sufficiently many keystreams, which can be solved to recover the internal state of LFSR. The algebraic immunity of the WGT-16($x^{1057}$) is equal to 8. According to the algebraic attack, the time complexity and the data complexity for recovering the internal state of the LFSR are about $\frac{7}{64} \cdot \binom{512}{8}^{\log_2 7} = 2^{155.764}$ and $\binom{512}{8} = 2^{56.622}$, respectively. For applying the fast algebraic attacks [9] to the stream cipher WG-16, one needs to find two multivariate polynomials $g$ and $h$ of degree $e$ and $d$ ($e < d$) such that $f \cdot g = h$, respectively. For the WGT-16($x^{1057}$) and $e = 1$, there does not exist a multivariate polynomial $h$ in 16 variables with degree less than 15. Hence, launching the fast algebraic attack requires to obtain more keystream bits with a higher complexity. In 4G-LTE networks, it is difficult for an attacker to obtain about $2^{56.622}$ keystream bits from one communication

session. Even if the attacker can get those many bits for a fixed key and IV, he must perform the operations with a time complexity $2^{155.764}$, which completely defeats this attack in 4G-LTE networks.

### 4.2 Correlation Attack

In the correlation attack, an attacker aims to find the correlation either between a keystream and an output sequence of LFSR or among keystreams [8, 15, 19]. Thanks to the ideal two-level autocorrelation property of the keystream generated by WG-16, the correlation attack among keystreams is automatically thwarted. We now consider the fast correlation attack in which the keystream generated by a stream cipher is considered as a distorted version of the LFSR output. For launching a fast correlation attack, the linear approximation of WGT-16($x^{1057}$) can be used to derive a generator matrix of a linear code that can be decoded by a maximum likelihood decoding (MLD) algorithm. Letting $f(x)$ be a linear function in 16 variables, we have $\Pr(\text{WGT-16}(x^{1057})(x) = f(x)) = \frac{(2^{16}-32160)}{2^{16}} = 0.509277$. Applying the results of [8] for $t = 3$, the amount of keystream (denoted by $N$) required for the attack to be successful is given by $N \approx (k \cdot 12 \cdot \ln 2)^{\frac{1}{3}} \cdot \epsilon^{-2} \cdot 2^{\frac{160-k}{3}}$ and the decoding complexity is given by $C_{dec} = 2^k \cdot k \cdot \frac{2 \ln 2}{(2\epsilon)^6}$, where $\epsilon = (\Pr(\text{WGT-16}(x^{1057}) = f(x)) - 0.5) = 0.009277$ and $k$ is the number of LFSR internal state bits recovered. If we choose a small value of $k$ (e.g., $k = 7$), the number of bits required to launch the attack is about $2^{66.46}$, which is not possible in 4G-LTE networks. Similarly, if we choose a large value of $k$ (e.g., $k = 80$), the number of bits required to mount the attack is about $2^{43.3}$. However, the decoding complexity of the attack is approximately $2^{121.31}$, which is worse than the exhaustive search. Therefore, the stream cipher WG-16 is also secure against the fast correlation attack.

### 4.3 Differential Attack

The initialization phase in the first design of the WG stream cipher was vulnerable to the chosen IV attack [22], where an attacker can distinguish several output bits by building a distinguisher based on the differential cryptanalysis. This weakness has been fixed in the later design by placing the WG permutation module at the last position of the LFSR [16]. Note that the WG-16 permutation WGP-16($x^{1057}$) is applied for 64 times during the initialization phase. As a result, all internal state bits of LFSR will be affected after 64 clock cycles and it would be quite difficult for an adversary to distinguish the output keystream bits as the differentials become more complex and contain most key/IV bits. Therefore, WG-16 is secure against the differential attack.

### 4.4 Cube Attack

Cube attack [11] is a generic key-recovery attack that can be applied to any cryptosystem, provided that the attacker can obtain a bit of information that

can be represented by a low-degree decomposition multivariate polynomial in Algebraic Normal Form (ANF) of the secret and public variables of the target cryptosystem. In the WG-16 stream cipher, after 64 rounds of key/IV initialization, the degree of the output polynomial can be very high. Hence, it would be difficult for an attacker to collect low-degree relations among the secret key bits.

### 4.5 Distinguishing Attack

A distinguishing attack has been recently proposed against the stream cipher WG-7 [17]. Due to a small number of tap positions in the LFSR of the WG-7, the characteristic polynomial of the LFSR allows an attacker to build a distinguisher for distinguishing a keystream generated by WG-7 from a truly random keystream. For the WG-16 cipher, the characteristic polynomial of the LFSR consists of 4 tap positions and a similar distinguisher as in [17] can be built as

$$F(S_i, S_{i+9}, S_{i+22}, S_{i+31}) = \mathsf{WGT\text{-}16}((\omega^{11} \otimes S_i) \oplus S_{i+9} \oplus S_{i+22} \oplus S_{i+31}) \oplus \mathsf{WGT\text{-}16}(S_i) \oplus \mathsf{WGT\text{-}16}(S_{i+9}) \oplus \mathsf{WGT\text{-}16}(S_{i+22}) \oplus \mathsf{WGT\text{-}16}(S_{i+31}),$$

which is a Boolean function in 64 variables. For the distinguisher $F$, the probability $\Pr(F(x) = 0) = \frac{1}{2} \pm \epsilon$, where $x = (x_0, ..., x_{15}), x_i \in \mathbb{F}_{2^{16}}$. Note that the value of $\epsilon$ will be quite small due to a huge number of variables in the distinguisher, which requires an attacker to obtain more keystream bits for distinguishing the keystream. However, the computation of the exact value of $\epsilon$ is infeasible in this case because the number of possible values of $x$ is $2^{64}$. Therefore, the stream cipher WG-16 is resistant to the distinguishing attack. In [13], Gong *et al.* has extended this type of distinguishing attacks to the case in which a distinguisher can be built using a linear relation of a remote term of the LFSR, say $S_\tau$ for not large $\tau$, and the sequences addressed in a subset of tap positions of the LFSR, denoted by $I = \{i_1, \cdots, i_t\} \subset \{0, 1, \cdots, 31\}$. In other words, a distinguisher could be built using the linear relation $S_\tau = S_{i_1} + \cdots + S_{i_t}$. Since this property is controlled by the characteristic polynomial of the LFSR, it can be easily teared done by a proper selection of the characteristic polynomial of the LFSR. For our selection of the characteristic polynomial $l(x)$, there is no remote term $S_\tau$ for $32 \leq \tau \leq 2^{40}$ for which the size of set $I$ is less than 5. Thus the WG-16 stream cipher is also resistant to the general distinguishing attack.

### 4.6 Discrete Fourier Transform Attack

The Discrete Fourier Transform (DFT) attack is a new type of attack to recover the internal state of a filtering generator, which was first proposed by Rønjom and Helleseth in [18] and extended to attacking filtering generators over $\mathbb{F}_{2^n}$ by Gong *et al.* in [14]. In a DFT attack, an adversary is able to recover the internal state of a filtering generator by exploiting $D$ keystream bits with the online complexity $O(D)$, where $D$ is the linear complexity of the keystream, after a pre-computation of complexity $O(D(\log_2 D)^3)$. For launching the DFT attack against the WG-16 stream cipher, an attacker needs to obtain $2^{79.046}$ (i.e., the

linear complexity) consecutive keystream bits. Hence, the online complexity of this attack for recovering the internal state is $2^{79.049}$, after an offline computation with complexity $2^{97.96}$. For typical communication sessions in 4G-LTE networks, an attacker can never obtain $2^{79.046}$ consecutive keystream bits, thereby making the DFT attack infeasible.

### 4.7 Time-Memory-Data Tradeoff Attack

The Time-Memory-Data (TMD) tradeoff attack [6] is a generic cryptanalytic attack that is applicable to any stream cipher, especially those with low sampling resistance. The complexity of the TMD tradeoff attack is $O(2^{\frac{n}{2}})$, where $n$ is the size of the internal state. For the WG-16 stream cipher, the size of the internal state is 512-bit and thus the complexity of launching a TMD attack is expected to be $2^{256}$. Moreover, the sampling resistance of the WG-16 stream cipher is high due to the usage of the WGT-16($x^{1057}$) as the filtering function. Therefore, WG-16 is resistant to the TMD attack.

## 5 Confidentiality and Integrity Algorithms

In this section, we describe the confidentiality and integrity algorithms WGEA-128 and WGIA-128 built from the core stream cipher WG-16, respectively.

### 5.1 Confidentiality Algorithm WGEA-128

The confidentiality algorithm WGEA-128 is a stream cipher that is used to encrypt/decrypt data under a confidentiality key. The length of data can be between 1 and $2^{512}$ bits. The inputs and the output of the algorithm are given in Tables 1 and 2, respectively.

**Table 1.** The Inputs of WGEA-128

| Parameter | Size(bits) | Remark |
|---|---|---|
| COUNT | 32 | The counter ($\mathsf{COUNT}_{31}, \ldots, \mathsf{COUNT}_0$) |
| BEARER | 5 | The bearer identity ($\mathsf{BEARER}_4, \ldots, \mathsf{BEARER}_0$) |
| DIRECTION | 1 | The direction of transmission $\mathsf{DIRECTION}_0$ |
| CK | 128 | The confidentiality key ($\mathsf{CK}_{127}, \ldots, \mathsf{CK}_0$) |
| LENGTH | $[1, 2^{512}]$ | The length of the input message |
| M | LENGTH | The input bit stream ($\mathsf{M}_{\mathsf{LENGTH}-1}, \ldots, \mathsf{M}_0$) |

**Table 2.** The Output of WGEA-128

| Parameter | Size(bits) | Remark |
|---|---|---|
| C | LENGTH | The output bit stream $(\mathsf{C}_{\mathsf{LENGTH}-1}, \ldots, \mathsf{C}_0)$ |

**Initialization.** Here we define how WG-16's parameters, the initial key $K$ and the initial vector $IV$, are initialized with the confidentiality key CK and initialization variables before the generation of keystream. Using the notations in Table 1, we set the initial key $K$ and the initial vector $IV$ as follows:

$$K = (\mathsf{CK}_{127}, \ldots, \mathsf{CK}_0)$$
$$IV = (\mathsf{COUNT}_{31}, \ldots, \mathsf{COUNT}_0, \mathsf{BEARER}_4, \ldots, \mathsf{BEARER}_0, \mathsf{DIRECTION}_0, \underbrace{0, \ldots, 0}_{26},$$
$$\mathsf{COUNT}_{31}, \ldots, \mathsf{COUNT}_0, \mathsf{BEARER}_4, \ldots, \mathsf{BEARER}_0, \mathsf{DIRECTION}_0, \underbrace{0, \ldots, 0}_{26})$$

WG-16 is initialized as described in Section 3.1.

**Keystream Generation.** WG-16 is run as described in Section 3.2 to produce the sequence of keystream bits $z_0, \ldots, z_{\mathsf{LENGTH}-1}$. The bit produced first is $z_0$, the next bit $z_1$ and so on. We denote the generated keystream by $z = (z_{\mathsf{LENGTH}-1}, \ldots, z_0)$.

**Encryption/Decryption.** Encryption/decryption operations are identical operations and are performed by the exclusive-OR of the input message M with the generated key stream $z$:

$$\mathsf{C}_i = \mathsf{M}_i \oplus z_i, \quad i = 0, 1, 2, \ldots, \mathsf{LENGTH} - 1.$$

### 5.2 Integrity Algorithm WGIA-128

The integrity algorithm WGIA-128 is a message authentication code (MAC) function that is used to compute the MAC of an input message using an integrity key IK. The message may be between 1 and $2^{32}$ bits in length. The inputs and the output of the algorithm are given in Tables 3 and 4, respectively.

**Initialization.** Here we define how WG-16's parameters, the initial key $K$ and the initial vector $IV$, are initialized with the integrity key IK and initialization variables before the generation of keystream. Using the notations in Table 3, we set the initial key $K$ and the initial vector $IV$ as follows:

$$K = (\mathsf{IK}_{127}, \ldots, \mathsf{IK}_0)$$
$$IV = (\mathsf{COUNT}_{31}, \ldots, \mathsf{COUNT}_0, \mathsf{BEARER}_4, \ldots, \mathsf{BEARER}_0, \underbrace{0, \ldots, 0}_{27},$$
$$\mathsf{COUNT}_{31} \oplus \mathsf{DIRECTION}_0, \ldots, \mathsf{COUNT}_0, \mathsf{BEARER}_4, \ldots, \mathsf{BEARER}_0,$$
$$\underbrace{0, \ldots, 0}_{11}, \mathsf{DIRECTION}_0, \underbrace{0, \ldots, 0}_{15})$$

**Table 3.** The Inputs of WGIA-128

| Parameter | Size(bits) | Remark |
|---|---|---|
| COUNT | 32 | The counter ($\mathsf{COUNT}_{31}, \ldots, \mathsf{COUNT}_0$) |
| BEARER | 5 | The bearer identity ($\mathsf{BEARER}_4, \ldots, \mathsf{BEARER}_0$) |
| DIRECTION | 1 | The direction of transmission $\mathsf{DIRECTION}_0$ |
| IK | 128 | The integrity key ($\mathsf{IK}_{127}, \ldots, \mathsf{IK}_0$) |
| LENGTH | 64 | The length of the input message |
| M | LENGTH | The input bit stream ($\mathsf{M}_{\mathsf{LENGTH}-1}, \ldots, \mathsf{M}_0$) |

**Table 4.** The Output of WGIA-128

| Parameter | Size(bits) | Remark |
|---|---|---|
| MAC | 32 | The output bit stream ($\mathsf{MAC}_{31}, \ldots, \mathsf{MAC}_0$) |

WG-16 is initialized as described in Section 3.1.

**Keystream Generation.** WG-16 is run as described in Section 3.2 to generate 160 keystream bits $z_0, \ldots, z_{159}$. The bit produced first is $z_0$, the next bit $z_1$ and so on. We denote the generated 160-bit keystream by $z = (z_{159}, \ldots, z_0)$.

**MAC Generation.** Let $L = \lceil \mathsf{LENGTH}/64 \rceil + 1$. We first split the generated 160-bit keystream into three blocks $P, Q$ and $OTP$ (i.e., $z = P \| Q \| OTP$), where $P = (z_{159}, \ldots, z_{96})$ and $Q = (z_{95}, \ldots, z_{32})$ are 64-bit blocks and $QTP = (z_{31}, \ldots, z_0)$ is a 32-bit block. We set $B_i = (M_{64i+63}, \ldots, M_{64i})$ for $0 \le i \le L-3$, $B_{L-2} = (M_{\mathsf{LENGTH}-1}, \ldots, M_{64(L-2)}, 0, \ldots, 0)$, and $B_{L-1} = (\mathsf{LENGTH}_{63}, \ldots, \mathsf{LENGTH}_0)$. The MAC is computed as follows:

1. Set the 64-bit value $T_0 = 0$;
2. Compute $T_{i+1} = T_i P + B_i \pmod{2^{64}}$ for $0 \le i \le L-1$;
3. Compute $\mathsf{MAC} = [T_L]_{0..31} + OTP \pmod{2^{32}}$.

Note that in the above procedure the multiplication is computed over a finite field $\mathbb{F}_{2^{64}}$ and the addition is calculated over a ring $\mathbb{Z}_{2^{64}}$. Since the operations for generating a MAC are no longer linear, the forgery attack proposed in [20] is not applicable.

## 6 Conclusion

In this report, we present a bit-oriented stream cipher WG-16 targeted for emerging 4G-LTE networks, which inherits all the good randomness and cryptographic properties of the well-known WG stream cipher family. A detailed cryptanalysis shows that the stream cipher WG-16 is resistant to the most common attacks against stream ciphers. The confidentiality and integrity algorithms based on the core WG-16 stream cipher have been proposed to protect communications

in 4G-LTE networks. The new integrity algorithm is able to thwart a recently proposed forgery attack against the current integrity protection in current 4G-LTE standard. Therefore, the stream cipher WG-16 based cipher suite is a competitive candidate for securing 4G-LTE networks.

## References


1. The 3rd Generation Partnership Project (3GPP), "TS 35.202: Specification of the 3GPP Confidentiality and Integrity Algorithms; Document 2: Kasumi specification (V10.0.0)", available at `http://www.etsi.org/deliver/etsi_ts/135200_135299/135202/10.00.00_60/ts_135202v100000p.pdf`, April 2011.
2. The 3rd Generation Partnership Project (3GPP), "TS 35.216: Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2; Document 2: SNOW 3G specification (V10.0.0)", available at `http://www.etsi.org/deliver/etsi_ts/135200_135299/135216/10.00.00_60/ts_135216v100000p.pdf`, April 2011.
3. The 3rd Generation Partnership Project (3GPP), "TS 35.222: Specification of the 3GPP Confidentiality and Integrity Algorithms EEA3 & EIA3; Document 2: ZUC specification (V11.0.1)", available at `http://www.etsi.org/deliver/etsi_ts/135200_135299/135222/11.00.01_60/ts_135222v110001p.pdf`, May 2012.
4. A. Biryukov, D. Priemuth-Schmid, and B. Zhang, "Multiset Collision Attacks on Reduced-Round SNOW 3G and SNOW 3G$^{\oplus}$", *The 8th International Conference on Applied Cryptography and Network Security - ACNS 2010*, LNCS 6123, J. Zhou and M. Yung (eds.), Berlin, Germany: Springer-Verlag, pp. 139-153, 2010.
5. A. Biryukov, D. Priemuth-Schmid, and B. Zhang, "Differential Resynchronization Attacks on Reduced Round SNOW 3G$^{\oplus}$", *The 7th International Joint Conference on E-Business and Telecommunications - ICETE 2010*, CCIS 222, M. S. Obaidat, G. A. Tsihrintzis, and J. Filipe (eds.), Berlin, Germany: Springer-Verlag, pp. 147-157, 2012.
6. A. Biryukov and A. Shamir, "Cryptanalytic Time/Memory/Data Tradeoffs for Stream Ciphers", *Advances in Cryptology - ASIACRYPT 2000*, LNCS 1976, T. Okamoto (Ed.), Berlin, Germany: Springer-Verlag, pp. 1-13, 2000.
7. L. Chen and G. Gong, *Communication System Security*, Boca Raton, Florida, USA: Chapman & Hall/CRC, 2012.
8. V. V. Chepyzhov, T. Johansson, and B. J. M. Smeets, "A Simple Algorithm for Fast Correlation Attacks on Stream Ciphers", *The 7th International Workshop on Fast Software Encryption - FSE 2000*, LNCS 1978, B. Schneier (Ed.), Berlin, Germany: Springer-Verlag, pp. 181-195, 2001.
9. N. Courtois, "Fast Algebraic Attacks on Stream Ciphers with Linear Feedback", *Advances in Cryptology - CRYPTO 2003*, LNCS 2729, D. Boneh (Ed.), Berlin, Germany: Springer-Verlag, pp. 176-194, Springer-Verlag, 2003.
10. N. Courtois and W. Meier, "Algebraic Attacks on Stream Ciphers with Linear Feedback", *Advances in Cryptology - EUROCRYPT 2003*, LNCS 2656, E. Biham (Ed.), Berlin, Germany: Springer-Verlag, pp. 345-359, 2003.
11. I. Dinur and A. Shamir, "Cube Attacks on Tweakable Black Box Polynomials", *Advances in Cryptology - EUROCRYPT'09*, LNCS 5479, A. Joux (Ed.), Berlin, Germany: Springer-Verlag, pp. 278-299, 2009.

12. T. Fuhr, H. Gilbert, J.-R. Reinhard, and M. Videau, "Analysis of the Initial and Modified Versions of the Candidate 3GPP Integrity Algorithm 128-EIA3", *The 18th International Workshop on Selected Areas in Cryptography - SAC 2011*, LNCS 7118, A. Miri and S. Vaudenay (eds.), Berlin, Germany: Springer-Verlag, pp. 230-242, 2011.
13. G. Gong, M. D. Aagaard, and X. Fan, "Resilience to Distinguishing Attacks on WG-7 Cipher and Their Generalizations", *Centre for Applied Cryptographic Research (CACR) Technical Reports*, CACR 2012-30, available at `http://cacr.uwaterloo.ca/techreports/2012/cacr2012-30.pdf`.
14. G. Gong, S. Rønjom, T. Helleseth, and H. Hu. "Fast Discrete Fourier Spectra Attacks on Stream Ciphers", *IEEE Transactions on Information Theory*, Vol 57, No. 8, pp. 5555-5565, 2011.
15. W. Meier and O. Staffelbach, "Fast Correlation Attacks on Certain Stream Ciphers", *Journal of Cryptology*, Vol. 1, No. 3, pp. 159-176, 1989.
16. Y. Nawaz and G. Gong, "WG: A Family of Stream Ciphers with Designed Randomness Properties", *Information Science*, vol. 178, no. 7, pp. 1903-1916, 2008.
17. M. A. Orumiehchiha, J. Pieprzyk, and R. Steinfeld, "Cryptanalysis of WG-7: A Lightweight Stream Cipher", *Cryptography and Communications*, Vol. 4, Iss. 3-4, pp. 277-285, 2012.
18. S. Rønjom and T. Helleseth, "A New Attack on the Filtering Generator", *IEEE Transactions on Information Theory*, Vol 53, No. 5, pp. 1752-1758, 2007.
19. T. Siegenthaler, "Decrypting a Class of Stream Ciphers Using Ciphertext Only", *IEEE Transactions on Computers*, Vol. 34, No. 1, pp. 81-85, 1985.
20. T. Wu and G. Gong, "The Weakness of Integrity Protection for LTE", to appear in *Proceedings of the 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec'13)*, April 17-19, Budapest, Hungary, 2013.
21. H. Wu, T. Huang, P. H. Nguyen, H. Wang, and S. Ling, "Differential Attacks against Stream Cipher ZUC", *Advances in Cryptology - ASIACRYPT 2012*, LNCS 7658, X. Wang and K. Sako (eds.), Berlin, Germany: Springer-Verlag, pp. 262-277, 2012.
22. H. Wu and B. Preneel, "Chosen IV Attack on Stream Cipher WG", *ECRYPT Stream Cipher Project Report 2005/045*. Available at `http://cr.yp.to/streamciphers/wg/045.pdf`.