f8 Keystream Generator with SMS4 as Core Algorithm

Wang Yingsong
Math, Physics and Information Engineering College of Zhejiang Normal University, Jinhua, China, 321004
2007210470@zjnu.net

Chen Wei*
Math, Physics and Information Engineering College of Zhejiang Normal University, Jinhua, China, 321004 cyberella2000@163.com

Abstract
Since KASUMI algorithm is forbidden to use commercially in China, a new f8 keystream generator is presented by using SMS4 as core algorithm in this paper. Then the new f8 is implemented in software to produce 100 groups 20000-bit keystreams with different cipher keys. Later, FIPS 140-2 is applied to these keystreams to test randomness. Based on the data obtained, six figures are drawn to specify the results of FIPS 140-2 Test. From these figures, the keystreams obtained exhibit excellent statistical properties. Conclusions are made that the f8 have passed the monobit test, the poker test, the runs test and the long runs test of FIPS 140-2. Therefore, in term of randomness, the new f8 can be used safely in Chinese 3G communication system.

1. Introduction
The 3GPP Task Force has developed a detailed specification of f8 keystream generator [1], which shall only be used to protect the confidentiality between User equipment (UE) and Radio network controller (RNC) in 3G system. KASUMI is a block cipher, which has been adopted by 3GPP as a standard core algorithm of f8. However, KASUMI is forbidden to use commercially in China, according to the Regulations of Supervision and Management on Commercial Cipher (RSMCC), published by Office of State Commercial Cipher Administration (OSCCA) in 1999. Therefore, China must firstly design a new block cipher to substitute for KASUMI, performed as the requests of 3GPP [2]. Then, the new block cipher should be sent to 3GPP for security evaluation. If passed, the 3GPP would assign to a 4-bit identifier and give a detailed specification of the cipher openly. It is not likely to be done, because it would cause to conflict with relevant provisions in RSMCC, such as special control and secrecy. To solve the problem, another method [3] was presented by using a particular core algorithm in China and using the standard core algorithm when wandering abroad. In this paper, SMS4 [4], which is published by OSCCA in 2006, is proposed to use as the particular core algorithm of f8.

2. Simplified Description of f8
f8 keystream generator is a stream cipher algorithm, based on a block cipher as core algorithm in Output Feedback Mode (OFB). It requires 64-bit or 128-bit output as feedback input. When the output of the core algorithm is 64-bit, 64-bit Mode is used in f8. Similarly, 128-bit Mode is used with 128-bit output. The construction is depicted in Figure 1.

*Corresponding author. E-mail: cyberella2000@163.com.
The input parameters of f8 are the cipher key \((CK, 128 \text{ bits})\), a time dependent input \((COUNT, 32 \text{ bits})\), the bearer identity \((BEARER, 5 \text{ bits})\), the direction of transmission \((DIRECTION, 1 \text{ bit})\) and the length of the keystreams required \((LENGTH)\). Additionally, \(KM\) \((128 \text{ bits})\) is used to modify \(CK\). \(A\) is a 64-bit or 128-bit register that is used to hold intermediate values. Based on these input parameters the algorithm generates the output keystream block \((KS)\), which are used to encrypt the input plaintext and to produce the output ciphertext.

3. The New f8 Keystream Generator

In this section, a new f8 keystream generator is presented to solve Chinese 3G security problem mentioned in the introduction section. SMS4 is adopted as the core algorithm of f8. Three main reasons are responsible for this choice. Firstly, SMS4 is a block cipher that produces a 128-bit output from a 128-bit input under the control of a 128-bit key. It satisfies the requirement of 128-bit output feedback. Secondly, SMS4 is developed by China. Thus, the S-box is secure and reliable, so that we do not worry about the trapdoor problem. The last reason is that 32-round SMS4 can immune to varies of attacks until recently.

3.1. The Architecture of the New f8

SMS4 is used in OFB mode and generates the output keystreams in multiples of 128 bits. The feedback data is modified by static data held in a 128-bit register \(A\), and a 128-bit counter \(BLKCNT\). The construction is depicted in Figure 2.

![Figure 2. f8 keystream generator with SMS4 as core algorithm](image)

The \(LENGTH\), from 1 to 20000, specifies the number of bits in the output keystreams. \(BLOCKS\), an integer variable, indicates the number of successive applications of SMS4 that need to be performed. So let \(BLOCKS\) be equal to \((LENGTH/128)\) rounded up to the nearest integer. For example, if \(LENGTH=256\) then \(BLOCKS=2\); if \(LENGTH=257\) then \(BLOCKS=3\). More parameters are depicted in Table 1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Size</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUNT</td>
<td>32</td>
<td>frame dependent</td>
</tr>
<tr>
<td>BEARER</td>
<td>5</td>
<td>bearer identity</td>
</tr>
<tr>
<td>DIRECTION</td>
<td>1</td>
<td>direction of transmission</td>
</tr>
<tr>
<td>CK</td>
<td>128</td>
<td>cipher key</td>
</tr>
<tr>
<td>KM</td>
<td>128</td>
<td>to modify CK</td>
</tr>
<tr>
<td>A</td>
<td>128</td>
<td>hold intermediate values</td>
</tr>
<tr>
<td>BLKCNT</td>
<td>128</td>
<td>a 128-bit counter</td>
</tr>
<tr>
<td>KS</td>
<td>128</td>
<td>the output keystreams</td>
</tr>
</tbody>
</table>

3.2. Initialization

We define how the new f8 keystream generator is initialized with the parameter variables before the generation of keystreams. We set the 64-bit register \(A\) to \(COUNT \| BEARER \| DIRECTION \| 0...0\) \(\) (the right most 90 bits are set to 0), that is, \(A=COUNT[0]...COUNT[31]BEARER[0]...BEARER[4]DIRECTION[0]0...0\).

We set the counter \(BLKCNT\) to zero and set the key modifier \(KM\) to \(0x55555555555555555555555555555555\). The first operation result of SMS4 is then applied to the register \(A\), using a modified version of the cipher key \((CK \triangleq KM)\), as flow:

\[ A = \text{SMS4}[A]_{CK \triangleq KM} \]

By now, the f8 keystream generator can produce keystreams in multiples of 128 bits. Between 0 and 127 of the least significant bits are discarded from the last block depending on the total number of bits required by \(LENGTH\). Thus, encryption operations are performed by the exclusive-OR of the input data with the generated keystream \((KS)\). Decryption operations are identical to that.

4. Experiment

In this section, we will test the randomness of keystreams produced in the new f8 keystream generator described above.

4.1. Test Principle
We choose a method called Federal Information Processing Standard (FIPS) 140-2 [5], presented by American National Institute of Standards and Technology (NIST). FIPS 140-2 specifies four statistical tests for randomness: monobit test, poker test, runs test and long runs test. If any one fails, the generator fails the randomness test.

- **Monobit Test**: The purpose of this test is to count the number of ones in the 20000-bit keystreams. Denote this quantity by $X$. The test is passed if $9725 < X < 10275$

- **Poker Test**: This test determines whether the each 4-bit sequence appear approximately the same number of times in the the 20000-bit keystreams. Denote $f(i)$ as the number of each 4-bit value $i$, where $0 \leq i \leq 15$. Evaluate the following formula:
  
  $$X = 16 \times \left[ f(0)^2 + f(1)^2 + \ldots + f(15)^2 \right] / 5000-5000.$$  
  
  The test is passed if $2.16 < X < 46.17$

- **Runs Test**: The test is passed if the runs (length from 1 to 6) are each within the corresponding interval specified in Table 2. This must consider both zeros and ones. For the purpose of this test, lengths greater than 6 are considered to be of length 6

- **Long Runs Test**: A long run is defined to be a run of length 26 or more (of either zeros or ones). On the sample of 20000-bit keystreams, the test is passed if there are no long runs

### Table 2. Required interval of Poker Test

<table>
<thead>
<tr>
<th>Length of Run</th>
<th>Required Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2315–2685</td>
</tr>
<tr>
<td>2</td>
<td>1114–1386</td>
</tr>
<tr>
<td>3</td>
<td>527–723</td>
</tr>
<tr>
<td>4</td>
<td>240–384</td>
</tr>
<tr>
<td>5</td>
<td>103–209</td>
</tr>
<tr>
<td>6+</td>
<td>103–209</td>
</tr>
</tbody>
</table>

As for $CK$, we use 100 different values for 100 times keystream generations. Accordingly, FIPS 140-2 test is performed by using the generated keystreams in each time.

### 4.3. Result and Evaluation

Exploiting the 100 groups generated 20000-bit keystream data, we draw six figures (depicted from Figure 3 to Figure 8) to specify the results of FIPS 140-2 Test.

According to the figures, the number of ones is mainly between 9900 and 10100 and the poker value is mainly between 10 and 30. Six kind of runs are all within standard intervals. Also, the length of the longest run is mostly between 10 and 20. It is proved that the keystreams generated by the new $f8$ generator have passed the test of FIPS 140-2. The keystreams obtained exhibit excellent statistical properties. Therefore, they can be applied to the field of 3G confidentiality system.

### 5. Conclusion and Future Work

In this paper, a new $f8$ keystream generator is constructed with SMS4 as core algorithm. Then the randomness of the $f8$ is tested by using FIPS 140-2. And the test results indicate that the keystreams generated by the $f8$ are of high quality and have got through four tests of FIPS 140-2. So the new $f8$ can be applied to Chinese 3G communication confidentiality system in terms of randomness. In the future, it is possible and important to study and design the new $f8$ chip with less circuit area, higher operation speed, lower implementation cost and less power consumption. Also, it is necessary to study $f9$ with SMS4 as core algorithm in a similar way.

### 6. References


