

# СИММЕТРИЧНЫЕ БЛОЧНЫЕ ШИФРЫ И ГЕНЕРАТОРЫ СЛУЧАЙНЫХ ЧИСЕЛ

УДК 621.3.06

## ОЦЕНКА КОЛИЧЕСТВА ДОПУСТИМЫХ ВНУТРЕННИХ СОСТОЯНИЙ В ПОТОЧНОМ АЛГОРИТМЕ MISKEY

А.В.КАЗИМИРОВ, Р.В.ОЛЕЙНИКОВ

Рассматривается метод анализа количества ветвлений симметричного поточного шифра Mickey, основанный на обратных функциях. При помощи данного подхода оценивается количество возможных состояний, в которые может переходить алгоритм. Установлено, что возможны коллизии при различных вариантах вектора инициализации и ключа.

*Ключевые слова:* поточный шифр, криптоанализ, коллизия, Mickey.

### ВВЕДЕНИЕ

На сегодняшний день поточные шифры являются наиболее производительными среди всех видов криптографических алгоритмов, которые используются для обеспечения конфиденциальности. Вместе с рядом преимуществ, этот класс алгоритмов имеет и ряд недостатков, среди которых одним из основных является конечный период гаммы, что приводит к необходимости периодической смены векторов инициализации или ключей шифрования.

В 2008 году закончился европейский конкурс eSTREAM [1]. По итогам конкурса все представленные на конкурс поточные шифры разделили на две категории [1]. К первой отнесли шифры, рекомендованные к программной реализации, ко второй – к аппаратной.

Алгоритм Mickey является одним из поточных шифров, рекомендованных к применению по результатам конкурса, и относится ко второй категории [1, 2]. Преимуществом данного алгоритма является высокая производительность и высокая криптостойкость. На сегодняшний день опубликовано достаточно малое количество работ, посвящённых криптоанализу данного алгоритма [3].

В настоящей статье получено среднее значение количества допустимых внутренних состояний конечного автомата, которым является шифр Mickey, на основе использования обратных тактовых функций.

### 1. ОПИСАНИЕ АЛГОРИТМА ШИФРОВАНИЯ MISKEY

Шифр micky строится на основе линейного ( $R$ ) и нелинейного ( $S$ ) регистров.

Алгоритм принимает на вход два параметра: вектор инициализации ( $IV$ ), длина которого может составлять от 0 до 80 бит, и сеансовый ключ ( $K$ ) длиной 80 бит [2]. Существует 128-битная версия шифра [4], однако в данной работе рассматривается лишь 80 битный вариант, но аналогичные исследования можно провести и для 128 версии.

Пусть  $r_0 \dots r_{99}$  и  $s_0 \dots s_{99}$  биты регистров  $R$  и  $S$  соответственно. Тактирование линейного регистра ( $CLOCK\_R(R, IB\_R, CB\_R)$ ) происходит следующим образом:

$$- FB\_R = r_{99} \oplus IB\_R;$$

$$- \text{For } 1 \leq i \leq 99, r'_i = r_{i-1}; r'_0 = 0;$$

$$- \text{For } 0 \leq i \leq 99, \text{ if } i \in RTAPS,$$

$$- r'_i = r'_i \oplus FB\_R;$$

$$- \text{If } CB\_R = 1, \text{ For } 0 \leq i \leq 99, r'_i = r'_i \oplus r_i,$$

где  $r'_i$  – биты регистра после тактирования, а  $RTAPS$  представляет собой вектор константных бит, определённый в [2].

Тактирование нелинейного регистра ( $CLOCK\_S(S, IB\_S, CB\_S)$ ) описывается следующим алгоритмом:

$$- FB\_S = s_{99} \oplus IB\_S;$$

$$- \text{For } 1 \leq i \leq 98, \hat{s}_0 = 0; \hat{s}_{99} = s_{98};$$

$$\hat{s}_i = s_{i-1} \oplus ((s_i \oplus COMP0_i) \wedge (s_{i+1} \oplus COMP1_i)),$$

$$- \text{If } CB\_S = 0,$$

$$\text{For } 0 \leq i \leq 99, s'_i = \hat{s}_i \oplus (FB0_i \wedge FB\_S);$$

$$- \text{If } CB\_S = 1,$$

$$\text{For } 0 \leq i \leq 99, s'_i = \hat{s}_i \oplus (FB1_i \wedge FB\_S),$$

где  $s'_i$  – состояние регистра  $S$  после тактирования, значения постоянных векторов  $FB0$ ,  $FB1$ ,  $COMP0$  и  $COMP1$  представлены в [2].

Ниже представлены этапы тактирование всего шифра ( $CLOCK\_KG(R, S, MIXING, IB)$ ):

$$- CB\_R = s_{34} \oplus r_{67};$$

$$- CB\_S = s_{67} \oplus r_{33};$$

$$- \text{If } MIXING = TRUE, IB\_R = IB \oplus s_{50};$$

$$\text{Else } IB\_R = IB;$$

$$- IB\_S = IB;$$

- $CLOCK\_R(R, IB\_R, CB\_R)$ ;
- $CLOCK\_S(S, IB\_S, CB\_S)$ .

Инициализация ключа и  $IV$  происходит в четыре этапа:

а)  $R$  и  $S$  регистры инициализируются нулями;

б) For  $0 \leq i \leq |IV| - 1$ ,  
 $CLOCK\_KG(R, S, TRUE, iv_i)$  ;

в) For  $0 \leq i \leq 79$ ,  
 $CLOCK\_KG(R, S, TRUE, k_i)$  ;

г) сто тактов холостого хода при  $MIXING = TRUE$  и  $IB = 0$ ,

где  $iv_i$  и  $k_i$   $i$ -е биты вектора инициализации и ключа соответственно,  $|IV|$  – длина вектора инициализации в битах.

Таким образом, весь алгоритм можно представить в виде конечного автомата, состояниями которого будут выступать значения  $R$  и  $S$  регистров. При данной интерпретации количество возможных состояний составляет  $2^{200}$ . Далее будет показано, что на практике таких состояний гораздо меньше.

## 2. АЛГОРИТМ НАХОЖДЕНИЯ ВЕТВЛЕНИЙ

Как было показано выше, шифр Mickey имеет 3 основные функции для тактирования регистров. Идея нахождения ветвлений основана на переборе всех возможных значений  $FB\_S, IB\_S, CB\_S, FB\_R, IB\_R, CB\_R$  и анализе состояний регистра после обратного тактирования.

Предположим, что мы знаем текущее состояние регистров, тогда для нахождения предыдущих состояний необходимо вычислить функции  $CLOCK\_S^{-1}, CLOCK\_R^{-1}, CLOCK\_KG^{-1}$ , которые являются обратными к  $CLOCK\_S, CLOCK\_R, CLOCK\_KG$  соответственно. На рис. 1, 2 и 3 представлены алгоритмы этих функций.

**Входные значения:** биты  $IB\_S, CB\_S, FB\_S$  и регистр  $S$   
**Выходные значения:** TRUE, если ветвление существует, в противном случае FALSE

```

S' ← S
if FB_S = TRUE then
  if CB_S = TRUE then
    for i=0 to 99 do
      s_i ← s_i ⊕ FB1_i;
    end for
  else
    for i=0 to 99 do
      s_i ← s_i' ⊕ FB0_i;
    end for
  end if
else
  for i=0 to 99 do
    s_i ← s_i';
  end for
end if
if s_0 ≠ 0 then
  return FALSE
end if
s_99 ← FB_S ⊕ IB_S;
s_98 ← s_99;
for i=98 to 1 do
  s_{i-1} ← s_i ⊕ ((s_i ⊕ COMP0_i) ∧ (s_{i+1} ⊕ COMP1_i));
end for
return TRUE
    
```

Рис. 1. Алгоритм функции  $CLOCK\_S^{-1}$

**Входные значения:** биты  $IB\_R, CB\_R, FB\_R$  и регистр  $R$   
**Выходные значения:** TRUE, если ветвление существует, в противном случае FALSE

```

R' ← R
if CB_R = TRUE then
  r_0 ← r_0 ⊕ (FB_R ∧ RTAPS_0);
  for i=1 to 99 do
    r_i ← r_i ⊕ r_{i-1} ⊕ (FB_R ∧ RTAPS_i);
  end for
  if r_99 ≠ (FB_R ⊕ IB_R) then
    return FALSE
  end if
else
  if (r_0 ⊕ (FB_R ∧ RTAPS_0)) ≠ 0 then
    return FALSE
  end if
  for i=1 to 99 do
    r_{i-1} ← r_i ⊕ (FB_R ∧ RTAPS_i);
  end for
  r_99 ← FB_R ⊕ IB_R;
end if
return TRUE
    
```

Рис. 2. Алгоритм функции  $CLOCK\_R^{-1}$

**Входные значения:** регистры  $R$  и  $S$   
**Выходные значения:** все возможные ветвления

```

for s = 0 to 64 do
  FB_S ← (s >> 0) ∧ 1;
  IB_S ← (s >> 1) ∧ 1;
  CB_S ← (s >> 2) ∧ 1;
  FB_R ← (s >> 3) ∧ 1;
  IB_R ← (s >> 4) ∧ 1;
  CB_R ← (s >> 5) ∧ 1;
  TS ← S;
  TR ← R;
  if CLOCK_S^{-1}(TS, IB_S, CB_S, FB_S) ≠ TRUE then
    continue;
  end if
  if CLOCK_R^{-1}(TR, IB_R, CB_R, FB_R) ≠ TRUE then
    continue;
  end if
  INPUT_BIT_S ← IB_S;
  if MIXING = TRUE then
    INPUT_BIT_R ← IB_R ⊕ s_50;
  else
    INPUT_BIT_R ← IB_R;
  end if
  CONTROL_BIT_S ← s_67 ⊕ r_33;
  CONTROL_BIT_R ← s_33 ⊕ r_67;
  if CONTROL_BIT_R = CB_R and CONTROL_BIT_S = CB_S and
  INPUT_BIT_R = IB_R then
    branch ← Append(branch, [S; R; INPUT_BIT_S]);
  end if
end for
return branch
    
```

Рис. 3. Алгоритм функции  $CLOCK\_KG^{-1}$

Пусть  $X_1$  известное состояние конечного автомата, тогда под веткой понимается  $m$  состояний вида:

$$X_1 \xrightarrow{\text{такт}} X_2 \xrightarrow{\text{такт}} \dots \xrightarrow{\text{такт}} X_B,$$

где  $\text{такт}$  – для алгоритма Mickey эквивалентен  $CLOCK\_KG(CLOCK\_KG^{-1})$ .

Под циклом понимается  $n$  состояний следующего вида:

$$X_1 \xrightarrow{\text{такт}} X_2 \xrightarrow{\text{такт}} \dots \xrightarrow{\text{такт}} X_i \xrightarrow{\text{такт}} \dots \xrightarrow{\text{такт}} X_{n=i}.$$

Из описания алгоритмов обратных функций видно, что в определенных случаях вообще не возможны предыдущие состояния. Таким образом, существуют состояния, ветки или циклы, в которые конечный автомат никогда не перейдет, при любых значениях  $K$  и  $IV$ .

### 3. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЙ

Используя описанные выше алгоритмы обратных функций, можно восстановить все предыдущие состояния, из которых могло получиться текущее состояние регистров  $R$  и  $S$ . На основании этого можно посчитать невозможные состояния и количество возможных ветвлений.

В табл. 1 приведены результаты практического поиска ветвлений для шифра Mickey. Результаты были получены для начального состояния, после инициализации 80 тактов  $IV$  и 20 тактов ключа. В данном случае каждое новое состояние получалось из предыдущего, что соответствует реальному поиску начального состояния.

Таблица 1

Результаты расчёта количества ветвлений для детерминированного состояния

Ветвления	Количество	Количество, %
0	3078948	28,664
1	969	0,009
2	4723204	43,971
3	1019	0,009
4	2859210	26,618
6	159	0,001
8	78045	0,727

В табл. 2 представлены результаты для случайных состояний, т.е. каждое новое состояние генерировалось случайным образом.

Таблица 2

Результаты расчёта количества ветвлений для случайного состояния

Ветвления	Количество	Количество, %
0	3017259	28,090
1	7402	0,069
2	4710004	43,848
3	6072	0,057
4	2912621	27,115
6	17	0,000
8	88179	0,821

Как видно из таблиц 1 и 2, результаты для случайно сгенерированных значений и взятых для определённого значения  $IV$  и  $K$  практически совпадают.

Интересным является тот факт, что отсутствуют ветвления 5 и 7. Это может быть связано с внутренней структурой регистров  $S$  и  $R$ . Вероятность появления данных ветвлений приближается к 0, и не будет существенно влиять на результат.

Существует приблизительно  $2^{198}$  состояний, в которые конечный автомат не сможет перейти, из-за того, что приблизительно 28 % состояний не имеют предыдущих.

В табл. 3 приведена зависимость количества состояний конечного автомата от тактов  $CLOCK_{KG^{-1}}$ . Очевидно, что количество возможных состо-

яний приблизительно равно  $2^{n+2}$ . Данное свойство проявляется за счёт того, что ветвления «3» и выше компенсируются за счёт «0» ветвлений.

Таблица 3

Зависимость количества состояний от тактов обратной функции

Такты	Количество состояний	Количество состояний, $\log_2$
8	1200	10,23
10	4814	12,23
12	18458	14,17
14	83959	16,36
16	337064	18,36
18	1317142	20,33
20	5410768	22,37

Высокие вероятности «2» и «4» ветвлений говорят о том, что возможны различные комбинации  $IV$  и ключ шифрования, приводящие к одному и тому же состоянию.

### ВЫВОДЫ

В работе представлены результаты, которые позволяют на практике анализировать состояния шифра Mickey и выполнять поиск возможных битов ключа или  $IV$ , при знании текущего состояния.

Авторы алгоритма рекомендуют использовать шифр для генерации  $2^{40}$  бит гаммы на одном векторе инициализации. Такая длина гораздо меньше  $2^{199}$ , что позволяет говорить о том, что шифр Mickey остаётся криптографически стойким.

В дальнейших исследованиях планируется разработка аналитической модели и поиск длин невозможных веток (циклов), что позволит получить точную оценку количества допустимых состояний шифра Mickey.

### Литература

- [1] Babbage S. "The eSTREAM portfolio" [Electronic resource] / S. Babbage, C. Cannire, A. Canteaut, C. Cid, H. Gilbert, T. Johansson, M. Parker, B. Preneel, V. Rijmen, M. Robshaw. Mode of access : WWW.URL: <http://www.ecrypt.eu.org/stream/portfolio.pdf> – Last access: 2011. – Title from the screen.
- [2] Babbage S. "The stream cipher MICKEY 2.0" [Electronic resource] / S. Babbage, M. Dodd. Mode of access : WWW.URL: [www.ecrypt.eu.org/stream/p3ciphers/mickey/mickey\\_p3.pdf](http://www.ecrypt.eu.org/stream/p3ciphers/mickey/mickey_p3.pdf) – Last access: 2011. – Title from the screen.
- [3] Hong J., Kim W. "TMD-Tradeoff and State Entropy Loss Considerations of Streamcipher MICKEY" [Text] / Hong J., Kim W. Lecture Notes in Computer Science, 2005, Volume 3797/2005. pp. 169-182.
- [4] Babbage S. "The stream cipher MICKEY-128 2.0" [Electronic resource] / S. Babbage, M. Dodd. Mode of access : WWW.URL: [http://www.ecrypt.eu.org/stream/p3ciphers/mickey/mickey128\\_p3.pdf](http://www.ecrypt.eu.org/stream/p3ciphers/mickey/mickey128_p3.pdf) – Last access: 2011. – Title from the screen.

Поступила в редколлегию 1.04.2011



**Олейников Роман Васильевич**, кандидат технических наук, докторант кафедры БИТ ХНУРЭ. Область научных интересов: криптография и криптоанализ БСШ, сетевая безопасность.



**Казимиров Александр Владимирович**, магистр кафедры БИТ ХНУРЭ. Область научных интересов: криптография и криптоанализ БСШ, сетевая безопасность.

УДК 621.3.06

**Оцінка кількості допустимих внутрішніх станів у поточному алгоритмі MICKEY / О.В. Казиміров, Р.В. Олійников // Прикладна радіоелектроніка: наук.-техн. журнал. – 2011. Том 10. № 2. – С. 112–115.**

Розглядається метод аналізу кількості розгалужень симетричного потокового шифру Mickey, що заснований на зворотних функціях. За допомогою даного підходу оцінюється кількість можливих станів, у які може переходити алгоритм. Встановлено, що можливі колізії при різних варіантах вектора ініціалізації і ключа.

*Ключові слова:* потоковий шифр, криптоаналіз, колізія, Mickey.

Табл. 3. Іл. 3. Бібліогр.: 4 найм.

UDC 621.3.06

**Estimation of the number of admissible internal states in the current algorithm MICKEY / O.V.Kazimirov, R.V. Oleinikov // Applied Radio Electronics: Sci. Journ. – 2011. Vol. 10. № 2. – P. 112–115.**

The paper analyzes the number of stream cipher Mickey branches. Using this approach, which is built on the basis of inverse functions, it is possible to estimate the number of possible states the algorithm may transfer. A possibility of collisions between different initialization vectors and keys is shown.

*Keywords:* stream cipher, cryptanalysis, collision, Mickey. Tab. 3. Fig. 3. Ref.: 4 items.