ПРОБЛЕМЫ ПЕРЕДАЧИ ИНФОРМАЦИИ

Том 41 2005 Вып. 4

УДК 621.391.1:004.056

© 2005 г. **Б.Я. Рябко, В.А. Монарев, Ю.И. Шокин**

НОВЫЙ ТИП АТАК НА БЛОКОВЫЕ ШИФРЫ[1]

Предлагается и экспериментально исследуется новая атака, названная "градиентной статистической", на блоковые шифры. Показана возможность ее применения к шифрам, для которых не известно других атак, отличных от метода полного перебора ключей.

§ 1. Введение

Блоковые шифры с секретным ключом находят самое широкое применение в системах защиты передаваемой и хранимой информации, и поэтому некоторые исследователи называют их "рабочей лошадью" криптографии. Такое широкое практическое применение делает актуальными как задачи построения надежных блоковых шифров, так и поиск эффективных криптологических атак на эти шифры (т.е. методов определения секретного ключа шифра на основе экспериментов с зашифрованными сообщениями). Исследования в этих областях ведутся параллельно и зачастую одними и теми же специалистами, и как правило, изобретение новой атаки приводит к появлению шифров, к ней устойчивых. Отметим сразу, что для криптографии представляют интерес атаки, которые менее трудоемки, чем метод "прямого" перебора ключей. Например, если некоторая атака требует перебора 2^{200} ключей вместо, скажем, 2^{250}, требуемых для полного перебора, то и такая атака представляет интерес для криптологии [1]. Обзор современных блоковых шифров, методов их построения и анализа, а также различных типов атак может быть найден во многих работах, например, в [1–5], где описаны десятки современных шифров и атак. Об актуальности проблемы свидетельствует и наличие многочисленных национальных и международных программ и конкурсов (направленных на построение надежных блоковых шифров), проводимых в настоящее время в США, странах Европейского сообщества, Японии и Корее (см. обзор в [2]).

Большинство блоковых шифров могут быть описаны как функция, определенная на множестве всех двоичных слов длины $(l+k)$ и принимающая значения в множестве двоичных слов длины l, где l – длина шифруемого слова (или блока) и длина зашифрованного слова, а k – длина (секретного) ключа. В современных шифрах длина блока обычно 128 или 64 бита, а длина ключа у разных шифров (и разных режимов использования одного шифра) принимает значения от нескольких десятков до нескольких тысяч бит. Например, у шифра AES, победителя проводимого в 1999–2001 гг. в США конкурса на блоковый шифр 21-го века, длина блока l равна 128 бит, а длина ключа может принимать три значения – 128, 196 и 256 бит. У популярных шифров RC5 и RC6, предложенных Ривестом, длина блока может быть 32, 64 или 128 бит, а длина ключа в разных вариантах принимает значения

[1] Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (номер проекта 03-01-00495), INTAS (Grant 00-738) и Royal Society, UK (Grant 15995).

от 64 до нескольких тысяч бит. Стоит отметить, что RC5 и RC6 имеют очень простое описание и, по-видимому, поэтому в последние годы являются одними из самых популярных объектов криптоанализа. Краткое описание шифра RC5 дано в Приложении.

В работах [1, 5–7] описаны результаты криптоанализа шифров RC5 и RC6 и делается вывод об их высокой устойчивости ко всем описанным в литературе атакам.

Процесс шифрования во многих (если не во всех) современных блоковых шифрах разбивается на последовательность сравнительно простых этапов, часто называемых раундами. В ходе каждого нового раунда проводится шифрование данных, полученных на предыдущем этапе с так называемым ключом раунда. Все эти этапы отмечены в шифре RC5. (Здесь необходимо сделать одно терминологическое замечание: по традиции при описании RC5 используют термин "полураунд", а два полураунда объединяются в раунд [3, 7]. Поэтому при описании предлагаемой атаки будем говорить о раундах, а при анализе ее применимости к RC5 – о полураундах, надеясь, что это не приведет к путанице.)

В RC5, RC6 и многих других шифрах количество раундов является параметром, и часто криптоаналитики исследуют "стойкость" шифров как функцию числа раундов. Одна из целей такого анализа – нахождение числа раундов, гарантирующих высокую надежность шифра.

При дешифровании фактически повторяются в обратном порядке те же самые раунды с теми же ключами раундов; в качестве примера в Приложении приведено описание дешифрования для шифра RC5.

Таким образом, схематично процесс шифрования для RC5, RC6 и многих других шифров можно представить как цепочку "элементарных" этапов (или раундов) шифрования

$$x_1 = \mathrm{Encr}_1(x_0, k_1), \quad x_2 = \mathrm{Encr}_2(x_1, k_2), \ldots, \quad x_r = \mathrm{Encr}_r(x_{r-1}, k_r), \tag{1}$$

где x_0 – исходное l-битовое слово, которое необходимо зашифровать, Encr_i – операция (функция) шифрования на i-м этапе, k_i – ключ, используемый на i-м этапе, x_i – l-битовое слово, являющееся "выходом" i-го этапа и "входом" $(i+1)$-го, и наконец, x_r – результат шифрования.

В данной статье мы описываем новую атаку на блоковые шифры данного типа, названую градиентной статистической, и в качестве примера исследуем возможность ее применения для криптоанализа шифра RC5. Приведенные экспериментальные данные позволяют сделать вывод о том, что эта атака может быть применена и что для некоторых режимов этого шифра ее трудоемкость может быть существенно меньше, чем у прямого перебора. Здесь стоит отметить, что хотя предлагаемая нами атака ранее не описывалась и является новой, исследование статистических свойств блоковых шифров в криптоанализе использовалось (см. [6, 7]).

Описание предлагаемой атаки дано в § 2, а в § 3 исследуется возможность ее применения к шифру RC5.

§ 2. Описание статистической атаки

Описываемый нами метод относится к классу атак с выбираемым шифруемым текстом (chosen plaintext attack) (см. [1, 3, 5]). При реализации этой атаки криптоаналитик может подавать на вход шифра любой текст и анализировать полученное зашифрованное сообщение. Цель атаки – нахождение (секретного) ключа, причем при этом предполагается, что криптоаналитик знает все характеристики шифра, кроме этого ключа. Такие атаки представляют практический интерес, и считается, что современные блоковые шифры должны быть стойки к ним [1, 3].

Мы рассматриваем шифры, в которых кодируемое битовое (двоичное) сообщение является блоком длины l, $l > 1$, и шифруется с помощью ключа K, являющегося словом из $|K|$ случайно выбранных бит. (Здесь и ниже $|u|$ – длина u, если u – слово, и мощность u, если u – множество.)

У большинства современных шифров существует этап инициализации, в ходе которого начальный ключ K преобразуется в так называемые ключи раундов k_1, $k_2, \ldots, k_r$, которые используются последовательно для шифрования на разных этапах (см. (1)). В разных шифрах эта процедура осуществляется по-разному, причем это зависит не только от шифра, но и от значений длин блока l, ключа k и числа раундов r, которые для многих шифров являются параметрами. Например, для шифра RC5 длина блока может принимать значения 32, 64 или 128 бит, количество раундов может быть любым целым числом, а длина ключа должна быть кратна 8 и может принимать любое значение, начиная с 8 бит. Отметим, что значения $l = 64$, $r = 12$, $k = 128$ рекомендованы разработчиками и широко исследованы. Часто рассматриваются и схемы, в которых длина ключа K равна суммарной длине ключей раундов ($|K| = \sum_{i=1}^{r} |k_i|$).

Дешифрование проводится по схеме, обратной к шифрованию (1):

$$x_{r-1} = \mathrm{Decr}_r(x_r, k_r), \quad x_{r-2} = \mathrm{Decr}_{r-1}(x_{r-1}, k_{r-1}), \ \ldots, \ x_0 = \mathrm{Decr}_1(x_1, k_1), \qquad (2)$$

где используются те же ключи раундов, а операции Decr_i обратны этапам кодирования Encr_i.

Оценим трудоемкость атаки полного перебора. Для ее проведения достаточно иметь одно зашифрованное сообщение (двоичное слово), длина которого не меньше длины ключа. Затем необходимо пытаться дешифровать это зашифрованное сообщение, последовательно перебирая все возможные ключи в каком-либо порядке и сравнивая полученный результат с исходным незашифрованным текстом; совпадение означает, что неизвестный ключ найден. Обычно предполагается, что ключ принимает любое значение из множества всех двоичных слов длины $|K|$ с вероятностью $2^{-|K|}$, поэтому среднее значение числа перебираемых ключей равно $\frac{2^{|K|}+1}{2}$.

К современным блоковым шифрам предъявляется много различных требований, одно из которых можно сформулировать следующим образом: любое зашифрованное сообщение должно быть "похоже" на реализацию бернуллиевского процесса, в котором вероятность порождения нуля и единицы равна 1/2. (В дальнейшем для краткости будем называть такие последовательности случайными.) В частности, все шифры, принимавшие участие в конкурсе на шифр 21-го века, проводившемся в США в 1999–2001 гг., проверялись на выполнение этого условия (см. [8, 9]). Мы не будем останавливаться на логическом анализе этого требования (которое в некотором смысле вообще не выполнимо), а приведем пример, поясняющий его смысл. Для этого определим l-битовое слово α_i как двоичную запись числа i, $i = 0, 1, 2, \ldots, 2^l - 1$, где, как и ранее, l – длина блока рассматриваемого шифра (т.е. α_0 состоит из l-битовой цепочки двоичных нулей, α_1 – из $(l-1)$ нулей и единицы, α_2 – из $(l-2)$ нулей, после которых идет последовательность 10 и т.д.). От современного блокового шифра требуется, чтобы при любом значении ключа последовательность l-битовых слов

$$\mathrm{Encr}(\alpha_0)\mathrm{Encr}(\alpha_1)\mathrm{Encr}(\alpha_2)\ldots,$$

рассматриваемая как двоичная последовательность, была статистически не отличима от случайной. (Здесь $\mathrm{Encr}(\alpha_i)$ означает зашифрованное слово α_i.) Это требование, в частности, позволяет использовать блоковые шифры как генераторы псевдослучайных чисел (см. [3–5]).

Перейдем к описанию предлагаемой статистической атаки на блоковые шифры, у которых кодирование и декодирование разбивается на последовательность раундов (1) и (2), начав с очень неформального предварительного рассмотрения. При этом мы будем использовать совершенно не строгие термины – "более" и "менее" случайные последовательности, понимая под этим, что некоторая последовательность более случайна, чем другая, если отклонения от случайности у первой достоверно выявляются при бóльшей длине, чем у второй. (При этом предполагается, что используется некоторый статистический тест при одном и том же уровне значимости. Другое "определение" более случайной последовательности – величина статистики критерия для этой последовательности меньше, чем для менее случайной.) Предположим, что на вход шифра, ключ которого неизвестен, подаются последовательно слова $\alpha_0\alpha_1\alpha_2\ldots$ Очевидно, эта последовательность "очень" не случайна. Последовательность

$$\mathrm{Encr}_1(\alpha_0,k_1)\mathrm{Encr}_1(\alpha_1,k_1)\mathrm{Encr}_1(\alpha_2,k_1)\ldots$$

после первого раунда шифрования, которую мы обозначим через $\beta_0\beta_1\ldots$, "более" случайна, чем исходная; получаемая после второго раунда последовательность

$$\mathrm{Encr}_2(\beta_0,k_2)\mathrm{Encr}_2(\beta_1,k_2)\mathrm{Encr}_2(\beta_2,k_2)\ldots$$

еще более случайна и т.д. Наконец, полученная после последнего раунда последовательность $\omega_0\omega_1\omega_2\ldots$ более случайна, чем предыдущая. Это неформальное утверждение подтверждается экспериментально в приведенных ниже данных для шифра RC5 и в многочисленных работах (см., например, [6–9]) практически для всех известных шифров рассматриваемого типа; объяснение этого факта довольно очевидно – шифрование на каждом раунде приводит к "перемешиванию" и тем самым повышает "случайность" шифруемых данных. Отметим и очевидное следствие – при дешифровании последовательности $\omega_0\omega_1\omega_2\ldots$ по схеме (2) случайность получаемых данных последовательно уменьшается. Это, конечно, справедливо только в том случае, когда при дешифровании используются "истинные" ключи раундов. Если же при "дешефровании", скажем, на первом раунде $x_{r-1}=\mathrm{Decr}_r(x_r,k_r)$ (соответствующем последнему раунду при шифровании (см. (1), (2))) вместо истинного ключа k_r используется какое-либо другое слово k_r^* той же длины, то эффект преобразования $\mathrm{Decr}_r(x_r,k_r^*)$ будет таким же, как при шифровании – выходная последовательность будет более случайной, чем входная. Это важное для нас наблюдение в общем виде состоит в следующем: при дешифровании на j-м раунде при использовани "неправильного" ключа k_j^* (вместо "правильного" ключа k_j) случайность выходной последовательности возрастает, тогда как при использовании "правильного" k_j – убывает.

На этом наблюдении и базируется предлагаемая атака, которая теперь схематично может быть описана следующим образом.

1. **Формулировка задачи.** Дан шифр, для которого шифрование и дешифрование проводятся по схемам (1), (2) соответственно. Предполагается, что все параметры шифра, кроме ключа K, известны. *Цель атаки* – найти неизвестные ключи раундов k_1, $k_2,\ldots,k_r$, где, как и ранее, r – число раундов (что эквивалентно нахождению K, так как дает возможность дешифровать любое сообщение, зашифрованное с этим ключом).

2. **Описание схемы алгоритма.** При проведении описываемой атаки сначала на вход шифра подается "простая" последовательность из m_r l-битовых слов (например, вышеописанная $\alpha_0\alpha_1\alpha_2\ldots\alpha_{m_r}$), где m_r – параметр метода. Обозначим полученную на выходе зашифрованную последовательность через $\omega_0\omega_1\omega_2\ldots\omega_{m_r}$. Предполагается, что используется некоторая количественная мера случайности, которую обозначим через $\gamma(w)$, где w – двоичное слово. (Например, в дальнейшем в качестве такой меры будет использоваться статистика, применяемая в широко известном критерии Пирсона χ^2.)

После этого поочередно для всех возможных значений ключа r-го раунда k_r вычисляем последовательность $\Gamma_r(u)$, определяемую как

$$\Gamma_r(u) = \mathrm{Decr}_r(\omega_0, u)\mathrm{Decr}_r(\omega_1, u)\mathrm{Decr}_r(\omega_2, u)\ldots\mathrm{Decr}_r(\omega_{m_r}, u), \tag{3}$$

где $u \in \{0,1\}^{|k_r|}$, и оцениваем "степень" ее случайности. Затем находим такое значение u^*, для которого случайность последовательности $\Gamma_r(u^*)$ ($\gamma(\Gamma_r(u^*))$) минимальна среди всех значений $\gamma(\Gamma_r(u)), u \in \{0,1\}^{|k_r|}$, и полагаем, что (неизвестный) ключ r-го раунда равен u^*: $k_r = u^*$. Отметим сразу, что количество операций дешифрования на этом этапе пропорционально $2^{|k_r|}m_r$.

Затем повторяем аналогичные вычисления для поиска ключа $(r-1)$-го раунда k_{r-1}, используя в качестве исходной последовательность $\Gamma_r(k_r)$ $(= \Gamma_r(u^*))$ (см. (3)). Точнее, вычисляем последовательность

$$\Gamma_{r-1}(u) = \mathrm{Decr}_{r-1}(\mathrm{Decr}_r(\omega_0, k_r), u)\mathrm{Decr}_{r-1}(\mathrm{Decr}_r(\omega_1, k_r), u)\ldots, \tag{4}$$

где теперь $u \in \{0,1\}^{|k_{r-1}|}$, и оцениваем случайность этой последовательности. Мы считаем, что количество l-битовых слов в этой последовательности, которое мы обозначим через m_{r-1}, не превосходит m_r (в противном случае можно вычислить недостающие слова, хотя, как будет видно из последующих данных, длина последовательности $\Gamma_{r-1}(u)$ (т.е. m_{r-1}) будет обычно меньше длины $\Gamma_r(u)$ (т.е. m_r), так как первая последовательность "менее" случайна, чем вторая). Слово u^{**}, минимизирующее случайность последовательности $\Gamma_{r-1}(u)$, и будет значением ключа $(r-1)$-го раунда. На этом этапе количество операций дешифрования пропорционально $2^{|k_{r-1}|}m_{r-1}$.

Повторяя описанные вычисления последовательно, мы найдем значения ключей раундов $k_{r-1}, k_{r-2}, k_{r-3}, \ldots, k_1$. Суммарное количество операций при нахождении всех ключей раундов пропорционально $\sum\limits_{i=1}^{r} 2^{|k_i|}m_i$; в типичном случае, когда длины ключей раундов равны ($|k_i| = |k|$), количество операций пропорционально величине $r\, m_{\max}\, 2^{|k|}$, тогда как для прямого перебора – $2^{|K|}$ (где $m_{\max} = \max\limits_{i=1,\ldots,r} m_i$ и K – (общий) ключ шифра). Эта разница в показателях степени и определяет область применимости предлагаемой атаки: если $r m_{\max}$ меньше $2^{|K|-|k|}$, то количество операций у предлагаемого метода меньше, чем у полного перебора ключей.

3. **Модификации, параметры и варианты описанного метода.** Мы описали основную идею метода в "чистом" виде, а здесь остановимся на возможных вариантах его реализации.

Во-первых, мера случайности $\gamma(\cdot)$ является параметром метода, причем можно использовать не только различные меры для разных шифров, но и для разных раундов. Как указано выше, любой статистический тест, который применим для проверки основной гипотезы H_0 о том, что двоичная последовательность порождается бернуллиевским источником с равными вероятностями для нуля и единицы, против альтернативной гипотезы H_1, являющейся отрицанием H_0, может быть использован для этой цели. При этом $\gamma(\cdot)$ может быть равна величине статистики критерия.

Во-вторых, в отличие от описанного выше варианта при поиске ключа j-го раунда можно выбирать не один "истинный" ключ, а несколько (например, s) "подозрительных" значений u, т.е. s таких слов, у которых мера случайности $\gamma(\Gamma_j(u))$ минимальна (среди $u \in \{0,1\}^{|k_j|}$). Кроме того, при поиске простых последовательностей и ключей раундов естественно использовать последовательные методы, аналогичные последовательным критериям в математической статистике.

В-третьих, начальная "очень неслучайная" последовательность $\alpha_0\alpha_1\alpha_2\ \ldots\alpha_{m_r}$ может выбираться различным образом. Например, кажется разумным выбирать по-

следовательность, в которой соседние слова α_i, α_{i+1} не только содержат много одинаковых символов, но и отличаются только одним знаком (такая последовательность может быть построена на основе кодов Грея (см. их описание, например, в [10])). Наконец, часть двоичных символов в словах последовательности $\alpha_0\alpha_1\alpha_2\ldots\alpha_{m_r}$ может выбираться случайно, а оставшиеся полагаются равными нулю (как в [6, 7]) и т.д.

Последняя модификация связана с тем фактом, что у многих современных шифров при большом числе раундов даже "очень" не случайная последовательность после шифрования статистически не отличима от случайной (при использовании известных статистических тестов и при приемлемом времени вычислений). Пусть, например, шифр использует r раундов и для некоторой "простой" начальной последовательности $\alpha^0 = \alpha_0^0\alpha_1^0\alpha_2^0\ldots\alpha_m^0$ последовательности

$$\alpha^1 = \mathrm{Encr}_1(\alpha_0^0, k_1)\mathrm{Encr}_1(\alpha_1^0, k_1)\mathrm{Encr}_1(\alpha_2^0, k_1)\ldots\mathrm{Encr}_1(\alpha_m^0, k_1),$$
$$\alpha^2 = \mathrm{Encr}_2(\alpha_0^1, k_2)\mathrm{Encr}_2(\alpha_1^1, k_2)\mathrm{Encr}_2(\alpha_2^1, k_2)\ldots\mathrm{Encr}_2(\alpha_m^1, k_2),$$
$$\alpha^d = \mathrm{Encr}_d(\alpha_0^{d-1}, k_d)\mathrm{Encr}_d(\alpha_1^{d-1}, k_d)\mathrm{Encr}_d(\alpha_2^{d-1}, k_d)\ldots\mathrm{Encr}_d(\alpha_m^{d-1}, k_d)$$

не случайны при всех ключах раундов $k_1, \ldots, k_d, d < r$. Тогда описанная выше атака может быть модифицирована следущим образом: для каждого набора ключей $k_{d+1}, \ldots, k_r$ раундов $d+1, \ldots, r$ повторяем описанную выше процедуру нахождения неизвестных ключей $k_1, \ldots, k_d$ раундов $1, \ldots, d$. Другими словами, ключи $k_{d+1}, \ldots, k_r$ находим полным перебором, а $k_1, \ldots, k_d$ – по описанному выше методу. Для проведения такой комбинированной атаки потребуется количество операций

$$m\ 2^{\sum\limits_{j=d+1}^{r}|k_j|}\ \sum_{j=1}^{d} 2^{|k_j|},$$

что при некоторых соотношениях параметров может быть меньше, чем количество операций, необходимое при прямом переборе всех ключей.

§ 3. Эксперименты с шифром RC5

Мы начнем описание с экспериментального анализа "степени случайности" зашифрованных сообщений в зависимости от числа раундов, точнее, полураундов, как упоминалось выше. (Используем выражения типа "3,5" раунда вместо, скажем, 7-й полураунд. К сожалению, эта терминология является общепринятой в работах, касающихся RC5, RC6 и ряда других шифров.)

Первый вопрос, который мы исследовали экспериментально[2], касался возможности различения зашифрованных с помощью RC5 "простых", явно не случайных, последовательностей при разном числе (полу)раундов. Для этого мы использовали в качестве исходной "неслучайной" вышеупомянутую последовательность α_i, $i = 0, 1, \ldots$, где α_i – запись числа i в двоичной системе счисления, под которую отводится 64 бита. (Напомним, что мы рассматриваем RC5 с длиной шифруемого блока 64 бита.) Во всех случаях эта последовательность зашифровывалась с помощью указанного шифра с заданным количеством полураундов, и по полученной последовательности проверялась гипотеза H_0 о том, что двоичная последовательность порождается бернуллиевским источником с равными вероятностями нуля и единицы, против альтернативной гипотезы H_1, являющейся отрицанием H_0. В дальнейшем, чтобы избегать повторов, будем называть эту задачу "гипотезой о случайности".

Понятно, что выбор статистического теста для проверки гипотез играет важную роль в описываемой атаке, поэтому мы кратко остановимся на этом вопросе. В на-

[2] Вычисления проводились на суперкомпьютерах Института вычислительных технологий СО РАН и Новосибирского государственного университета.

Таблица 1

Число последовательностей (из 100), для которых гипотеза о случайности была отвергнута

Раунды / Номер ключа t	1 2^{18}	1,5 2^{18}	2 2^{18}	2,5 2^{20}	3 2^{20}
1	100	63	64	51	52
2	100	100	100	74	70
3	100	61	61	17	17
4	100	81	78	62	64
5	100	100	100	65	6
6	100	85	86	12	9
7	100	100	100	11	8
8	100	98	99	99	99
9	100	80	79	14	14
10	100	100	100	7	5

стоящее время имеется довольно много работ, посвященных построению и исследованию тестов для проверки гипотезы о случайности, что, по-видимому, объясняется важностью этой задачи для криптографии, численных методов и других многочисленных приложений. Так, Институт стандартов и технологий США (NIST) недавно провел исследование известных тестов для проверки гипотезы о случайности и рекомендовал 16 методов для практического применения в криптографии (см. [11]). В [12] показано, что описанные в [13, 14] тесты превосходят методы из [11], что подтверждалось и нашими предварительными расчетами, связанными с RC5. Поэтому мы использовали для анализа тесты "стопка книг" и "адаптивный χ^2" из [13, 14] соответственно. Оказалось, что мощность теста "стопка книг" в среднем выше, чем у адаптивного теста χ^2, но скорость вычислений у последнего значительно выше и он более удобен для реализации на многопроцессорном компьютере. Поэтому предпочтение было отдано этому тесту, и все приводимые ниже данные получены для него.

В табл. 1 приведены данные по проверке гипотезы о случайности при использовании адаптивного критерия χ^2 для RC5 с разным количеством раундов. Все вычисления проводились для 10 случайно выбранных ключей и повторялись 100 раз при шифровании следующих 100 последовательностей слов длины t:

$$\alpha_0\alpha_1\ldots\alpha_{t-1},\ \alpha_t\alpha_{t+1}\ldots\alpha_{2t-1},\ldots,\ \alpha_{99t}\alpha_{99t+1}\ldots\alpha_{100t-1}\,, \tag{5}$$

где t – длина одной подпоследовательности. В таблице приведено число случаев, когда гипотеза о случайности отвергалась при уровне значимости 0,0001. Например, мы видим из таблицы, что гипотеза о случайности была отвергнута 100 раз (из 100) при использовании первого (случайно выбранного) ключа и при длине последовательности $t = 2^{18}$ слов. Таким образом, мы видим из этой таблицы, что зашифрованные последовательности явно не случайны, так как в противном случае в среднем гипотеза отвергалась бы приблизительно в $0{,}0001 \cdot 100 = 0{,}1$ случаях из 100.

Для большего числа раундов мы проводили вычисления с меньшим числом вариантов (или повторностей), так как в этом случае требуются последовательности большей длины и, соответственно, большее время вычислений. Снова проверялась гипотеза о случайности H_0 для той же зашифрованной последовательности $\alpha_0\alpha_1\ldots\alpha_{t-1}$ с разными (случайно выбранными) ключами и разным числом раундов; результаты приведены в табл. 2. Мы видим, что зашифрованная последовательность $\alpha_0\alpha_1\ldots\alpha_{t-1}$ довольно надежно отличается от случайной до восьмого раунда.

Как сказано при описании теста, главное предположение, без выполнения которого данная атака невозможна, состоит в следующем: при дешифровании на любом

Таблица 2

Проверка гипотезы о случайности для большого числа раундов при уровне значимости 0,01

Раунд	Длина t	Число тестов	Число случаев, для которых гипотеза о случайности отвергнута
5	2^{28}	30	30
5,5	2^{29}	22	10
6	2^{31}	6	6
6,5	2^{32}	6	6
7	2^{32}	6	5
7,5	2^{33}	3	3
8	2^{37}	3	2

Таблица 3

Число "неслучайных" последовательностей (из 100) при дешифровании с истинным и 10 случайными ключами полураундов

2,5 раунда	Длина t одной последовательности (из 100) равна 2^8										
Ключ / Серия	Истинный	1	2	3	4	5	6	7	8	9	10
1	54	9	10	10	15	13	13	8	10	19	12
2	69	34	34	35	34	36	33	34	36	39	33
3	87	44	37	36	38	38	37	41	42	42	41
3 раунда	Длина t одной последовательности (из 100) равна 2^{16}										
Ключ / Серия	Истинный	1	2	3	4	5	6	7	8	9	10
1	97	81	84	84	81	84	83	84	82	83	83
2	73	35	39	36	32	36	32	33	40	39	42
3	94	0	1	2	0	1	1	0	4	0	1
3,5 раунда	Длина t одной последовательности (из 100) равна 2^{19}										
Ключ / Серия	Истинный	1	2	3	4	5	6	7	8	9	10
1	100	28	16	23	9	15	26	18	17	22	22
2	48	9	10	9	11	10	8	9	10	10	11
3	65	20	21	18	20	19	20	19	19	18	17

раунде при использовании "неправильного" ключа k^* (вместо "правильного" ключа k) случайность выходной последовательности возрастает, тогда как при использовании "правильного" k – убывает. Выполнение этого предположения проверялось экспериментально по следующей схеме: для трех случайно выбранных ключей 100 вышеописанных сообщений (5) шифровались с помощью RC5 до j-го полураунда. Затем проводилось дешифрование на один полураунд с "истинным" ключом полураунда и с 10 случайно выбранными "неправильными" ключами, и для всех 11 последовательностей оценивалась степень случайности полученных после этого преобразования данных. Отметим, что в соответствии с нашей гипотезой разница в степени случайности последовательности, дешифрованной с "правильным" ключом, и 10 других, "дешифрованных" со случайными ключами, должна соответствовать различиям в случайности, получаемым при шифровании в один дополнительный раунд. (Действительно, правильный ключ уменьшает случайность на полраунда, а неправильный увеличивает случайность на полраунда.)

В табл. 3 приведены данные экспериментов для различного количества раундов при уровне значимости 0,0001. Выбор параметров теста и длины последовательностей t определялся в ходе предварительных экспериментов, проводимых по независимым данным, полученным с использованием других случайных ключей.

Таблица 4

Разница в сложности последовательностей, дешифрованных с "истинным" ключом полураунда и с 5 случайными

Раунд	Ключ					
	Истинный	1	2	3	4	5
4	10	4	4	4	3	3
4,5	5	0	0	0	0	0

Оказалось, что среди 100 последовательностей, расшифрованных с правильным ключом полураунда, неслучайными признаны 54 из 100 (при уровне значимости 0,0001), тогда как из 100 последовательностей, "расшифрованных" с первым "неправильным" ключом, признано неслучайными только 9 последовательностей, со вторым – 10, с третьим – 10 и т.д.; т.е. последовательности, расшифрованные с правильным ключом, менее случайны, чем последовательности, "расшифрованные" с неправильными ключами.

К сожалению, проведение расчетов по этой схеме для большего числа раундов оказалось практически невозможным из-за резкого увеличения времени вычислений. (Действительно, при этой схеме для каждого полураунда проводятся вычисления для 330 файлов одной длины – 3 серии, 11 ключей полураунда и 100 подпоследовательностей.) В табл. 4 приведены результаты для 4 и 4,5 раундов, в которых использовалась последовательность (5) длины 2^{24} бит. Она зашифровывалась со случайно выбранным ключом, а затем расшифровывалась один раз на полраунда с "правильным" ключом полураунда, и 5 раз с неправильными, случайно выбранными. Эти вычисления повторялись 10 раз; все остальные условия были те же, что и в ранее описанных экспериментах. Мы видим, что при 4 раундах шифрования последовательности, расшифрованные с правильным ключом, признаны неслучайными в 10 случаях из 10, тогда как "расшифрованные" с неправильным ключом – только в 4 или 3 случаях из 10. Аналогично, при 4,5 раундах последовательности, расшифрованные с правильным ключом, признаны неслучайными в 5 случаях из 10, а все, "расшифрованные" с неправильным ключом, признаны случайными.

Мы видим, что данные, приведенные в табл. 1–4, подтверждают предположения, выполнение которых необходимо для принципиальной возможности проведения предлагаемой атаки: во-первых, "случайность" зашифрованной последовательности возрастает при увеличении числа полураундов; во-вторых, "случайность" последовательности, "расшифрованной" с неправильным ключом полураунда, больше, чем у расшифрованной с правильным ключом.

Таким образом, приведенные нами данные экспериментов показывают, что условия, необходимые для проведения градиентной статистической атаки на шифр RC5, выполняются. Это, в свою очередь, позволяет сделать вывод о принципиальной возможности проведения этой атаки на шифр RC5. Кроме того, полученные данные позволяют также предположить, что градиентная статистическая атака может быть применима и к другим шифрам рассматриваемого вида.

ПРИЛОЖЕНИЕ. Описание шифра RC5

Алгоритм 1 (шифрование).

На входе: $2w$-бит текст $M = (A, B)$; r; ключ $K = K[0] \dots K[b-1]$.

На выходе: $2w$-бит шифротекст C.

Для шифрования используются операции сложения по модулю $2^w(\boxplus)$, $XOR(\oplus)$ и циклического сдвига(влево $\hookleftarrow$).

1. Вычисляем $2r+2$ ключа (полу)раундов $K_0, \dots, K_{2r+1}$ по Алгоритму 2, используя K и r.
2. $A \leftarrow A \boxplus K_0$, $B \leftarrow B \boxplus K_0$.

Таблица 5

Таблица, используемая при формировании ключей полураундов в RC5

w	16	32	64
P_w	B7E1	B7E15163	B7E15162 8AED2A6B
Q_w	9E37	9E3779B9	9E3779B9 7F4A7C15

3. Цикл: For i to r do:
$A \leftarrow ((A \oplus B) \hookleftarrow B) \boxplus K_{2i}$, (комментарий: полураунд $2i$),
$B \leftarrow ((B \oplus A) \hookleftarrow A) \boxplus K_{2i+1}$. (комментарий: полураунд $2i+1$).
4. Выход $C \leftarrow (A, B)$.

Замечание. Для дешифрования используется алгоритм шифрования, использующий шифротекст $C = (A, B)$ следующим образом (через $\boxminus$ обозначим вычитание по модулю 2^w и через $\hookrightarrow$ циклический сдвиг вправо):

For i from r down to 1 do : $B \leftarrow ((B \boxminus K_{2i+1}) \hookrightarrow B) \oplus A$, $A \leftarrow ((A \boxminus K_{2i}) \hookrightarrow B) \oplus B$.
Окончательно получаем $M \leftarrow (A \boxminus K_0, B \boxminus K_1)$.

Алгоритм 2 (инициализация ключа).

На входе: размер слова w; число раундов r; b-байтный ключ $K[0] \dots K[b-1]$.

На выходе: подключ $K_0, \dots K_{2r+1}$ (где K_i – w-битные слова).

1. Полагаем $u = w/8$ (число байт в слове) и $c = \lceil b/u \rceil$.
2. Далее, пусть $K[j] \leftarrow 0$ для всех $b \leqslant j \leqslant c * u - 1$.
Цикл: For i from 0 to $c-1$ do: $L_i \leftarrow \sum_{j=0}^{u-1} 2^{8j} K[i \cdot u + j]$.
3. $K_0 \leftarrow P_w$; for i from 1 to $2r+1$ do: $K_i \leftarrow K_{i-1} \boxplus Q_w$ (см. табл. 5).
4. $i \leftarrow 0, j \leftarrow 0, A \leftarrow 0, B \leftarrow 0, t \leftarrow \max(c, 2r+2)$ For s from 1 to $3t$ do:
(а) $K_i \leftarrow (K_i \boxplus A \boxplus B) \hookleftarrow 3$, $A \leftarrow K_i, i \leftarrow i+1 \bmod (2r+2)$;
(б) $L_i \leftarrow (L_i \boxplus A \boxplus B) \hookleftarrow (A \boxplus B)$, $B \leftarrow L_i, j \leftarrow j+1 \bmod c$.

СПИСОК ЛИТЕРАТУРЫ

1. *Schneier B.* A Self-Study Course in Block-Cipher Cryptanalysis // Cryptologia. 2000. V. 24. № 1. P. 18–34.
2. *Biryukov A.* Block Ciphers and Stream Ciphers: The State of the Art // Cryptology ePrint Archive: Report 2004/094 (http://eprint.iacr.org/2004/094/)
3. *Menzes A., van Oorschot P., Vanstone S.* Handbook of Applied Cryptography. Boca Raton: CRC Press, 1997.
4. *Рябко Б.Я., Фионов А.Н.* Основы современной криптографии для специалистов в информационных технологиях. М.: Научный мир, 2004.
5. *Schneier B.* Applied Cryptography. New York: Wiley, 1996.
6. *Knudsen R.L., Meier W.* Correlation in RC6. Private Communication (Available from http://www.ii.uib.no/ larsr/papers/rc6.ps).
7. *Shimoyama T., Takeuchi K., Hayakawa Ju.* Correlation Attack to the Block Cipher RC5 and the Simplified Variants of RC6 // Proc. AES3. New York, 2001.
http://csrc.nist.gov/encryption/aes/round2/conf3/aes3papers.html
8. *Nechvatal J., Barker E., Bassham L., Burr W., Dworkin M., Foti J., Roback E.* Report on the Development of the Advanced Encryption Standart (AES), 2000.
http://csrc.nist.gov/encryption/aes/round2/r2report.pdf
9. *Soto J., Bassham L.* Randomness Testing of the Advanced Encryption Standard Finalist Candidates // Proc. AES3. New York, 2001.
http://csrc.nist.gov/encryption/aes/round2/conf3/papers/30-jsoto.pdf
10. *Knuth D.E.* The Art of Computer Programming. V. 1. New York: Addison Wesley, 1981.

11. *Rukhin A., Soto J., Nechvatal J., Smid M., Barker E.* A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22 (with revision dated May, 15, 2001). http://csrc.nist.gov/rng/SP800-22b.pdf
12. *Ryabko B.Ya., Monarev V.A.* Using Information Theory Approach to Randomness Testing // J. Statistical Planning Inference. 2005. V. 133. № 1. P. 95–110.
13. *Рябко Б.Я., Стогниенко В.С., Шокин Ю.И.* Адаптивный критерий χ^2 для различения близких гипотез при большом числе классов и его применение к некоторым задачам криптографии // Пробл. передачи информ. 2003. Т. 39. № 2. С. 53–62.
14. *Рябко Б.Я., Пестунов А.И.* "Стопка книг" как новый статистический тест для случайных чисел // Пробл. передачи информ. 2004. Т. 40. № 1. С. 73–78.

Рябко Борис Яковлевич
Сибирский государственный университет
телекоммуникаций и информатики,
Институт вычислительных технологий СО РАН
boris@ryabko.net
Монарев Виктор Александрович
Шокин Юрий Иванович
Институт вычислительных технологий СО РАН
vitox@gorodok.net
shokin@ict.nsc.ru

Поступила в редакцию
07.12.2004