УДК 519.7

Применение метаэвристических алгоритмов псевдобулевой оптимизации к поиску линеаризующих множеств в криптоанализе криптографических генераторов

Антонов Кирилл Валентинович[1], **Семенов Александр Анатольевич**[2]

[1] Иркутский государственный университет, e-mail: aknitr@mail.ru

[2] Институт динамики систем и теории управления им. В. М. Матросова СО РАН, e-mail: biclop@rambler.ru

В статье рассматривается новый подход к построению атак типа «угадывай и определяй» на генераторы ключевого потока, основанный на понятии линеаризующего множества. Сложность атаки для конкретного линеаризующего множества оценивается как значение специально определённой псевдобулевой функции. Для решения задачи оптимизации псевдобулевой функции реализованы метаэвристические алгоритмы поиска: tabu search, генетический алгоритм, (1 + 1) эволюционный алгоритм, GBFS. Приведены оценки сложности атак указанного типа, которые были построены для поточных шифров A5/1 и ASG.

Ключевые слова: алгебраический криптоанализ, атаки из класса «угадывай и определяй», псевдобулева оптимизация, метаэвристические алгоритмы.

Понятие линеаризационного множества ввёл в 2003 году Г. П. Агибалов в работе [1]. Атаки на криптографические функции, использующие данное понятие, относятся к области, известной как алгебраический криптоанализ [2]. В алгебраическом криптоанализе задача обращения криптографической функции сводится к поиску решения системы алгебраических уравнений над некоторым конечным полем (чаще всего над $GF(2)$). Хорошо известно (см., например, [3]), что задача проверки совместности даже квадратичной системы уравнений над $GF(2)$ является NP-полной. Соответственно, в любом случае при решении таких систем приходится использовать переборную стратегию. Техника линеаризационных множеств, предложенная в [1], подразумевает ослабление систем уравнений, кодирующих криптоанализ рассматриваемой функции, до линейных систем засчёт подстановки угаданных значений некоторых переменных. Таким образом, атаки из [1] относятся к атакам типа «угадывай и определяй» (guess and determine) [2]. В целом ряде случаев множество угадываемых бит можно подобрать так, что трудоёмкость соответствующей атаки будет существенно ниже, чем при полном переборе вариантов секретного ключа.

Криптографический генератор ключевого потока — это дискретная функция вида

$$g : \{0,1\}^n \to \{0,1\}^*, \tag{1}$$

которая задаётся некоторым эффективным алгоритмом. Рассматриваются всюду определённые функции вида (1). Произвольный $\gamma \in \{0,1\}^m$ такой, что $\gamma = g(\alpha)$ для некоторого $\alpha \in \{0,1\}^n$, называется фрагментом ключевого потока длины m генератора g. С функцией (1) естественным образом связывается функция

$$f : \{0,1\}^n \to \{0,1\}^m, \tag{2}$$

область значений ($Range\ f$) которой совпадает с множеством всех возможных фрагментов ключевого потока длины m, порождаемых генератором (1). Задача обращения ключевого потока длины m генератора g ставится следующим образом: по известному $\gamma \in Range\ f$ требуется найти такой $\alpha \in \{0,1\}^n$, что $f(\alpha) = \gamma$.

Известно, что по алгоритму, задающему f, можно эффективно построить систему алгебраических уравнений степени не более 2 над полем $GF(2)$. Для этой цели можно использовать техники сводимости, подобные тем, которые были описаны в [4]. Пусть E_f — результирующая система и X — множество фигурирующих в ней булевых переменных. В X выделим два подмножества: $X^{in}, \left|X^{in}\right| = n$ и $Y, |Y| = m$. Переменные, образующие X^{in}, соответствуют входу f, а переменные из Y — её выходу. Результат подстановки в E_f известного $\gamma \in Rangef$ обозначим через $E_f(\gamma)$ (подстановка определяется стандартным способом — см., например [5]).

В соответствии с [1] назовём множество $X', X' \subset X$ линеаризационным, если для любого $\gamma \in Rangef$ и любого набора значений переменных из X' (используем обозначение $\beta \in \{0,1\}^{|X'|}$) подстановка β в $E_f(\gamma)$ даёт линейную систему. Таким образом, если X' — линеаризационнное множество, то задачу обращения произвольного $\gamma \in Rangef$ можно эффективно свести к решению $2^{|X'|}$ систем линейных уравнений над $GF(2)$. Известен целый ряд примеров, когда такая атака оказывается более эффективной, чем опробование всех 2^n возможных входов рассматриваемой функции.

Приведём один пример такого типа. Речь идёт об атаке, предложенной Россом Андерсоном [6] на известный генератор ключевого потока A5/1, который долгое время использовался для шифрования траффика в сетях сотовой связи. Данный алгоритм состоит из 3 несинхронно сдвигаемых регистров сдвига с линейной обратной связью (LFSR, [7]), детали его работы можно найти в многочисленных статьях, в том числе в Wikipedia. Р. Андерсон заметил, что угадывание 53 бит (все биты первого и третьего регистров и младшие 11 бит второго регистра) позволяет эффективно восстановить значения оставшихся 11 бит секретного ключа для любого фрагмента ключевого потока длины не менее 64. Несложно показать, что множество Андерсона — это линеаризационное множество в смысле [1].

В настоящей работе рассматривается одно обобщение понятия линеаризационного множества, которое можно построить, отталкиваясь от результатов

статьи [8]. В этой статье были введены т. н. «инверсные множества с лазейками» (Inverse Backdoor Sets, IBS). Концептуально, IBS — это (так же, как и линеаризационное множество) множество угадываемых бит в некоторой атаке типа «угадывай и определяй». Однако для решения ослабленных систем можно использовать алгоритм A решения какой-либо NP-трудной задачи, который работает быстро на значительной доле входов. В [8] для этой цели использовались алгоритмы решения задачи булевой выполнимости (SAT). Главная особенность описанных в [8] атак состоит в том, что не все ослабленные задачи должны быть простыми для алгоритма A. Эффективные IBS-атаки можно строить и для ситуаций, когда простой оказывается лишь некоторая доля от всех возможных ослабленных задач. Мы переносим эту идею на ситуацию, когда в роли A используется произвольный алгоритм решения линейных уравнений над $GF(2)$ (например, метод Гаусса). Получаемое в результате понятие отличается от понятия линеаризационного множества требованием, чтобы система вида $E_f(\gamma)$ линеаризовалась не любыми, а лишь некоторыми наборами значений переменных из X'.

С этой целью мы вводим специальную оценочную функцию, значения которой оценивают долю наборов из $\{0,1\}^{|X'|}$, линеаризующих системы вида $E_f(\gamma)$. Более точно, нам требуется оценить долю булевых векторов из $\{0,1\}^{|X'|}$ подстановка которых в $E_f(\gamma)$ обращает эту систему в линейную. Если эта доля относительно велика, то мы можем повторить нашу атаку применительно к нескольким различным выходам γ. В роли лазеек X' рассматриваются подмножества множества X^{in}. Каждое $X' \in 2^{X^{in}}$ можно задать булевым вектором длины n, в котором единицы отмечают те переменные из X^{in}, которые попадают в X'.

Рассмотрим N случайных входов функции f $\alpha_1, \ldots, \alpha_N$, выбранных из $\{0,1\}^n$ в соответствии с равномерным распределением. Построим выходы $\gamma_1, \ldots, \gamma_N : \gamma_j = f(\alpha_j), j \in \{1, \ldots, N\}$. Рассмотрим произвольное множество $X', X' \in X^{in}$. Тогда, входам α_j соответствуют наборы значений переменных из X', которые будем обозначать через $\beta_j, j \in \{1, \ldots, N\}$. Для входов $\alpha_1, \ldots, \alpha_N$ рассмотрим системы $E_f(\gamma_j, \beta_j), j \in \{1, \ldots, N\}$: каждая такая система получается в результате подстановки γ_j, β_j, порождённых α_j. Обозначим через δ долю систем вида $E_f(\gamma_j, \beta_j)$, которые являются линейными. Рассуждая по аналогии с [8], можем заключить, что δ — оценка вероятности следующего события: результат подстановки векторов γ_j и β_j, порождённых случайным входом $\alpha_j \in \{0,1\}^n$, в E_f — линейная система над $GF(2)$. Для фиксированного X' обозначим эту вероятность через $p_{X'}$.

Определение 1. *В контексте задачи обращения функции (2) назовём множество X' линеаризующим с вероятностью линеаризации $p_{X'}$.*

Таким образом, линеаризационное множество из [1] — это линеаризующее множество с вероятностью линеаризации 1.

С использованием $p_{X'}$ можно построить атаку из класса «угадывай и определяй» по смыслу близкую к атакам, описанным в [8] с той лишь разницей, что успехом в соответствующей последовательности испытаний Бернулли считается ситуация, когда выбран такой вход $\alpha \in \{0,1\}^n$, что $E_f(\gamma, \beta)$ — линейная система над $GF(2)$. Трудоёмкость такой атаки оценивается через оценку вероятности $p_{X'}$. С этой целью рассматривается специальная функция следующего вида:

$$\Phi : \{0,1\}^n \to \mathbb{R} \tag{3}$$

Функция (3) задаётся следующим образом. На вход она получает произвольный вектор $\chi \in \{0,1\}^n$, который задаёт множество X' (единицы в χ соответствуют переменным из X^{in}). Затем строится случайная выборка $\alpha_1, \ldots, \alpha_N$, по которой вычисляется величина δ. Данная величина принимается за оценку вероятности $p_{X'}$, на основании которой строится оценка трудоёмкости атаки из класса «угадывай и определяй», использующей множество угадываемых бит X'. Полученная оценка — значение функции (3).

Всё сказанное означает, что (3) — это псевдобулева функция [9]. Минимальное её значение на гиперкубе $\{0,1\}^n$ — это оценка трудоёмкости наиболее эффективной атаки. Для минимизации (3) мы использовали следующие метаэвристические алгоритмы: алгоритм, использующий поиск с запретами (tabu search, [10]) в том виде, как он описан в [11]; алгоритм, использующий стратегию GBFS (Greedy Best First Search, [11]), а также генетический алгоритм в варианте, который был использован в [13].

В вычислительных экспериментах рассматривались задачи построения линеаризующих множеств и соответствующих атак на следующие криптографические генераторы: A5/1, ASG96 и ASG192 (ASG — alternating step generator). У A5/1 анализировалось 114 бит выходного потока, у ASG92 — 112 бит, у ASG192 — 200 бит. Лучшие результаты экспериментов приведены в следующей таблице.

Таблица 1: Результаты экспериментов. GA — генетический алгоритм, TS — tabu search

Шифр	Алгоритм	Мощность множества	Сложность атаки (число решаемых систем ЛУ)	Вероятность
A5/1	GA	47	5.32e+15	0.0793
ASG96	GA, TS	31	6.44e+9	1
ASG192	GA, TS	65	1.11e+20	1

Из таблицы 1 видно, что лучшие результаты показали алгоритмы GA и TS. В случаях ASG96 и ASG192 эти алгоритмы в качестве множества X' построили множество переменных, которое кодирует управляющий регистр. Данный факт соответствует известной атаке на ASG. Для алгоритма A5/1 с использованием GA была построена атака, трудоёмкость которой примерно в 5 раз ниже, чем у атаки Андерсона. Особо подчеркнём, что в этом случае оценка вероятности $p_{X'}$ существенно меньше 1.

Работа выполнена при финансовой поддержке Российского научного фонда, проект №16-11-10046.

Список литературы

[1] Агибалов Г. П. Логические уравнения в криптоанализе генераторов ключевого потока // Вестник Томского государственного университета. Приложение. 2003. № 6. С. 31–41.

[2] Bard G. Algebraic cryptanalysis. 1st edition. Springer Publishing Company, Incorporated, 2009.

[3] Goldreich O. Computational Complexity: A Conceptual Perspective. Cambridge University Press, 2008.

[4] Otpuschennikov I., Semenov A., Gribanova I., Zaikin O., Kochemazov S. Encoding cryptographic functions to SAT using TRANSALG system // Frontiers in Artificial Intelligence and Applications. 2016. Vol. 285. P. 1594–1595.

[5] Чень Ч., Ли Р. Математическая логика и автоматическое доказательство теорем. М. : Наука, 1983. 360 с.

[6] Anderson R. A5 (Was: Hacking digital phones) / Newsgroup Communication. 1994. URL: http://yarchive.net/phone/gsmcipher.html.

[7] Alfred J. Menezes, Scott A. Vanstone, Paul C. Van Oorschot Handbook of Applied Cryptography. 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 1996.

[8] Semenov A., Zaikin O., Otpuschennikov I., Kochemazov S., Ignatiev A. On cryptographic attacks using backdoors for SAT // The Thirty-Second AAAI Conference on Artificial Intelligence. 2018. P. 6641–6648.

[9] Boros E., Hammer P. L. Pseudo-Boolean optimization // Discrete Applied Mathematics. 2002. Vol. 123, N 1-3. P. 155–225.

[10] Glover F., Laguna M. Tabu Search. Kluwer Academic Publishers, 1997.

[11] Semenov A., Zaikin O. Algorithm for finding partitionings of hard variants of boolean satisfiability problem with application to inversion of some cryptographic functions // SpringerPlus. 2016. 5:554. P. 1–16.

[12] Норвиг П., Рассел С. Искусственный интеллект. Prentice Hall, 1994.

[13] Pavlenko A., Semenov A., Ulyantsev V. Evolutionary computation techniques for constructing SAT-based attacks in algebraic cryptanalysis. Lecture Notes in Computer Science. 2019. 11454. P. 237–253.

Applying Metaheuristic Pseudo-Boolean Optimization Algorithms to Search for Linearizing Sets in Cryptanalysis of Cryptographic Generators

Antonov Kirill Valentinovich[1], **Semenov Aleksandr Anatolevich**[2]

[1] Irkutsk State University, e-mail: aknitr@mail.ru

[2] Institute for System Dynamics and Control Theory SB RAS, e-mail: biclop@rambler.ru

In this paper we consider a new approach to the construction of guess-and-determine attacks on keystream generators based on the concept of linearizing sets. The complexity of the attack for a particular linearizing set is estimated as a value of a specially defined pseudo-Boolean function. To solve the optimization problem for the considered pseudo-Boolean function, various metaheuristic search algorithms are implemented: tabu search, genetic algorithm, (1 + 1) evolutionary algorithm, GBFS. For stream ciphers A5/1 and ASG the complexity estimates of the attacks of considered type are given.

Keywords: algebraic cryptanalysis, guess-and-determine attack, pseudo-boolean optimization, metaheuristic algorithms.