

Выводы

1) Последовательность выходных функций генератора гаммы, построенного на основе полноцикловой подстановки множества состояний V_n , имеет высокую линейную сложность Λ , а именно $2^{n-1} + 1 \leq \Lambda \leq 2^n$.

2) Для порядка множества мономов на периоде последовательности выходных функций генератора верны оценки: $2^{n-1} \leq |M(f(H))| \leq 2^n - 1$. При $n \rightarrow \infty$ и при случайном равновероятном выборе функции переходов h из класса всех полноцикловых подстановок множества V_n математическое ожидание величины $|M(f(H))|/(2^n - 1)$ стремится к 1.

3) Сложность T_n определения начального состояния методом формального кодирования оценивается как $TL(2^{n-1}) < T_n < TL(2^n)$, где $TL(m)$ — сложность решения над $GF(2)$ системы линейных уравнений размера $m \times m$.

ЛИТЕРАТУРА

1. Фомичёв В. М. Дискретная математика и криптология. М.: ДИАЛОГ-МИФИ, 2003.
2. Shamir A., Patarin J., Courtois N., and Klimov A. Efficient Algorithms for solving Overdefined Systems of Multivariate Polynomial Equations // Eurocrypt'2000. Springer. LNCS. 2001. V. 1807.

УДК 65.012.810(075.8)

АВТОМАТИЗИРОВАННЫЕ СРЕДСТВА АНАЛИЗА ПРОТОКОЛОВ¹

А. В. Черемушкин

Приведем примеры средств автоматизированного анализа криптографических протоколов, которые в настоящее время можно отнести к наиболее эффективным, и рассмотрим математические аспекты работы этих систем. Подчеркнем, что данные средства позволяют не только проверять заданные свойства протоколов, но и находить конкретные атаки на протоколы в случае, когда эти свойства не выполнены. Поскольку информация о конкретных механизмах работы этих систем не всегда доступна, то при их оценке будем опираться только на те сведения, которые опубликованы в печати.

AVISPA

Программный продукт AVISPA появился в начале осени 2005 года. Разработка данного средства проводилась в рамках единого европейского проекта, в котором участвовали LORIA-INRIA (Франция), ETH Цюрих (Швейцария), университет г. Генуя (Италия), Siemens AG (Германия).

Архитектура AVISPA допускает анализ протокола одним из четырех выходных модулей (TA4SP, SATMC, OFMC, CL-AtSe). Спецификация протокола, основанная на ролевом представлении, записывается на языке высокого уровня HLPSL, а затем транслируется во внутренний язык IF. Проверяемые свойства записываются в терминах темпоральной логики. Модуль TA4SP реализует технику, основанную на построении древовидных автоматов и развитую для систем автоматического доказательства. Строится верхняя аппроксимация древовидного автомата, реализующего систему переписывания термов, которая описывает максимальные знания нарушителя. Исследование свойства конфиденциальности теперь сводится к проверке наличия в этом множестве терма, содержащего секрет. Модули SATMC, OFMC, CL-AtSe осуществляют верификацию методом проверки на модели (model checking). Протокол представля-

¹Работа выполнена при поддержке гранта Президента РФ НШ № 4.2008.10.

ется как бесконечная система переходов, а задача верификации сводится к проверке выполнимости формулы, решения которой соответствуют атакам на протокол. Для сведения к конечному случаю применяются разные подходы. В модуле SATMC используются методы, разработанные в рамках теории решения задач планирования, в модуле OFMC — символический метод, позволяющий группировать различные состояния в бесконечные классы, а в CL-AtSe — применяется техника, основанная на построении ограничений.

С помощью AVISPA проанализировано большинство протоколов, встречающихся в документах IETF. Полная информация о разработке и публикациях, лежащих в его основе, а также исполняемый программный код этого средства вместе с удобной графической оболочкой SPAN доступны на интернет-сайте <http://www.avispa-project.org>.

Scyther

Создан в ЕТН (Цюрих). Верифицирует ограниченное и неограниченное число сеансов протокола. Использует символический анализ в сочетании с обратным поиском, основанный на частично упорядоченных шаблонах. Scyther не требует задания сценария атаки. Он требует только задания параметров, ограничивающих либо максимальное число запусков, либо пространства перебираемых траекторий. В первом случае всегда дает результат и показывает найденные траектории атаки. Во втором случае завершение не гарантировано. В качестве ответа возможна одна из трех ситуаций: установлено, что проверяемое свойство выполнено; свойство не выполнено, так как найдена атака; свойство может быть корректно для заданного пространства траекторий.

HERMES

Разработан лабораторией VERIMAG из IMAG (Франция) в рамках объединенного проекта EVA и PROUVE. Использует язык LaEva верхнего уровня для задания спецификаций протоколов и их свойств, аналогичный языку HLPSP. Затем этот язык транслируется во внутреннее представление на языке *cpl*, которое является общим языком для нескольких продуктов, таких, как Securify, Cpv и HERMES.

Позволяет проверять свойства протокола при неограниченной длине сообщения и неограниченном числе участников. Дает на выходе условия на исходные знания противника, которые гарантируют, что он не сможет узнать секрета. HERMES не оценивает знания нарушителя, а исследует множество безопасных сообщений. Для аппроксимации бесконечного множества сообщений используется символическое представление, основанное на шаблонах. Если в результате найдена атака, то предоставляет траекторию атаки. В случае, когда в результате получается ответ, что свойство доказано, выдает также дерево полного доказательства, что бывает полезным при сертификации протокола. Доступен в интерактивном онлайн-режиме на сайте <http://www.rverimag.imag.fr/~async/hermes/>.

ProVerif

Разработан в рамках проекта, финансируемого INRIA (Франция). Анализирует неограниченное число сеансов протокола с использованием верхней аппроксимации и представления протокола с помощью хорновских выражений. ProVerif предлагает два типа входных файлов: хорновские выражения и подмножество Pi-исчисления. При использовании Pi-исчисления ProVerif основывается на описании множества процессов, каждый из которых может выполняться неограниченное число раз. На выходе возможны четыре ситуации: свойство не выполнено; доказано, что свойство выполнено; свойство не может быть доказано, так как есть пример атаки (могут быть найдены ложные атаки); работа не завершается. ProVerif корректно моделирует множество тра-

екторий, соответствующих определенному сценарию, и осуществляет полный перебор возможных траекторий. Исходный код доступен по адресу <http://www.proverif.ens.fr/>.

УДК 519.725

ОБОВЩЕННЫЕ АВТОМОРФИЗМЫ КОДА РИДА–МАЛЛЕРА И КРИПТОСИСТЕМА МАК–ЭЛИСА–СИДЕЛЬНИКОВА

И. В. Чижов

Криптосистема Мак–Элиса–Сидельникова относится к классу кодовых криптосистем с открытым ключом. Криптосистема была предложена В. М. Сидельниковым в работе [1].

Кратко опишем устройство криптосистемы Мак–Элиса–Сидельникова. Пусть R — $(k \times n)$ -порождающая матрица кода Рида–Маллера $RM(r, m)$. Секретным ключом криптосистемы является кортеж $(H_1, H_2, \dots, H_u, \Gamma)$. Здесь $H_1, H_2, \dots, H_u$ — невырожденные $k \times k$ -матрицы над полем $F_2 = \{0, 1\}$, которые выбираются случайно и равновероятно из множества $GL_k(F_2)$ всех двоичных невырожденных $k \times k$ -матриц над полем F_2 . Матрица Γ — перестановочная $(u \cdot n \times u \cdot n)$ -матрица.

Открытым ключом криптосистемы Мак–Элиса–Сидельникова является матрица $G' = (H_1 R \| H_2 R \| \dots \| H_u R) \cdot \Gamma$, где символом $\|$ обозначена конкатенация матриц по столбцам. Алгоритмы шифрования и расшифрования подробно описаны в [1].

Два секретных ключа $(H_1, H_2, \dots, H_u, \Gamma)$ и $(H'_1, H'_2, \dots, H'_u, \Gamma')$ назовём *эквивалентными*, если соответствующие им открытые ключи совпадают, то есть выполняется соотношение $(H_1 R \| H_2 R \| \dots \| H_u R) \cdot \Gamma = (H'_1 R \| H'_2 R \| \dots \| H'_u R) \cdot \Gamma'$.

Рассмотрим множество $\mathcal{G}(H_1, H_2, \dots, H_u)$, состоящее из перестановок $\Gamma \in S_{un}$, для которых существуют невырожденные двоичные матрицы $H'_1, H'_2, \dots, H'_u$, такие, что $(H_1 R \| H_2 R \| \dots \| H_u R) \Gamma = (H'_1 R \| H'_2 R \| \dots \| H'_u R)$. В работе такие множества называются множествами обобщённых автоморфизмов кода Рида–Маллера. Отметим, что эти множества, в отличие от множества обычных автоморфизмов, не всегда являются группами.

Вопрос изучения эквивалентных секретных ключей, а значит, и вопрос изучения множества открытых ключей, сводится к изучению множеств $\mathcal{G}(H_1, \dots, H_u)$, то есть обобщённых автоморфизмов.

Обобщённые автоморфизмы и структура множества открытых ключей могут оказаться полезными для криптоанализа криптосистемы Мак–Элиса–Сидельникова. Так, знание некоторой структуры группы автоморфизмов обобщённых кодов Рида–Соломона позволило В. М. Сидельникову и С. О. Шестакову [2] произвести взлом криптосистемы Мак–Элиса на основе этих кодов.

Перейдём к описанию множеств обобщённых автоморфизмов.

В случае произвольного u справедлива следующая теорема.

Теорема 1. Пусть для невырожденных матриц $D_1, D_2, \dots, D_u$ существуют такие перестановки P_i ($1 \leq i \leq n$) из S_n , что $D_1 R = R \cdot P_1$, $D_2 R = R \cdot P_2$, $\dots$, $D_u R = R \cdot P_u$. Обозначим через $\mathcal{P}_1[1], \mathcal{P}_2[2], \dots, \mathcal{P}_u[u]$ перестановки из $\mathcal{A}_u(RM(r, m))$, соответствующие перестановкам $P_1, P_2, \dots, P_u$. И пусть H — любая невырожденная матрица.

Тогда $\mathcal{G}(E, \dots, E) = \mathcal{P}_1[1] \cdot \mathcal{P}_2[2] \cdot \dots \cdot \mathcal{P}_u[u] \cdot \mathcal{G}(H D_1, \dots, H D_u)$.

Описание множества $\mathcal{G}(E, \dots, E)$ получено Г. А. Карпуниным [3].

Рассмотрим теперь случай $u = 2$.