Теорема 1. Марковский процесс имеет единственное финальное распределение вероятностей, и на выходе каждого блока сигнал имеет равномерное распределение.

Даются оценки интервала времени между двумя съёмами сигналов, обеспечивающего независимость этих сигналов.

Подробное изложение представленных результатов можно найти в [4].

ЛИТЕРАТУРА

1. *Sheng L., Yong-Bin K., and Lombardi F.* CNTFET-Based Design of Ternary Logic Gates and Arithmetic Circuits // IEEE Trans. Nanotechnology. 2011. V. 10. No. 2. P. 217–225.
2. *Кузнецов В. М., Песошин В. А., Столов Е. Л.* Марковская модель цифрового стохастического генератора // АиТ. 2008. № 9. С. 62–68.
3. *Хинчин А. Я.* Работы по математической теории массового обслуживания. М.: Физматгиз, 1963. 236 с.
4. *Столов Е. Л.* Математическая модель генератора случайных чисел на основе трёхзначной логики // Прикладная дискретная математика. 2012. № 2(12). С. 43–49.

УДК 519.7

ОДНОРАЗОВАЯ КОЛЬЦЕВАЯ ПОДПИСЬ И ЕЁ ПРИМЕНЕНИЕ В ЭЛЕКТРОННОЙ НАЛИЧНОСТИ

Г. О. Чикишев

Кольцевая подпись (*ring signature*) [1] позволяет участнику группы подписывать сообщения от имени всей группы (указывая для проверки вместо своего открытого ключа ключи всех участников группы). Проверяющий уверен, что использован один из секретных ключей, но чей именно — он не знает. В работе мы вводим новый вид кольцевой подписи, наделяя её свойством одноразовости: в случае повторного использования одного и того же секретного ключа личность его автора будет раскрыта (иными словами, анонимно подписать можно лишь одно сообщение).

Такое свойство востребовано во многих областях: электронные выборы (каждый участник может проголосовать только один раз), цифровые деньги (электронные монеты можно потратить лишь единожды) и т. д. Продемонстрируем применение алгоритма в сфере открытых электронных транзакций на примере децентрализованной p2p-валюты Bitcoin [2]. Это решение позволяет участнику совершать полностью неотслеживаемый платёж (что на данный момент невозможно в Bitcoin), открыто публикуя детали операций по переводу и получению средств.

Алгоритм одноразовой кольцевой подписи состоит из нескольких этапов:

— **GEN** — генерация ключей. Каждый участник готовит два закрытых ключа $0 < x, y < N$ и публикует два открытых ключа $P_x = xQ_x$ и $P_y = yQ_y$ электронной подписи ECDSA.

— **RING-SIG** — создание кольца подписей. Алиса готовит $n-1$ «подделку» чужих подписей ECDSA (m_i, A_i, β_i), применяя технику «2-parameter forgery» [3]:

$$A_i = u_i Q_x + v_i P_{x_i}; \quad \beta_i = -\mathcal{H}(A_i) \cdot v_i^{-1} \bmod N; \quad m_i = u_i \cdot \beta_i \bmod N,$$

и выбирает такое m_{i_s}, чтобы все m_i рекуррентно замкнулись в кольцо

$$z_{i_0} = \mathcal{H}\left(m, m_{i_0+n-1} \oplus \mathcal{H}(m, m_{i_0+n-2} \oplus \mathcal{H}(m, \cdots \oplus \mathcal{H}(m, m_{i_0} \oplus z_{i_0}) \cdots))\right) \qquad (1)$$

для произвольно выбранного $i_0 \in \{1, \ldots, n\}$.

После этого Алиса использует оба секретных ключа (x и y) для создания одной настоящей подписи $(m_{i_s}, A_{i_s}, \beta_{i_s})$: $(A_{i_s}, \beta_{i_s}) = (yQ_x, (m_{i_s} - \mathcal{H}(A_{i_s})x)y^{-1} \bmod N)$.
Теперь проверяющему достаточно проверить все n подписей ECDSA обычным образом, а также уравнение (1).

— **NIZK-SIG** — кроме кольца подписей, Алиса конструирует неинтерактивное доказательство эквивалентности двух «логарифмов»: $A_{i_s} = y_s Q_x$ (набор из «половинок» подписей ECDSA) и $P_{y_s} = y_s Q_y$ (набор из вторых открытых ключей). Доказательство обладает свойством нулевого разглашения в том смысле, что индекс Алисы i_s остаётся в секрете.
Алиса генерирует случайные пары (q_i, w_i) из $\{1, \ldots, N-1\}^2$ и вычисляет

$$L_i = \begin{cases} q_i Q_x, & \text{если } i = s, \\ q_i Q_x + w_i A_i, & \text{если } i \neq s; \end{cases} \qquad R_i = \begin{cases} q_i Q_y, & \text{если } i = s, \\ q_i Q_y + w_i P_{y_i}, & \text{если } i \neq s. \end{cases}$$

После этого Алиса вычисляет хэш $c = \mathcal{H}(L_1, \ldots, L_n, R_1, \ldots, R_n, m)$ и конструирует подпись-доказательство

$$c_i = \begin{cases} w_i, & \text{если } i \neq s, \\ c - \sum\limits_{i=1}^{n} c_i \bmod N, & \text{если } i = s; \end{cases} \qquad r_i = \begin{cases} q_i, & \text{если } i \neq s, \\ q_s - c_s y \bmod N, & \text{если } i = s. \end{cases}$$

Доказательством, являющимся частью подписи, служит набор пар (c_i, r_i).

— **VER** — проверка подписи включает в себя проверку следующих равенств:

$$A_i = (m_i \beta_i^{-1}) Q_x + (\mathcal{H}(A_i)\beta_i^{-1}) P_{x_i},$$
$$z_{i_0} = \mathcal{H}\left(m, m_{i_0+n-1} \oplus \mathcal{H}(m, m_{i_0+n-2} \oplus \mathcal{H}(m, \cdots \oplus \mathcal{H}(m, m_{i_0} \oplus z_{i_0}) \cdots))\right),$$
$$\sum_{i=1}^{n} c_i = \mathcal{H}(L'_1, \ldots, L'_n, R'_1, \ldots, R'_n, m) \bmod N,$$

где $L'_i = r_i Q_x + c_i A_i$; $R'_i = r_i Q_y + c_i P_{y_i}$

Для того чтобы проверка доказательства была успешной, Алиса должна использовать свой второй закрытый ключ на этапе **RING-SIG** при создании своей подписи $(m_{i_s}, A_{i_s}, \beta_{i_s})$. Поэтому повторное его применение приведет к появлению одинаковых значений $A_{i_s} = y_s Q_x$ в разных кольцевых подписях, что укажет на соответствующую пару открытых ключей и приведёт к идентификации Алисы.

Используя одноразовую кольцевую подпись, можно совершать полностью неотслеживаемые платежи в системе электронных денег Bitcoin. Транзакции Bitcoin хранятся в публичной децентрализованной базе данных и явно содержат адреса отправителей и получателей. Совершая перевод в два этапа, можно достигнуть полной анонимности:

1) **Перевод:** отправитель Алиса совершает платёж на «системный кошелёк», регистрируя оба публичных ключа, полученных на этапе **GEN**, для подписи анонимного чека. Так же поступают и другие пользователи, поэтому в любой конкретный момент в текущей базе одновременно находится множество ключей.
2) **Получение:** Боб получает от Алисы чек (сообщение, подписанное на этапах **RING-SIG** и **NIZK-SIG**) и публикует его вместе с транзакцией по переводу денег с системного кошелька на свой счёт.

Все возможные отправители, которые могли подписать чек, равновероятны, поэтому платёж между Алисой и Бобой является неотслеживаемым.

ЛИТЕРАТУРА

1. *Rivest R., Shamir A., and Tauman Y.* How to Leak a Secret // Proc. 7th Internat. Conf. on the Theory and Application of Cryptology and Information Security: Advances in Cryptology, 2001. P. 552–565.
2. *Nakamoto S.* Bitcoin: A Peer-to-Peer Electronic Cash System. `http://bitcoin.org/bitcoin.pdf`. 2009.
3. *Rivest R., Micali S., and Goldwasser S.* A Digital Signature Scheme Secure against Adaptive Chosen-Message Attacks // SIAM J. Computing. 1988. V. 17. No. 2. P. 281–308.
4. *Cramer R., Damgard I., and Schoenmakers B.* Proofs of partial knowledge and simplified design of witness hiding protocols // LNCS. 1994. V. 839. P. 174–187.

УДК 519.7

ОПТИМАЛЬНЫЕ ЛИНЕЙНЫЕ ПРИБЛИЖЕНИЯ СЕТЕЙ ФЕЙСТЕЛЯ. ОЦЕНКА СТОЙКОСТИ ШИФРА SMS4 К ЛИНЕЙНОМУ КРИПТОАНАЛИЗУ

Г. И. Шушуев

Для применения линейного криптоанализа к итеративному блочному шифру требуется найти линейное приближение для конкретного числа раундов. Если известно лучшее приближение, то можно определить минимальную трудоёмкость линейного криптоанализа шифра, то есть оценить его криптографическую стойкость. Таким образом, задача оценки криптографической стойкости сводится к поиску линейных приближений и нахождению среди них лучшего. В связи с этим возникает ряд проблем, так как, помимо того, что нужно исследовать раундовую функцию, необходимо правильно согласовывать приближения раундов. Как правило, при проведении линейного криптоанализа выбирается некоторое найденное линейное приближение без доказательства того, что оно является лучшим.

Для поиска лучшего приближения можно перебирать всевозможные линейные соотношения, но на это может потребоваться больше времени, чем на составление словаря, поэтому необходимо придумывать другие способы. Предлагается подход к нахождению линейных приближений сети Фейстеля и поиску оптимального.

Определение 1. *Раундовая функция* — функция вида $F : (\mathbb{Z}_2^m)^n \to \mathbb{Z}_2^m$. Пусть $F(X_1, X_2, \ldots, X_n) = Y$, где $Y, X_i \in \mathbb{Z}_2^m, i = 1, 2, \ldots, n$.

Как правило, раундовая функция сети Фейстеля удовлетворяет некоторым ограничениям, например, является простой (применяется в SMS4) или псевдо-простой (DES, TEA). Функция F называется *простой*, если $F(X_1, X_2, \ldots, X_n) = G(X_1 \oplus X_2 \oplus \ldots \oplus X_n)$ для некоторой функции $G : \mathbb{Z}_2^m \to \mathbb{Z}_2^m$. Функция F называется *псевдопростой*, если $F(X_1, X_2, \ldots, X_n) = G(L_1(X_1) \oplus L_2(X_2) \oplus \ldots \oplus L_n(X_n))$ для некоторых функции $G : \mathbb{Z}_2^m \to \mathbb{Z}_2^m$ и набора функций $L_1, \ldots, L_n$, где $L_i : \mathbb{Z}_2^m \to \mathbb{Z}_2^m$.

Определение 2. *Линейным приближением* функции F называется соотношение

$$b_1 \cdot X_1 \oplus b_2 \cdot X_2 \oplus \ldots \oplus b_n \cdot X_n = a \cdot F(X_1, \ldots, X_n), \tag{1}$$

выполняющееся с вероятностью $1/2 + \varepsilon$, где $|\varepsilon| \leqslant 1/2$, $a, b_i \in \mathbb{Z}_2^m$, $i = 1, 2, \ldots, n$.

Величину ε назовём *линейным преобладанием соотношения* или просто *преобладанием*. Вектор b_i назовём *маской* для вектора X_i. Определим функцию