Секция 3

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

УДК 511.48 DOI 10.17223/2226308X/13/14

ПРОЕКТ СТАНДАРТИЗАЦИИ ПОСТКВАНТОВОЙ ЦИФРОВОЙ ПОДПИСИ

Е. А. Киршанова, Н. С. Колесников, Е. С. Малыгина, С. А. Новоселов

Предлагается цифровая подпись, безопасность которой основана на задачах MLWR и MSIS в алгебраических решётках. Конструкция подписи основана на парадигме Фиата — Шамира. Доказывается безопасность схемы в квантовой модели безопасности и описываются конкретные параметры, при которых схема достигает уровня безопасности в 100 бит. Благодаря модульной структуре решёток, уровень безопасности легко изменить в большую или меньшую стороны. Наше предложение может служить основой проекта по стандартизации постквантовых примитивов на решётках.

Ключевые слова: *цифровая подпись, криптография на решётках, постквантовая криптография, парадигма Фиата — Шамира.*

Введение

Криптографические примитивы на решётках — одно из самых обещающих направлений современной криптографии не только ввиду стойкости этих примитивов к атакам на квантовом компьютере, но и вследствие большого спектра конструкций (гомоморфное шифрование, электронные голосования, различные типы подписей), а также их надёжности по отношению к *классическим* атакам. Криптографические конструкции на решётках не только элегантны в теории, но и значимы на практике, поэтому в достаточно скором будущем будут стандартизированы. Пробные версии обмена ключами New Hope уже тестированы в TLS-соединениях для браузера Google Chrome [1]. Процесс стандартизации постквантовых схем доступен по адресу `https://csrc.nist.gov/projects/post-quantum-cryptography`.

В этой работе мы предлагаем схему цифровой подписи, основанную на алгебраических решётках, конструкция которой удовлетворяет следующими основным свойствам:

1) безопасность схемы основана на задачах «в среднем», а именно на задачах LWR (Learning With Rounding) и SIS (Shortest Integer Solution) — классических трудных задачах на решётках, определения которых даны в п. 1);
2) для эффективности схемы используется так называемый *модульный* вариант задач, а именно module-LWR, module-SIS [2], что не только позволяет уменьшить размеры параметров схемы и время операций, но и даёт возможность легко варьировать уровни безопасности схемы;
3) стойкость схемы доказана в квантовой модели QROM (Quantum Random Oracle Model) для «сильного» атакующего, а именно для атаки вида $\mathsf{UF}-\mathsf{sCMA}$; доказательство можно найти в [3];

4) в процессе генерации ключей и подписи вместо нормального распределения используется равномерное распределение из интервала, что уменьшает риск сторонних атак;
5) предлагается конкретный набор параметров схемы с битовой оценкой сложности атак на предложенные параметры (см. п. 3).

Представленная здесь схема основана на парадигме Фиата — Шамира [4, 5] и по идеологии продолжает серию работ, предлагающих конкретные схемы подписи [6 – 8]. Основное отличие нашей схемы от ранее предложенных заключается в том, что безопасность ключей основана на задаче LWR (а не на задаче LWE (Learning With Errors)). Мы считаем, что такой подход упрощает описание и потенциально ускоряет вычисления.

1. Предварительные сведения

1.1. Обозначения

Будем обозначать $\mathbb{Z}/q\mathbb{Z}$ кольцо целых по чётному модулю q, результат $z \bmod q$ представляем в интервале $\{0, \ldots, q-1\}$; R, R_q и R_p — кольца многочленов $\mathbb{Z}[x]/(x^n+1)$, $\mathbb{Z}/q\mathbb{Z}[x]/(x^n+1)$ и $\mathbb{Z}/p\mathbb{Z}[x]/(x^n+1)$ соответственно. Векторы будем обозначать жирными строчными буквами (например, $\mathbf{x}$), матрицы — прописными (например, $\mathbf{A}$), константы — обычными строчными; $\mathbb{I}$ — единичная матрица. Элементы кольца $\mathbb{Z}[x]/(x^n+1)$ будем понимать как векторы-коэффициенты многочленов. Векторы по умолчанию являются вектор-столбцами. Евклидова (или ℓ_2) норма вектора $\mathbf{x}$ определяется как $\|\mathbf{x}\| = \|\mathbf{x}\|_2 = \sqrt{\sum_i x_i^2}$, а ℓ_∞-норма — как $\|\mathbf{x}\|_\infty = \max_i |x_i|$.

Многочленам из кольца R ставим в соответствие векторы-коэффициенты длины n, поэтому произведение векторов $\mathbf{x} \cdot \mathbf{y}$ надо понимать как произведение соответствующих многочленов. Элементу $\mathbf{a} \in R_q$ ставим в соответствие матрицу $\mathrm{rot}(\mathbf{a}) \in (\mathbb{Z}/q\mathbb{Z})^{n\times n}$, i-я строка которой — коэффициенты многочлена $x^{i-1} \cdot \mathbf{a}$. Такая матрица задаёт произведение любого элемента из R_q на многочлен $\mathbf{a}$.

Для конечного множества S запись $s \leftarrow S$ обозначает, что s выбрано в соответствии со случайным равномерным распределением на S. Через S_β^ℓ обозначим множество векторов длины ℓ, каждый коэффициент которого взят в соответствии с равномерным распределением из множества $\{-\beta, \ldots, \beta\}$.

Для любого $x \in \mathbb{Q}$ запись $\mathsf{Round}(x) \in \mathbb{Z}$ означает взятие ближайшего целого, где $1/2$ округляется до 1. Для целого x функция $\mathsf{MSB}(x, d)$ (соответственно $\mathsf{LSB}(x, d)$) означает взятие d старших (соответственно младших) бит. Все операции распространяются на векторы и матрицы покоэффициентно.

В нашей схеме мы будет использовать два модуля: $q = 2^\nu$ и $p = 2^\mu$. «Конвертирование» элемента $x \in \mathbb{Z}/q\mathbb{Z}$ в $x' \in \mathbb{Z}/p\mathbb{Z}$ происходит по правилу $x' = \mathsf{Round}(x \cdot p/q)$. Так как модули — степени двойки, этот же результат можно получить, добавив к x константу $h = 2^{\nu-\mu-1}$ и взяв μ старших бит: $x' = \mathsf{MSB}(x + h, \mu)$. Такое представление операции Round использовано, например, в [9]. Вектор, каждая координата которого равна h, обозначим $\mathbf{h}$. Для всякого целого $w > 0$ положим $B_w = \{\mathbf{x} \in R : \|\mathbf{x}\|_\infty = 1, \|\mathbf{x}\| = \sqrt{w}\} \subseteq R$.

1.2. Синтаксис и модели безопасности цифровых подписей

Определение 1. Цифровая подпись — примитив, состоящий из трёх алгоритмов:

— вероятностный алгоритм генерации ключевой пары $\mathsf{KeyGen}(\mathsf{par})$, возвращающий секретный ключ sk и ключ верификации vk;

— вероятностный алгоритм генерации подписи $\mathsf{Sign}(\mathsf{sk}, m)$, который для сообщения $m \in \mathcal{M}$ возвращает подпись σ;
— детерминированный алгоритм $\mathsf{Verify}(m, \sigma, \mathsf{vk})$, который возвращает либо «Accept» (подпись σ корректна для (m, vk)), либо «Reject» (подпись σ не корректна для (m, vk)).

Цифровая подпись корректна с долей ошибки ε, если для всех пар $(\mathsf{sk}, \mathsf{vk}) \in \in \mathsf{KeyGen}(\mathsf{par})$ и всех сообщений $m \in \mathcal{M}$ имеем

$$\mathsf{P}\left[\mathsf{Verify}(m, \mathsf{Sign}(\mathsf{sk}, m), \mathsf{vk}) = \text{«Accept»}\right] \geqslant 1 - \varepsilon.$$

1.3. Сложные задачи на решётках

Безопасность нашей подписи основывается на двух «сложных в среднем» задачах. Первая — задача Обучения с Округлением (Learning With Rounding (LWR)) [10] — детерминированная версия задачи Обучения с Ошибками (Learning With Errors (LWE)) [11]. В основе безопасности ключей подписи лежит трудность этой модульной версии задачи над фактор-кольцом R_q [2]. Все вычисления производятся в фактор-кольце R_q, матрица $\mathbf{A}$ формируется как блочная матрица из $k \cdot \ell$ элементов из R_q, где каждый блок — матрица $\mathrm{rot}(\mathbf{a})$.

Для предлагаемой схемы, в отличие от классических задач LWR и LWE, где матрица $\mathbf{A}$ берётся случайным образом из $R_q^{k \times \ell}$, будем требовать, чтобы хотя бы один из $k \cdot \ell$ многочленов был обратим в R_q. Будем обозначать такую матрицу через $\widetilde{\mathbf{A}}$. Это требование не влияет на безопасность схемы, поскольку, как минимум, константное число многочленов обратимы в R_q[1]. Значит, если атакующий имеет непренебрежимо малую вероятность успеха для $\widetilde{\mathbf{A}}$, этот же атакующий имеет непренебрежимо малую вероятность успеха для $\mathbf{A} \leftarrow R_q$.

Определение 2 (*задача обучения с округлением* (*MLWR*)). Пусть $q \geqslant p \geqslant 1$, $k, \ell \geqslant 1$ — целые числа. MLWR-распределение для вектора $\mathbf{s} \leftarrow R_q^{\ell}$ есть множество пар вида $\left(\mathbf{A}, \mathsf{Round}\left(\frac{p}{q} \cdot \mathbf{A} \cdot \mathbf{s}\right)\right)$, где $\widetilde{\mathbf{A}} \leftarrow R_q^{k \times \ell}$. *Задача поиска*: для заданного произвольным образом большого числа выборок из MLWR-распределения для вектора $\mathbf{s} \leftarrow R_q^{\ell}$ восстановить $\mathbf{s}$. *Задача различения распределений*: для заданного произвольным образом большого числа выборок из $\widetilde{R}_q^{k \times \ell} \times R_p^k$ определить, являются ли они равномерно распределёнными или MLWR-распределёнными для вектора $\mathbf{s} \leftarrow R_q^{\ell}$.

Обе версии задачи эквивалентны (то есть, имея оракул, решающий одну задачу, можно решить другую за полиномиальное от n время) [12]. В доказательстве безопасности схемы подписи нам понадобится вторая версия. Безопасность подписи основана на задаче нахождения Короткого Целочисленного Решения (Short Integer Solution (SIS) problem) [13]. Нам потребуется модульная версия этой задачи.

Определение 3 (*задача нахождения Короткого Целочисленного Решения* (MSIS)). Зафиксируем $b \in \mathbb{N}$ и пусть $\mathbf{A} \leftarrow R_q^{k \times \ell}$. Модульная задача нахождения короткого целочисленного решения, параметризованная посредством $b > 0$, заключается в нахождении «короткого» ненулевого прообраза $\mathbf{y} \leftarrow R_q^{k+\ell}$ в решётке, определяемой $\mathbf{A}$, т. е.

$$\mathbf{y} \neq 0, \quad [\mathbb{I} | \mathbf{A}] \cdot \mathbf{y} = 0 \quad \text{и} \quad \|\mathbf{y}\|_{\infty} \leqslant b.$$

[1] Вероятность обратимости случайного многочлена в R_q, где q — степень двойки, не столь тривиальна (и не столь велика), как в случае простого q. Случайный многочлен $\mathbf{a} \in R_q$ обратим тогда и только тогда, когда $\mathrm{rot}(\mathbf{a})$ — обратимая матрица в $\mathbb{Z}/q\mathbb{Z}^{n \times n}$, что, в свою очередь, верно тогда и только тогда, когда $\det(\mathrm{rot}(\mathbf{a}))$ — обратимый элемент в $\mathbb{Z}/q\mathbb{Z}$. В случае $q = 2^{\nu}$ имеем $|\mathbb{Z}_q^*| = 2^{\nu-1}$, а значит, случайный элемент из $\mathbb{Z}/q\mathbb{Z}$ обратим с вероятностью $|\mathbb{Z}_q^*|/q = 1/2$.

Для доказательства безопасности схемы потребуется вариант задачи SIS, так называемый SelfTargetSIS, предложенный в [14]. В этой же работе описана редукция от SIS к SelfTargetSIS.

Определение 4 (*задача* SelfTargetSIS). Пусть $\mathcal{H} : \{0,1\}^* \to B_w$ — криптографическая хэш-функция. Зададим случайным образом $\mathbf{A} \leftarrow R_q^{k\times\ell}$ и доступ к квантовому случайному оракулу $\mathcal{H}(\cdot)$. Для исходного сообщения $M \in \{0,1\}^*$ задача SelfTargetSIS сводится к нахождению

$$\mathbf{y} = [\mathbf{r}, \mathbf{c}]^T, \quad \text{где} \quad 0 \leqslant \|\mathbf{y}\|_\infty \leqslant \gamma, \quad \mathcal{H}([\mathbf{A}|\mathbb{I}] \cdot \mathbf{y}, M) = \mathbf{c}.$$

2. Описание схемы

Цифровая подпись (алгоритмы 1–3) зависит от следующих параметров: $q = 2^\nu$, $p = 2^\mu$, $\nu > \mu$. Используется криптографическая хэш-функция $\mathcal{H} : \{0,1\}^* \to B_w$ [7]. Параметры k, ℓ отвечают за размерности ключей; s, γ задают интервалы для коэффициентов многочленов в процессе генерации ключей или подписи; d, β отвечают за корректность и безопасность схемы. Подпись формируется для сообщений $M \in \{0,1\}^*$. Конкретные значения параметров заданы в п. 3.

Алгоритм 1. Генерация ключей

Вход: $\ell > k > 1$, $q > p$, s.
Выход: $\mathbf{A}$, $\mathbf{t}$.
1: $\mathbf{A} \leftarrow R_q^{k\times\ell}$;
2: $\mathbf{s} \leftarrow S_s^\ell$;
3: $\mathbf{t} := \mathsf{Round}\left(\frac{p}{q} \cdot \mathbf{As}\right)$. // $\|\mathbf{t} - \mathbf{As}\|_\infty \leqslant 2^{\nu-\mu}$
4: **Вернуть** $\mathsf{sk} = \mathbf{s}$, $\mathsf{vk} = (\mathbf{A}, \mathbf{t})$.

Алгоритм 2. Генерация подписи

Вход: $q = 2^\nu$, $p = 2^\mu$, $\ell > 1$, M, $\mathbf{A}$, $\mathbf{t}$, $\mathbf{s}$, d, $\mathcal{H}$, β, γ, w.
Выход: $(\mathbf{z}, \mathbf{c})$.
1: $\mathbf{y} \leftarrow S_{\gamma-1}^\ell$;
2: $\mathbf{c} := \mathcal{H}\left(\mathsf{MSB}(\mathbf{A} \cdot \mathbf{y}, d), M\right)$;
3: $\mathbf{z} := \mathbf{y} + \mathbf{sc}$;
4: $\mathbf{w} := \mathbf{Az} - \mathbf{t} \cdot 2^{\nu-\mu} \cdot \mathbf{c}$;
5: **Если** $(\|\mathsf{LSB}(\mathbf{w}, \nu - d)\|_\infty \geqslant 2^{\nu-d} - w \cdot 2^{\nu-\mu+1})$ или $(\|\mathbf{z}\|_\infty \geqslant \gamma - \beta)$, **то** **restart**.
6: **Вернуть** $(\mathbf{z}, \mathbf{c})$.

Алгоритм 3. Проверка подписи

Вход: M, $\mathbf{z}$, $\mathbf{c}$, $\mathbf{A}$, $\mathbf{t}$, d, $\mathcal{H}$, β, γ.
Выход: «Accept» или «Reject».
1: $\mathbf{w} := \mathbf{A}\mathbf{z} - \mathbf{t} \cdot 2^{\nu-\mu} \cdot \mathbf{c}$;
2: $\mathbf{c}' := \mathcal{H}(\mathsf{MSB}(\mathbf{w}, d)), M)$;
3: **Если** $\mathbf{c}' = \mathbf{c}$ и $\|\mathbf{z}\|_\infty \leqslant \gamma - \beta$, **то**
4: **Вернуть** «Accept»,
5: **иначе**
6: **Вернуть** «Reject».

2.1. Корректность

Поскольку $\mathbf{w} = \mathbf{A} \cdot \mathbf{z} - \mathbf{t} \cdot 2^{\nu-\mu} \cdot \mathbf{c}$, $\mathbf{z} = \mathbf{y} + \mathbf{s} \cdot \mathbf{c}$ и $\mathbf{t} = \mathsf{Round}\left(\frac{p}{q} \cdot \mathbf{A}\mathbf{s}\right)$, то

$$\mathbf{w} = \mathbf{A} \cdot (\mathbf{y} + \mathbf{s} \cdot \mathbf{c}) - \mathbf{c} \cdot 2^{\nu-\mu} \cdot \mathsf{Round}\left(\frac{p}{q} \cdot \mathbf{A}\mathbf{s}\right) = \mathbf{A}\mathbf{y} + \mathbf{A}\mathbf{s}\mathbf{c} - \mathbf{c} \cdot 2^{\nu-\mu} \cdot \mathsf{Round}\left(\frac{p}{q} \cdot \mathbf{A}\mathbf{s}\right).$$

Согласно введённым обозначениям, $\mathsf{Round}\left(\frac{p}{q} \cdot \mathbf{A}\mathbf{s}\right) = \mathsf{MSB}(\mathbf{A}\mathbf{s} + \mathbf{h}, \mu)$, где $\mathbf{h}$ — вектор, каждая координата которого равна $h = 2^{\nu-\mu-1}$. Тогда

$$\mathbf{w} = \mathbf{A}\mathbf{y} + \mathbf{A}\mathbf{s}\mathbf{c} - \mathbf{c} \cdot 2^{\nu-\mu} \cdot \mathsf{MSB}(\mathbf{A}\mathbf{s} + \mathbf{h}, \mu) = \mathbf{A}\mathbf{y} + \mathbf{A}\mathbf{s}\mathbf{c} - \mathbf{c}\left(\mathbf{A}\mathbf{s} + \mathbf{h} + \mathsf{LSB}(\mathbf{A}\mathbf{s} + \mathbf{h}, \nu - \mu)\right).$$

Раскрывая скобки, окончательно получаем

$$\mathbf{w} = \mathbf{A}\mathbf{y} - \mathbf{c}\left(\mathbf{h} + \mathsf{LSB}(\mathbf{A}\mathbf{s} + \mathbf{h}, \nu - \mu)\right), \tag{1}$$

где $\|\mathbf{c}\left(\mathbf{h} + \mathsf{LSB}(\mathbf{A}\mathbf{s} + \mathbf{h}, \nu - \mu)\right)\|_\infty < w \cdot 2^{\nu-\mu+1}$, поскольку $\mathbf{c} \in B_w$ и $\|\mathsf{LSB}(\mathbf{A}\mathbf{s}, \nu - \mu)\|_\infty < 2^{\nu-\mu}$. Рассматривая $\mathsf{LSB}(\mathbf{w}, \nu - d)$ в алгоритме 2 на шаге 5 и учитывая ошибку $\mathbf{c}\left(\mathbf{h} + \mathsf{LSB}(\mathbf{A}\mathbf{s} + \mathbf{h}, \nu - \mu)\right)$, получаем, что при $\|\mathsf{LSB}(\mathbf{w}, \nu - d)\|_\infty > 2^{\nu-d} - w \cdot 2^{\nu-\mu+1}$ алгоритм отклоняет значение $\mathbf{w}$.

Так как $\mathbf{c}\left(\mathbf{h} + \mathsf{LSB}(\mathbf{A}\mathbf{s} + \mathbf{h}, \nu - \mu)\right)$ — малый вектор ошибки, то из равенства (1) очевидно, что $\mathsf{MSB}(\mathbf{w}, d) = \mathsf{MSB}(\mathbf{A}\mathbf{y}, d)$. Следовательно, вычисление $\mathbf{c}'$ на шаге 2 алгоритма 3 совпадает со значением вектора $\mathbf{c}$ на шаге 2 алгоритма 2.

2.2. О числе итераций в процедуре Sign

В процессе вычисления подписи алгоритм 2 на шаге 5 проверяет, попадают ли коэффициенты вектора $\mathbf{z}$ в интервал $\{-(\gamma - \beta - 1), \ldots, \gamma - \beta - 1\}$. Для фиксированного ключа $\mathbf{s}$ вероятность этого события зависит от $\|\mathbf{y}\|_\infty$, выбранного на шаге 1. Вычислим эту вероятность.

Пусть $\mathbf{z} = \mathbf{y} + \mathbf{v}$ такой, что $\mathbf{z} \in S^\ell_{\gamma-\beta-1}$. Обозначим $\beta = \|\mathbf{c}\mathbf{s}\|_\infty$. Так как $\|\mathbf{s}\|_\infty \leqslant s$ и $\mathbf{c} \in B_w$, то $\beta < ws$. Отсюда $\|\mathbf{v}\|_\infty \leqslant \beta$. Для каждого коэффициента $\mathbf{v}_i$ вектора $\mathbf{v}$ соответствующий коэффициент $\mathbf{z}_i$ лежит в интервале $\{-(\gamma - \beta - 1), \ldots, \gamma - \beta - 1\}$. Поскольку $\mathbf{y} = \mathbf{z} - \mathbf{v}$, то $\mathbf{y} \in S^\ell_{\gamma-1}$ и соответствующий коэффициент $\mathbf{y}_i$ лежит в интервале $\{-(\gamma - 1), \ldots, \gamma - 1\}$. Следовательно,

$$p_1 = \mathsf{P}_{\mathbf{y} \leftarrow S^\ell_{\gamma-1}}\left[\|\mathbf{z}\|_\infty < \gamma - \beta\right] = \frac{\left|S^\ell_{\gamma-\beta-1}\right|}{\left|S^\ell_{\gamma-1}\right|} = \left(\frac{2\gamma - 2\beta - 1}{2\gamma - 1}\right)^{n\ell} = \left(1 - \frac{\beta}{\gamma - 1/2}\right)^{n\ell} \approx \exp\left(-\frac{\beta n \ell}{\gamma}\right).$$

Алгоритм 2 на шаге 5 также проверяет, когда коэффициенты вектора $\mathsf{LSB}(\mathbf{w}, \nu - d)$ не попадают в интервал $\{-(2^{\nu-d} - w \cdot 2^{\nu-\mu+1} - 1), \ldots, 2^{\nu-d} - w \cdot 2^{\nu-\mu+1} - 1\}$. Вероятность этого события, очевидно, зависит от малого вектора ошибки $\mathbf{c}\,(\mathbf{h} + \mathsf{LSB}(\mathbf{As} + \mathbf{h}, \nu - \mu))$, который возникает при упрощении выражения $\mathbf{w} = \mathbf{A} \cdot \mathbf{z} - \mathbf{t} \cdot 2^{\nu-\mu} \cdot \mathbf{c}$ на шаге 4. Вычислим эту вероятность.

Как показано выше, каждый коэффициент вектора ошибки $\mathbf{c}\,(\mathbf{h} + \mathsf{LSB}(\mathbf{As}+\mathbf{h}, \nu-\mu))$ лежит в интервале $\{-(w \cdot 2^{\nu-\mu+1} - 1), \ldots, w \cdot 2^{\nu-\mu+1} - 1\}$. Для каждого такого коэффициента соответствующий коэффициент вектора $\mathsf{LSB}(\mathbf{w}, \nu - d)$ попадает в интервал $\{-(2^{\nu-d} - 1), \ldots, 2^{\nu-d} - 1\}$. Учитывая (эвристически) равномерный характер распределений, в итоге получаем

$$p_2 = \mathsf{P}_{\mathbf{w} \in S^k_{2^{\nu-d}-1}} \left[\|\mathsf{LSB}(\mathbf{w}, \nu - d)\|_\infty < 2^{\nu-d} - w \cdot 2^{\nu-\mu+1} \right] = \left(\frac{2^{\nu-d+1} - w \cdot 2^{\nu-\mu+2} - 1}{2^{\nu-d+1} - 1} \right)^{nk} =$$

$$= \left(1 - \frac{w 2^{\nu-\mu+2}}{2^{\nu-d+1} - 1} \right)^{n \cdot k} \approx \exp\left(-nk \, \frac{w 2^{\nu-\mu+2}}{2^{\nu-d+1} - 1} \right).$$

Таким образом, ожидаемое число повторений функции Sign алгоритма 2 равно

$$\mathbb{E}[\#\text{итераций}] = (p_1 \cdot p_2)^{-1}.$$

3. Атаки и выбор параметров

Безопасность нашей схемы подписи основана на двух классических задачах на решётках — MLWR и MSIS. Будем определять конкретные параметры схемы, основываясь на сложности атак на эти задачи.

Мы работаем с модульными решётками, определёнными над кольцом целых циклотомического расширения, а именно $R = \mathbb{Z}[x]/(x^{256} + 1)$, то есть выбираем $n = 256$. Такое n позволяет осуществлять быструю арифметику в R. Основные параметры, определяющие сложность задач MLWR и MSIS, — это k (задаёт ранг решёток) и ℓ (задаёт размер секретного вектора $\mathbf{s}$).

Решение задачи MLWR сводится к нахождению короткого вектора в q-арной решётке ранга d

$$\Lambda_{\text{MLWR}} = \{\mathbf{x} \in \mathbb{Z}^d : [\text{rot}(\mathbf{A}) \,|\, \mathbb{I} \,|\, \mathbf{t}]\, \mathbf{x} = \mathbf{0} \bmod q\},$$

где $d \leqslant n(\ell + k) + 1$. Мы используем знак $\leqslant$, так как оптимальная атака может не использовать некоторые строки матрицы $[\text{rot}(\mathbf{A}) \,|\, \mathbb{I} \,|\, \mathbf{t}]$. Нужный вектор $\mathbf{x} \in \Lambda_{\text{MLWR}}$ — это $\mathbf{x}_{\text{short}} = [\text{rot}(\mathbf{s})| - \mathbf{t}_{\text{low}}| - 1]$, где $\mathbf{t}_{\text{low}} = \mathbf{As} - \mathbf{t}$ и $\|\mathbf{t}_{\text{low}}\|_\infty \leqslant 2^{\nu-\mu}$. Это «короткий» вектор в решётке Λ_{MLWE}, так как он значительно короче $\sqrt{d} q^{1/nk}$ — границы Минковского для Λ_{MLWR}.

Это классическая «примальная» атака на LWR, сложность которой зависит от времени работы алгоритма BKZ для нахождения вектора длины $\|\mathbf{x}_{\text{short}}\|$. Оценить конкретное время работы BKZ — нетривиальная задача. Для получения значения 104 в таблице — консервативной оценки времени работы BKZ для решения задачи LWR — мы опирались на работу [15] и программный код [16]. Мы не приводим оценку для так называемой «дуальной» атаки на LWR, так как «примальный» метод для наших параметров оказался значительно эффективнее.

Рассмотрим теперь сложность задачи MSIS (так как задача SelfTargetSIS сводится к MSIS и для наших параметров атаки именно на MSIS работают эффективнее, определяющим фактором является сложность MSIS). Наиболее эффективная из всех известных атак на MSIS — нахождение короткого вектора в решётке

$$\Lambda_{\text{MSIS}} = \{\mathbf{x} \in \mathbb{Z}^d : [\text{rot}(\mathbf{A}) \,|\, \mathbb{I}]\, \mathbf{x} = \mathbf{0} \bmod q\}.$$

В отличие от атаки на MLWR, оптимальный алгоритм для задачи MSIS может опустить некоторые *столбцы* матрицы $[\mathrm{rot}(\mathbf{A}) \,|\, \mathbb{I}]$. Решением задачи MSIS считается короткий вектор $\mathbf{x} \in \Lambda_{\mathrm{MSIS}}$ с нормой $\|\mathbf{x}\|_\infty \leqslant \max\{2^{\nu-d+1}, 2(\gamma-\beta)\}$. Для параметров, приведённых в таблице, эти два значения примерно совпадают. Для получения конкретной сложности атаки MSIS мы пользовались стратегией [7, Appendix C]; скрипт, с помощью которого можно получить таблицу, доступен по ссылке `https://crypto-kantiana.com/elena.kirshanova/\#research`.

Предлагаемые параметры цифровой подписи и их уровень безопасности

n	k	ℓ	ν	μ	s	d	γ	$\mathbb{E}$[#итераций]	MSIS (BKZ-b)	MLWR (BKZ-b)
256	3	4	23	19	4	3	1048096	8	93 (320)	104 (357)

В таблице последние два параметра — 93 (соотв. 104) — соответствуют битовой сложности атаки на MSIS с оптимальным размером блока в алгоритме BKZ, равному 320 (соотв. MLWR с оптимальным размером блока 357). В обоих вычислениях полагаем (консервативно), что сложность нахождения короткого вектора в решётке размерности d равна $2^{0{,}292d}$, что асимптотически соответствует сложности алгоритма просеивания.

ЛИТЕРАТУРА

1. *Alkim E., Ducas L., Pöppelmann T., and Schwabe P.* Post-quantum key exchange: A new hope // USENIX Conf. Security Symposium. 2016. P. 327–343.
2. *Adeline L. and Stehlé S.* Worst-case to average-case reductions for module lattices // Des. Codes Cryptography. 2015. V. 75. No. 3. P. 565–599.
3. *Kirshanova E., Kolesnikov N., Malygina E., and Novoselov S.* Проект стандартизации постквантовой цифровой подписи (полная версия). `https://crypto-kantiana.com/main_papers/main_Signature.pdf`.
4. *Fiat A. and Shamir A.* How to prove yourself: Practical solutions to identification and signature problems // CRYPTO'86. LNCS. 1987. V. 263. P. 186–194.
5. *Lyubashevsky V.* Fiat — Shamir with aborts: Applications to lattice and factoring-based signatures // ASIACRYPT'2009. LNCS. 2009. V. 5912. P. 598–616.
6. *Bai S. and Galbraith S. D.* An improved compression technique for signatures based on learning with errors // Topics in Cryptology — CT-RSA 2014. LNCS. 2014. V. 8366. P. 28–47.
7. *Ducas L., Kiltz E., Lepoint T., et al.* CRYSTALS-Dilithium: A lattice-based digital signature scheme // IACR Trans. Cryptographic Hardware and Embedded Systems. 2018. No. 1. P. 238–268.
8. *Alkim E., Bindel N., Buchmann J., et al.* Revisiting TESLA in the quantum random oracle model // PQCrypto 2017. LNCS. 2017. V. 10346. P. 143–162.
9. *D'Anvers J.-P., Karmakar A., Roy S. S., and Vercauteren F.* Saber: Module-LWR based key exchange, CPA-secure encryption and CCA-secure KEM // Progress in Cryptology — AFRICACRYPT 2018. Springer, 2018. P. 282–305.
10. *Banerjee A., Peikert C., and Rosen A.* Pseudorandom functions and lattices // Ann. Intern. Conf. Theory and Appl. of Cryptographic Techniques. Springer, 2012. P. 719–737.
11. *Regev O.* On lattices, learning with errors, random linear codes, and cryptography // J. ACM. 2005. V. 56. No. 6. P. 84–93.
12. *Bogdanov A., Guo S., Masny D., et al.* On the hardness of learning with rounding over small modulus // Theory of Cryptography. LNCS. 2016. V. 9562. P. 209–224.
13. *Ajtai M.* Generating hard instances of lattice problems (extended abstract) // Proc. 28th Ann. ACM Symp. Theory Computing. 1996. P. 99–108.

14. *Kiltz E., Lyubashevsky V., and Schaffner C.,* A concrete treatment of Fiat — Shamir signatures in the quantum random-oracle model // Adv. Cryptology — EUROCRYPT 2018. Springer, 2018. P. 552–586.
15. *Albrecht M. R., Göpfert F., Virdia F., and Wunderer T.* Revisiting the expected cost of solving uSVP and applications to LWE // ASIACRYPT 2017. LNCS. 2017. V. 10624. P. 297–322.
16. *Albrecht M. R., Curtis B. R., Deo A., et al.* Estimate all the {LWE, NTRU} schemes! // SCN 2018. LNCS. 2018. V. 11035. P. 351–367.

УДК 512.64, 519.21, 519.72 DOI 10.17223/2226308X/13/15

КОНСТРУКЦИИ НЕЭНДОМОРФНЫХ СОВЕРШЕННЫХ ШИФРОВ

Н. В. Медведева, С. С. Титов

Исследуются совершенные по Шеннону (абсолютно стойкие к атаке по шифртексту) шифры. Получены достаточные условия того, что таблицы зашифрования неэндоморфных (эндоморфных) совершенных шифров не содержат латинских прямоугольников (квадратов). Приведён пример таких конструкций.

Ключевые слова: *совершенные шифры, эндоморфные шифры, неэндоморфные шифры.*

Рассмотрим вероятностную модель Σ_B шифра [1]. Пусть X, Y — конечные множества соответственно шифрвеличин и шифробозначений, с которыми оперирует некоторый шифр замены, K — множество ключей, причём $|X| = \lambda$, $|Y| = \mu$, $|K| = \pi$, где $\lambda > 1$, $\mu \geqslant \lambda$. Это означает, что открытые и шифрованные тексты представляются словами (ℓ-граммами, $\ell \geqslant 1$) в алфавитах X и Y соответственно. Согласно [2, 3], под *шифром* Σ_B будем понимать совокупность множеств правил зашифрования и правил расшифрования с заданными распределениями вероятностей на множествах открытых текстов и ключей. Шифры, для которых апостериорные вероятности открытых текстов совпадают с их априорными вероятностями, называются *совершенными*.

Описание эндоморфных ($\lambda = \mu$) с минимально возможным числом ключей ($|K| = |Y|$) совершенных шифров даёт теорема Шеннона, таблица зашифрования таких шифров — это латинский квадрат из равновероятных подстановок зашифрования [1].

Для неэндоморфных ($\lambda < \mu$) минимальных совершенных шифров характерно большое многообразие таблиц зашифрования: они не сводятся только к латинским прямоугольникам размера $\mu \times \lambda$ [4]. Для $\lambda = 2$, например, таблицы зашифрования могут быть составлены и из неравновероятных инъекций. Однако если все ключи равновероятны, то данный совершенный шифр является выпуклой оболочкой латинских прямоугольников, содержащихся в его таблице зашифрования, согласно аналогу теоремы Биркгофа [5]. Если $\lambda > 2$, то, даже для равновероятных инъекций зашифрования, неэндоморфный совершенный шифр может не содержать в своей таблице зашифрования латинских прямоугольников $\mu \times \lambda$ [6].

Таким образом, при $\mu > \lambda$ возникает естественная задача описания минимальных по включению (т. е. шифров, содержащих минимально возможное множество ключей зашифрования с ненулевыми вероятностями) совершенных шифров, не сводящихся к латинским прямоугольникам размера $\mu \times \lambda$, которые можно рассматривать как непосредственное обобщение теоремы Шеннона. Данную задачу можно трактовать как задачу описания выпуклого полиэдра, соответствующего совершенным шифрам, через нахождение его вершин [5].