Содержание

1 Введение

Развитие квантовых вычислений ставит под угрозу безопасность большинства современных криптосистем с открытым ключом, что обуславливает высокую актуальность задачи построения и анализа постквантовых (квантово-стойких) криптографических схем. Одним из ведущих и наиболее изученных направлений в постквантовой криптографии являются кодовые криптосистемы, родоначальником которых является схема Мак-Элиса на кодах Гоппы. Несмотря на почти полувековую историю криптоанализа, классическая система Мак-Элиса остается невзломанной, однако её широкому распространению препятствует значительный размер публичного ключа.

Для преодоления этого недостатка было предложено множество модификаций, направленных на уменьшение размера ключей за счет использования кодов с более компактным представлением или за счёт других способов сокрытия кода. К сожалению, многие из предложенных модификаций оказались уязвимыми к различным типам атак. Одним из интересных подходов являлась криптосистема AF (Augot-Finiasz), представленная в [1] и основанная на использовании кодов Рида-Соломона. Однако, несмотря на свою первоначальную привлекательность, криптосистема AF также оказалась впоследствии уязвимой [2].

В контексте продолжающегося поиска надежных постквантовых криптосистем, на конференции CBCrypto 2024 была представлена новая многомерная модификация криптосистемы AF. Вопрос её стойкости был поставлен как открытый, с указанием на необходимость поиска подходящих кодовых конструкций. Было также высказано предположение о потенциальной уязвимости при использовании GRS кодов к обобщению известной атаки . Следует отметить, что прямое многомерное обобщение существующей атаки представляется нетривиальной задачей (и не привело к существенному успеху в наших экспериментах). Таким образом, оценка безопасности многомерной схемы является актуальной задачей криптоанализа.

Ключевым результатом настоящей работы является разработка и анализ новых структурных атак, направленных непосредственно на предложенную многомерную модификацию криптосистемы AF и её усиленный вариант. Мы предлагаем эффективный метод атаки на ключ, использующий алгебраические свойства квадратов Шура-Адамара векторов, связанных с кодовыми словами секретного кода. Данный подход позволяет успешно провести атаку на исходную многомерную схему. Кроме того, анализируется усиленная версия криптосистемы, в которой для повышения стойкости публичный ключ дополнительно зашумляется секретной матрицей ранга 1. Также показано, что предложенная нами техника атаки эффективна и против этой усиленной модификации, позволяя восстановить структуру секретного ключа и расшифровывать впоследствии любые сообщения.

Структура работы. В разделе 2 приведены необходимые вспомогательные сведения из теории кодирования и криптографии. В разделе **??** приведены формальные описания криптосистемы AF и двух её атакуемых многомерных модификаций. В разделе 4 построены атаки на эти криптосистемы. В разделе 5 приведено заключение, а также выводы исследования.

2 Предварительные сведения

2.1 Обозначения

- Для произвольного вектора $x \in \mathbb{F}_q$ весом Хэмминга $\mathbf{wt}(x)$ будем называть количество его ненулевых координат.
- Операцией $\circ$ будем обозначать покомпонентное произведение (произведение Шура-Адамара) двух векторов $a = (a_1, \dots, a_n)$ и $b = (b_1, \dots, b_n)$ одинакового размера, т.е. $a \circ b = (a_1 \cdot b_1, \dots, a_n \cdot b_n)$.

2.2 Базовые понятия теории кодирования и криптографии

Конечное поле мощности q будем обозначать $\mathbb{F}_q$. Линейным блоковым кодом длины n над полем $\mathbb{F}_q$ называется линейное подпространство векторного пространства $\mathbb{F}_q^n$. Будем говорить, что линейный код $C(\subset \mathbb{F}_q^n)$ является $[n, k, d]$-кодом, если его длина равна n, размерность $\dim(C)$ равна k $(\leq n)$, а минимальное кодовое расстояние, определяемое как $\min_{c \in C \setminus \{0\}} wt(c)$, равно d. Матрица G размера $(k \times n)$ над полем $\mathbb{F}_q$, строки которой образуют базис кода C, называется порождающей матрицей кода C.

Рассмотрим далее криптосистему Мак-Элиса:

- **Генерация ключа** Пусть C - линейный [n,k,d]-код, G - его порождающая матрица, S - случайная матрица размера $(k \times k)$, P - перестановочная матрица размера $(n \times n)$. Публичный ключ G_{pub} генерируется следующим образом: $G_{pub} = S \cdot G \cdot P$. Отметим, что публичный ключ G_{pub} представляет собой матрицу трудно отличимую от случайной.
- **Шифрование** Пусть m - сообщение длины k, e - вектор ошибок веса $\leq t$, где $t = \lfloor \frac{d-1}{2} \rfloor$. Зашифрованное сообщение y вычисляется следующим образом: $y = m \cdot G_{pub} + e$
- **Расшифрование** К получателю приходит сообщение в виде $y = m \cdot G_{pub} + e$, где $G_{pub} = S \cdot G \cdot P$. Домножим обе части зашифрованного сообщения y на P^{-1}:
$$y \cdot P^{-1} = m \cdot S \cdot G + e \cdot P^{-1}.$$
Далее, используя алгоритм декодирования кода C, избавляемся от $e \cdot P^{-1}$ и получаем $m \cdot S$. Умножаем результат на S^{-1}:
$$m = (m \cdot S) \cdot S^{-1}$$
и получаем исходное сообщение m.

Преимуществами криптосистемы Мак-Элиса являются сложность задачи декодирования линейных кодов, которая является NP полной, и устойчивость к атакам на основе квантовых компьютеров.

2.3 Коды Рида-Соломона

Пусть $k \le n \le q$ и $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n) \in \mathbb{F}_q^n$ – вектор попарно различных элементов из поля $\mathbb{F}_q$. Опишем далее процедуру кодирования сообщения при помощи кодов Рида-Соломона.

Пусть $m = (m_0, m_1, ..., m_{k-1)}$, где $m_i \in \mathbb{F}_q$, – исходное сообщение. На основе сообщения m можно построить многочлен $f_m(x)$ степени $\le k-1$: $f_m(x) = \sum_{i=0}^{k-1} m_i x^i$.

Кодирующей функцией для кода Рида-Соломона $RS_q[\alpha, k]$ размерности k называется функция

$$ev: \begin{cases} \mathbb{F}_q^k \to \mathbb{F}_q^n \\ m \mapsto (f_m(\alpha_1), f_m(\alpha_2), ..., f_m(\alpha_n)), \end{cases}$$

т.е. для любого $m \in \mathbb{F}_q^k$ закодированное сообщение $ev(m)$ представляет собой упорядоченный набор значений многочлена $f_m(x)$ в точках $(\alpha_1, \alpha_2, ..., \alpha_n)$. Таким образом, код Рида-Соломона $RS_q[\alpha, k]$ длины n и размерности k представляет собой множество

$$RS_q[\alpha, k] = \left\{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f(x) \in \mathbb{F}_q[x],\ \deg(f) \le k-1\right\}$$

при этом вектор α называется носителем кода Рида-Соломона (RS-кода).

Порождающая матрица G_{RS} любого RS кода будет содержать в себе наборы элементов вектора $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)$ в степенях от 0 до k-1, таким образом, получается матрица Вандермонда размером $(k \times n)$:

$$G_{RS} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_j & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_j^2 & \cdots & \alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \alpha_1^i & \alpha_2^i & \cdots & \alpha_j^i & \cdots & \alpha_n^i \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \cdots & \alpha_j^{k-1} & \cdots & \alpha_n^{k-1} \end{pmatrix} = W \tag{1}$$

2.4 Декодер Велча-Берлекэмпа

Пусть m – сообщение длины k над полем $\mathbb{F}_q^k$, $m(x) = \sum_{i=0}^{k-1} m_i x^i$ – его представление в виде многочлена, $ev(m)$ – закодированное слово и e – вектор ошибки длины n и веса $wt(e) = t$, где t не превосходит корректирующую способность используемого $RS_q[\alpha, k]$-кода. Пусть $z = ev(m) + e$ – зашумлённое кодовое слово. Локатором ошибки будем называть такой многочлен L, удовлетворяющий правилу:

$$\forall i \in [1, n]: \ L(x_i) = \begin{cases} 0, & e_i \ne 0 \\ a \in \mathbb{F}_q^*, & e_i = 0 \end{cases}$$

Без потери общности будем считать, что $\deg(L) \le t$. В итоге, справедливо равенство:

$$L(x_i) \cdot m(x_i) = z_i \cdot L(x_i),\ \forall i \in [1, n].$$

Так как данное уравнение нелинейное, то сделав замену $N(x_i) = L(x_i) \cdot m(x_i)$, получим уже линейное уравнение:

$$N(x_i) = z_i \cdot L(x_i),\ \forall i \in [1, n],$$

при этом $\deg(N) \le k + t$. Далее перенесём всё в одну сторону:

$$N(x_i) - z_i \cdot L(x_i) = 0,\ \forall i \in [1, n],$$

чтобы составить матрицу коэффициентов:

$$M = \left(\begin{array}{cccc|cccc} x_1^0 & x_1^1 & \cdots & x_1^{k+t} & -z_1 x_1^0 & -z_1 x_1^1 & \cdots & -z_1 x_1^t \\ x_2^0 & x_2^1 & \cdots & x_2^{k+t} & -z_2 x_2^0 & -z_2 x_2^1 & \cdots & -z_2 x_2^t \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_n^0 & x_n^1 & \cdots & x_n^{k+t} & -z_n x_n^0 & -z_n x_n^1 & \cdots & -z_n x_n^t \end{array}\right)$$

Решая однородную систему линейных уравнений с матрицей M, получаем коэффициенты для многочленов N и L. Таким образом, мы легко восстанавливаем сообщение: $m = \frac{N}{L}$.

Замечание 1. *Многочлены N и L из системы уравнений могут находиться неоднозначно, однако это не влияет на восстановление сообщения.*

3 Криптосистема AF и её модификации

3.1 Криптосистема AF

Пусть $C = RS_q[\alpha, k]$ – код Рида-Соломона длины n, размерности k и носителем α, W – вес большой ошибки, такой что:

$$W > \frac{n-k}{2}$$

ω – вес маленькой ошибки, который ограничен сверху:

$$\omega \le \frac{n - W - k}{2}.$$

Теперь рассмотрим саму криптосистему AF, предложенную в работе [1]:

- **Генерация ключа** Пусть p – унитарный многочлен степени $k-1$ и E – случайный вектор «большой» ошибки длины n и веса W. Вычислим кодовое слово $c = ev(p)$ $RS_q[\alpha, k]$-кода. Публичный ключ g_{pub} генерируется следующим образом: $g_{pub} = c + E$, в то время, как секретным ключом является (p, E).

- **Шифрование** Пусть m – сообщение длины $k-1$ над полем $\mathbb{F}_q$, λ - случайный элемент поля $\mathbb{F}_q$ и e – случайный вектор «маленькой» ошибки веса ω. Сообщение m также может быть представлено в виде многочлена: $m = \sum_{i=0}^{k-1} m_i x^i$, $\deg(m) \leq k-2$. Зашифрованное сообщение y получается следующим образом:

$$y = ev(m) + \lambda \cdot g_{pub} + e.$$

- **Расшифрование** Определим код длины $n - W$, удалив все позиции, где $E_i = 0$ ($i \in [1, n]$). Таким образом, получается новый код Рида-Соломона $\overline{C} = RS_q[\overline{\alpha}, k]$ той же размерности с носителем

$$\overline{\alpha} = (\alpha_i)_{i \in [1,n]\ E_i=0}.$$

Удалим те же позиции в зашифрованном сообщении y и обозначим результат через $\overline{y}$. Аналогичные обозначения введём для $\overline{ev}(m)$, $\overline{c}$, $\overline{e}$. В итоге имеем:

$$\overline{y} = \overline{ev}(m) + \lambda \cdot \overline{c} + \overline{e}.$$

Зная, что $\overline{ev}(m) + \lambda \cdot \overline{c} \in \overline{C}$ и вес маленькой ошибки $\overline{e}$ меньше корректирующей способности кода $\overline{C}$, то можно составить уникальный многочлен r степени $k-1$ такой что

$$ev(r) = \overline{ev}(m) + \lambda \cdot \overline{c}.$$

Таким образом, переходя к самим многочленам, получаем:

$$r = m + \lambda \cdot p.$$

Так как степень $\deg(m) \leq k-2$, p – унитарный многочлен степени $\deg(p) = k-1$, то $\deg(m) < \deg(p)$, значит, элемент λ поля $\mathbb{F}_q$ является старшим коэффициентом многочлена r. В итоге получаем, что $\overline{y} = ev(r) + \overline{e}$, используя декодер Велча-Берлекэмпа для кода Рида-Соломона, вычисляем r, откуда получаем элемент λ. Теперь, зная секретный ключ p, сообщение m легко восстанавливается следующим образом:

$$m = r - \lambda \cdot p.$$

3.2 Многомерная модификация криптосистемы

В качестве первой модификации криптосистемы AF рассмотрим многомерный случай публичного ключа данной криптосистемы, представленный в докладе «Public-Key Encryption based on Supercode Decoding» на CBCrypto 2024:

- **Параметры** Пусть $l \leq k \leq n \leq q$, вектор α – общеизвестный носитель некоторого RS-кода.

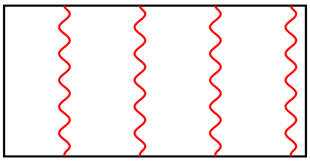

Рис. 1: Матрица E

- **Генерация ключа** Пусть G_{RS} – порождающая матрица $RS_q[\alpha, k]$-кода, A – случайная матрица размера $(k \times k)$, элементы которой принадлежат полю $\mathbb{F}_q$, E – $(l \times n)$-матрица «большой ошибки», которая содержит W ненулевых столбцов (см. рис. 1).

  Публичный ключ G_{pub} генерируется следующим образом:

  $$G_{pub} = A \cdot G_{RS} + E.$$

- **Шифрование** Пусть m – сообщение длины k над полем $\mathbb{F}_q$, λ - случайный вектор длины l над полем $\mathbb{F}_q$ и $e \in \mathbb{F}_q^n$ – случайный вектор «маленькой ошибки» веса ω. Зашифрованное сообщение y получается следующим образом:

  $$y = (m_1, \dots, m_{k-l}, \underbrace{0, \dots, 0}_{l \text{ штук}}) \cdot G_{RS} + \lambda \cdot G_{pub} + e.$$

- **Расшифрование** Для полученного сообщения справедливо равенство

  $$y = (m_1, \dots, m_{k-l}, \underbrace{0, \dots, 0}_{l \text{ штук}}) \cdot G_{RS} + \lambda \cdot A \cdot G_{RS} + \lambda \cdot E + e.$$

  Далее, без потери общности, будем считать, что случайная матрица A имеет следующий вид:

  $$A = (R \mid I_l),$$

  где R – случайная $(l \times k - l)$ - матрица, а I_l – единичная $(l \times l)$ матрица. Получаем, что

  $$y = (m_1, \dots, m_{k-l}, \underbrace{0, \dots, 0}_{l \text{ штук}}) \cdot G_{RS} + \lambda \cdot (R \mid I_l) \cdot G_{RS} + \lambda \cdot E + e =$$

  $$= \underbrace{[(m1, \dots, m_{k-l}) + \lambda \cdot R \mid \lambda_1, \dots, \lambda_l]}_{m'} \cdot G_{RS} + \lambda \cdot E + e.$$

  В полученном сообщении y и в «маленькой ошибке» e удалим координаты, в которых столбцы матрицы E не равны нулю, и аналогичное действие проделаем со столбцами матрицы G_{RS}:

  $$\overline{y} = (y_i)_{i \in [1,n] \; E_i = 0}$$

$$\overline{e} = (e_i)_{i\in[1,n]\ E_i=0}$$

$$\overline{G}_{RS} = (G_{RS_i})_{i\in[1,n]\ E_i=0}$$

Таким образом, избавляясь от «большой ошибки» E, получаем:

$$\overline{y} = m' \cdot \overline{G}_{RS} + \overline{e}.$$

Применяя к $\overline{y}$ декодер Велча-Берлекэмпа, извлекаем

$$m' = [(m_1, \ldots, m_{k-l}) + \lambda \cdot R \mid \lambda_1, \ldots, \lambda_l] \,.$$

Отбрасывая в m' последние l позиции, в которых находится копия вектора $\lambda = (\lambda_1 \ldots \lambda_l)$, получаем $\tilde{m} = (m_1, \ldots, m_{k-l}) + \lambda \cdot R$. Последним действием, зная λ и R, отнимаем их произведение от $\tilde{m}$ и восстанавливаем исходное сообщение:

$$\tilde{m} - \lambda \cdot R = (m_1, \ldots, m_{k-l}) = m.$$

3.3 Усиленная многомерная модификация

Рассмотрим ещё одну модификацию криптосистемы Augot и Finiasz, в которой также, как и в прошлой системе, используется многомерный публичный ключ, но дополнительно применяется зашумление публичного ключа матрицей ранга 1.

- **Генерация ключа** Пусть G_{RS} – порождающая матрица $RS_q[\alpha, k]$-кода, A – случайная матрица размера $(k{\times}k)$, элементы которой принадлежат полю $\mathbb{F}_q$, E – матрица большой ошибки, которая содержит W ненулевых столбцов, $\gamma(\in \mathbb{F}_q^k)$ – случайный вектор-столбец размера $(k \times 1)$ и $\beta(\in \mathbb{F}_q^k)$ – случайный вектор-строка длины k . Публичный ключ G_{pub} генерируется следующим образом:

$$G_{pub} = A \cdot G_{RS} + E + \gamma \cdot \beta.$$

- **Шифрование** Пусть m – сообщение длины k над полем $\mathbb{F}_q$, $\lambda \in \mathbb{F}_q^n$ - случайный вектор и e – случайный вектор «маленькой ошибки» веса ω. Зашифрованное сообщение y получается следующим образом:

$$y = (m_1, \ldots, m_{k-l}, \underbrace{0, \ldots, 0}_{l \text{ штук}}) \cdot G_{RS} + \lambda \cdot G_{pub} + e.$$

- **Расшифрование** Пусть $\tilde{m} = (m_1, \ldots, m_{k-l})$ – исходное сообщение без последних l - нулевых координат, матрица G_1 составлена из первых $k-l$ строчек матрицы G_{RS}, а матрица G_2 построена из последних l строчек матрицы G_{RS}, т.е. $G_{RS} = \left(\frac{G_1}{G_2}\right)$. Тогда справедливо равенство:

$$y = \tilde{m} \cdot G_1 + \lambda \cdot (A \cdot G_{RS} + E + \gamma \cdot \beta) + e.$$

В полученном сообщении y, в «маленькой ошибке» e и векторе β удалим координаты, в которых столбцы матрицы E не равны нулю, и аналогичное действие проделаем со столбцами матриц G_{RS}, G_1 и G_2:

$$\overline{y} = (y_i)_{i\in[1,n]\ E_i=0}$$

$$\overline{e} = (e_i)_{i\in[1,n]\ E_i=0}$$

$$\overline{\beta} = (\beta_i)_{i\in[1,n]\ E_i=0}$$

$$\overline{G}_{RS} = (G_{RS_i})_{i\in[1,n]\ E_i=0}$$

$$\overline{G}_1 = (G_{1_i})_{i\in[1,n]\ E_i=0}$$

$$\overline{G}_2 = (G_{2_i})_{i\in[1,n]\ E_i=0}$$

Далее, без потери общности, будем считать, что случайная матрица A имеет следующий вид:

$$A = (R \mid I_l),$$

где R – случайная $(l \times k - l)$ - матрица, а I_l – единичная $(l \times l)$ матрица. Таким образом, получаем новое сообщение:

$$\overline{y} = \tilde{m} \cdot \overline{G}_1 + \lambda \cdot A \cdot \overline{G}_{RS} + \lambda \cdot \gamma \cdot \overline{\beta} + \overline{e} =$$

$$= (\tilde{m} + \lambda \cdot R \mid \lambda) \cdot \begin{pmatrix} \overline{G}_1 \\ \overline{G}_2 \end{pmatrix} + \lambda \cdot \gamma \cdot \overline{\beta} + \overline{e}.$$

Ясно, что $(\lambda \cdot \gamma)$ – элемент поля $\mathbb{F}_q$, поэтому, перебирая все возможные значения $\theta \in \mathbb{F}_q$ и, отнимая от сообщения $\overline{y}$ значение $(\theta \cdot \overline{\beta})$, где исходный вектор β известен легальному пользователю, можно получить слово вида:

$$\overline{y} - (\lambda \cdot \gamma \cdot \overline{\beta}) = (\tilde{m} + \lambda \cdot R \mid \lambda) \cdot \underbrace{\begin{pmatrix} \tilde{G}_1 \\ \tilde{G}_2 \end{pmatrix}}_{\tilde{G}_{RS}} + \overline{e}$$

которое можно затем декодировать при помощи декодера Велча-Берлекэмпа.
В случае успеха, извлекаем

$$m' = (\tilde{m} + \lambda \cdot R \mid \lambda).$$

Теперь, отбрасывая последние l позиций, на которых стоит вектор $\lambda = (\lambda_1, \ldots, \lambda_l)$, получаем $m'' = (\tilde{m} + \lambda \cdot A)$. Последним действием, зная λ и R, отнимаем их произведение от m'' и восстанавливаем исходное сообщение:

$$m'' - \lambda \cdot R = (m_1, \ldots, m_{k-l}) = \tilde{m}.$$

4 Атаки на рассматриваемые криптосистемы

4.1 Атака на многомерную криптосистему AF

Атака на данную криптосистему будет основываться на использовании квадратов Шура-Адамара для линейных кодов. Итак, на входе злоумышленник получает зашифрованное сообщение:

$$y = (m_1, \ldots, m_{k-l}, \underbrace{0, \ldots, 0}_{l \text{ штук}}) \cdot G_{RS} + \lambda \cdot G_{pub} + e,$$

публичный ключ G_{pub} и порождающую матрицу G_{RS} RS-кода. Секретным ключом здесь является матрица «большой ошибки» E и вектор $\lambda = (\lambda_1, \ldots, \lambda_l)$. Покажем, что знание номеров ненулевых столбцов матрицы E достаточно, чтобы восстановить сообщение m.

Пусть $\tilde{m} = (m_1, \ldots, m_{k-l})$ и матрица G_1 составлена из первых $k - l$ строчек матрицы G_{RS}, тогда перепишем полученное зашифрованное сообщение в следующем виде:

$$y = \tilde{m} \cdot G_1 + \lambda \cdot G_{pub} + e =$$

$$= (\underbrace{m_1, \ldots, m_{k-l}}_{\tilde{m}} \mid \underbrace{\lambda_1, \ldots, \lambda_l}_{\lambda}) \cdot \left(\frac{G_1}{G_{pub}}\right) + e.$$

Составим матрицу

$$\tilde{G} = \left(\frac{G_{RS}}{G_{pub}}\right)$$

и применим к ней метод квадратов. Здесь будем говорить, что матрица возводится в квадрат по правилу:

$$\tilde{G}^2 = \begin{pmatrix} \tilde{g}_1 \circ \tilde{g}_1 \\ \tilde{g}_1 \circ \tilde{g}_2 \\ \vdots \\ \tilde{g}_1 \circ \tilde{g}_{k+l} \\ \vdots \\ \tilde{g}_2 \circ \tilde{g}_{k+l} \\ \vdots \\ \tilde{g}_{k+l} \circ \tilde{g}_{k+l} \end{pmatrix}, \ \tilde{g}_{i \in [1, k+l]} \text{ – столбцы матрицы } \tilde{G}$$

Суть метода заключается в том, что мы будем поочерёдно вырезать столбцы из матрицы $\tilde{G}$, возводить новую матрицу в квадрат и сравнить её ранг с рангом исходной матрицы $\tilde{G}$. В случае, если ранги не совпадают, то мы будем знать, что вырезанный столбец является зашумлённым. Таким образом, проделав этот алгоритм с каждым из столбцов матрицы $\tilde{G}$, мы получаем множество номеров

$$V = \{i \in [1, n] \mid i : \ \mathrm{rank}(\tilde{G}^2_{j \in \text{columns } i \neq j}) \neq \mathrm{rank}(\tilde{G}^2)\}$$

зашумлённых столбцов. Теперь вырежем из матриц G_{RS}, G_1 и G_{pub} столбцы с номерами множества V, аналогичное действие проделаем с координатами полученного сообщения y и вектором «маленькой ошибки» e. В итоге получаем:

$$\overline{y} = (y_i)_{\{i\in[1,n] \mid i:\, i\notin V\}}$$

$$\overline{e} = (e_i)_{\{i\in[1,n] \mid i:\, i\notin V\}}$$

$$\overline{G}_{RS} = (G_{pub_i})_{\{i\in[1,n] \mid i:\, i\notin V\}}.$$

$$\overline{G}_{pub} = (G_{RS_i})_{\{i\in[1,n] \mid i:\, i\notin V\}}.$$

$$\overline{G}_1 = (G_{1_i})_{\{i\in[1,n] \mid i:\, i\notin V\}}.$$

Построим матрицу $\overline{G} = \frac{\overline{G}_1}{\overline{G}_{pub}}$, в следствии чего, справедливо равенство:

$$\overline{y} = (\underbrace{m_1, \dots, m_{k-l}}_{\tilde{m}} \mid \underbrace{\lambda_1, \dots, \lambda_l}_{\lambda}) \cdot \overline{G} + \overline{e}.$$

Ясно, что существует такая матрица S, что $\overline{G} = S \cdot \overline{G}_{RS}$. При помощи метода Гаусса нетрудно найти матрицу S, после чего сообщение $\overline{y}$ с выколотыми координатами можно переписать в следующем виде:

$$\overline{y} = (\underbrace{m_1, \dots, m_{k-l}}_{\tilde{m}} \mid \underbrace{\lambda_1, \dots, \lambda_l}_{\lambda}) \cdot S \cdot \overline{G}_{RS} + \overline{e.}$$

Теперь в правой части порождающая матрица имеет вид матрицы Вандермонда, поэтому, применяя к $\overline{y}$ декодер Велча-Берлекэмпа, извлекаем сообщение

$$m' = (\underbrace{m_1, \dots, m_{k-l}}_{\tilde{m}} \mid \underbrace{\lambda_1, \dots, \lambda_l}_{\lambda}) \cdot S.$$

После домножаем полученное сообщение на S^{-1}:

$$m' \cdot S^{-1} = (\underbrace{m_1, \dots, m_{k-l}}_{\tilde{m}} \mid \underbrace{\lambda_1, \dots, \lambda_l}_{\lambda}) = m''.$$

Далее по аналогии с расшифрованием отбрасываем в m'' позиции, в которых находятся $(\lambda_1 \dots \lambda_l)$ и получаем $(m_1, \dots, m_{k-l})$ – исходное сообщение m.

4.2 Атака на усиленную многомерную криптосистему AF

Злоумышленник на входе получает: зашифрованное сообщение y, публичный ключ G_{pub} и порождающую матрицу G_{RS}. Таким образом, имеем:

$$y = (m1, \dots, m_{k-l}, \underbrace{0, \dots, 0}_{l \text{ штук}}) \cdot G_{RS} + \lambda \cdot G_{pub} + e.$$

Пусть $\tilde{m} = (m_1, \dots, m_{k-l})$, матрица G_1 состоит из первых $(k-l)$ строчек матрицы G_{RS}, тогда можем построить новую матрицу $G = \left(\frac{G_1}{G_{pub}}\right)$. Далее, применим к матрице G метод квадратов Шура-Адамара(как это делалось в предыдущей атаке) и получим множество номеров V – зашумлённых столбцов матрицы G, т.е.

$$V = \{i \in [1, n] \mid i: \ \operatorname{rank}(G^2_{j \in \text{columns } i \neq j}) \neq \operatorname{rank}(G^2)\}.$$

Теперь вырежем из матриц G_1, G_{pub}, G_{RS} и G зашумлённые столбцы с номерами из множества V и аналогичное действие проделаем с координатами вектора зашифрованного сообщения y и вектора «маленькой ошибки» e. В итоге, получаем:

$$\overline{y} = (y_i)_{\{i \in [1,n] \mid i:\ i \notin V\}}$$

$$\overline{e} = (e_i)_{\{i \in [1,n] \mid i:\ i \notin V\}}$$

$$\overline{G}_{RS} = (G_{RS_i})_{\{i \in [1,n] \mid i:\ i \notin V\}}.$$

$$\overline{G}_{pub} = (G_{pub_i})_{\{i \in [1,n] \mid i:\ i \notin V\}}.$$

$$\overline{G}_1 = (G_{1_i})_{\{i \in [1,n] \mid i:\ i \notin V\}}.$$

$$\overline{G} = (G_i)_{\{i \in [1,n] \mid i:\ i \notin V\}}.$$

Нетрудно заметить, что атакующий получает доступ к матрице

$$\overline{G} = \left(\frac{\overline{G}_1}{\overline{G}_{pub}}\right) = \left(\frac{\overline{G}_1}{A \cdot \overline{G}_{RS} + \gamma \cdot \beta}\right) = \left(\frac{\overline{G}_1}{R \cdot \overline{G}_1 + \overline{G}_2 + \gamma \cdot \beta}\right).$$

Замечание 2. *Напомним, что в последнем равенстве матрица A имеет вид: $A = (R \mid I_l)$ также, как и в этапе расшифрования.*

Проанализируем структуру кода, порождаемого матрицей $\overline{G}$. Пусть B – это такая обратимая $(l \times l)$ матрица, что

$$B \cdot \gamma = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{2}$$

(вообще говоря, матрица B в явном виде злоумышленнику не известна). Очевидно, что

$$\left(\frac{I_{k-l}}{B}\right) \cdot \overline{G} = \left(\frac{\overline{G}_1}{B \cdot R \cdot \overline{G}_1 + B \cdot \overline{G}_2 + B \cdot \gamma \cdot \beta}\right) = \begin{bmatrix} \overline{G}_1 \\ \hline \overline{G}_3 \\ \hline \text{—}v\text{—} \end{bmatrix},$$

где $\overline{G}_3$ это первые $l-1$ строк матрицы $B \cdot \overline{G}_{pub}$, а вектор v – последняя строка той же матрицы. В силу равенства (2) следует, что матрицы $\overline{G}_1$ и $\overline{G}_3$ состоят из кодовых слов $\overline{RS}$-кода, а вектор v этому коду не принадлежит, поэтому справедливо следующее разложение кода порождённого матрицей $\overline{G}$ в прямую сумму:

$$\left\langle \overline{G} \right\rangle = \left\langle \overline{G}_1 \right\rangle \oplus \left\langle \overline{G}_3 \right\rangle \oplus \langle v \rangle .$$

Атакующему известна матрица $\overline{G}_1$, поэтому следующий шаг атаки будет нацелен на восстановлении $\left\langle \overline{G}_3 \right\rangle$ (т.е. будет найден какой-нибудь базис $\left\langle \overline{G}_3 \right\rangle$). Для этого воспользуемся подходом, описанным в [3].

Пусть векторы z_1, z_2 – кодовые слова $\overline{RS}$-кода, т.е. $z_1 = a \cdot \overline{G}_{pub}$, $z_2 = b \cdot \overline{G}_{pub}$, где a и b – случайные векторы длины $(k-l)$ над полем $\mathbb{F}_q$.

А вектор z_3 получается умножением случайного вектора p длины $(k-l)$ над полем $\mathbb{F}_q$ на публичный ключ $\overline{G}_{pub}$ с вырезанными зашумлёнными столбцами, но испорченными строчками. Теперь составим матрицы всевозможных попарных поэлементных произведений строк матрицы $\overline{G}$ и векторов z_1, z_2, z_3:

$$\tilde{G}_1 = z_1 \circ \overline{g}_i, i \in [1, k-l]$$

$$\tilde{G}_1 = z_2 \circ \overline{g}_i, i \in [1, k-l]$$

$$\tilde{G}_1 = z_3 \circ \overline{g}_i, i \in [1, k-l]$$

где $\overline{g}_i$ – строка матрицы $\overline{G}$. Построим блочную матрицу G':

$$G' = \begin{pmatrix} \tilde{G}_1 \\ \tilde{G}_2 \\ \tilde{G}_3 \end{pmatrix}$$

и вычислим её ранг. Если $\operatorname{rank}(G') \leq 2 \cdot k - 1 + W$, то вектор z_3 является кодовым словом из $\left\langle \overline{G}_3 \right\rangle$. Далее, мы будем повторять генерацию вектора z_3 и вычисление ранга матрицы G' до тех пор, пока ранг матрицы M, составленной из найденных подходящих векторов z_3, не будет равен рангу матрицы $\overline{G}_3$, т.е. должно выполняться: $\operatorname{rank}(M) = l - 1$. Таким образом, восстанавливается базис $\left\langle \overline{G}_3 \right\rangle$, как множество из $l-1$ линейно-независимых строк матрицы M. Следующим шагом построим матрицу $K = \left(\frac{\overline{G}_1}{M'}\right)$, где матрица M' – состоит из $l-1$ независимых строк матрицы M. Далее, перебирая все строчки g_i матрицы $\overline{G}_{pub}$, $i \in [1, k-l]$, на каждом шаге строим матрицу $K_g = \begin{pmatrix} K \\ \hline -g_i- \end{pmatrix}$, и если $\operatorname{rank}(K_g) = \operatorname{rank}(K) + 1$, то строка g_i является зашумлённой(испорченной) в матрице $\overline{G}_{pub}$, обозначим найденную строку за p.

Покажем, как использовать восстановленную структуру для расшифрования. Чтобы избавиться от оставшегося зашумления в сообщении $\overline{y}$, будем перебирать все элементы поля $a \in \mathbb{F}_q$ и отнимать от $\overline{y}$ произведение найденной зашумлённой строки p и элемента a, т.е.

$$y' = \overline{y} - a \cdot p.$$

Используя декодер Велча-Берлекэмпа для сообщения y' и порождающей матрицы $\overline{G}_{RS}$, извлекаем сообщение m', которое скорее всего не будет совпадать с исходным m (т.к. декодер Велча-Берлекэмпа предполагает, что во время кодирования использовалось каноничная порождающая матрица (1)). После чего вычисляем кодовое слово $c = m' \cdot \overline{G}_{RS}$(которое принадлежит RS-коду), чтобы найти «маленькую ошибку» e. Атакующий знает вес ошибки $wt(e) = \omega$, поэтому, вычисляя:

$$e' = \overline{y} - c - a \cdot p$$

можем сравнить вес e и e'. В случае, если $wt(e) = wt(e')$, то мы нашли верное кодовое слово c, иначе возвращаемся на этап генерации элемента $a \in \mathbb{F}_q$. Последним шагом, зная G_{RS} и c, решаем систему $m \cdot G_{RS} = c$ и находим исходное сообщение m.

5 Заключение

В рамках данной работы представлен анализ безопасности двух многомерных модификаций криптосистемы AF, основанных на кодах Рида-Соломона. Ключевым элементом исследования является применение произведения Шура-Адамара для построения структурных атак. Данный подход отличается от метода, предложенного в [2] для атаки на исходную криптосистему AF. Построенные в работе атаки демонстрируют уязвимость обеих модификаций. На основании полученных данных можно сделать вывод о том, что коды Рида-Соломона не обеспечивают необходимой стойкости в рассматриваемых криптосистемах. Более того, результаты анализа указывают на потенциальную уязвимость родственных классов кодов (БЧХ, алгебро-геометрических, коды Гоппы, коды Рида-Маллера) к расширенным версиям предложенных атак, что планируется проверить в последующих экспериментах. Таким образом, обнаруженные атаки ставят под сомнение применимость широкого класса алгебраических кодов и актуализируют задачу поиска альтернативных кодовых конструкций. В качестве перспективного направления рассматривается использование кодов, ведущих себя подобно случайным относительно произведения Шура-Адамара.

Список литературы

[1] Daniel Augot и Matthieu Finiasz. «A public key encryption scheme based on the polynomial reconstruction problem». В: *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2003, с. 229—240.

[2] Jean-Sebastien Coron. «Cryptanalysis of a public-key encryption scheme based on the polynomial reconstruction problem». В: *Public Key Cryptography–PKC 2004: 7th International Workshop on Theory and Practice in Public Key Cryptography, Singapore, March 1-4, 2004. Proceedings 7*. Springer. 2004, с. 14—27.

[3] Alain Couvreur и др. «Distinguisher-based attacks on public-key cryptosystems using Reed–Solomon codes». В: *Designs, Codes and Cryptography* 73 (2014), с. 641—666.