Постквантовая криптография на кодах и решетках: общее и различия.

Кабатянский Григорий Анатольевич

Сколковский институт науки и технологии (Сколтех)

3 марта 2022

Современная криптография и система МакЭлиса

«Новые направления в современной криптографии» Диффи и Хеллмана (1976), и RSA(1978).

Идея, идущая от Диффи и Хелмана, взять NP-трудную задачу, у которой для некоторого множества входов есть простое (полиномиальное) решение и "замаскировать" эти входы как случайные. Они взяли задачу о рюкзаке, но ее через несколько лет (1982) сломали.

Но система МакЭлиса (1978), такж основанная на идее рюкзака, только алгебраического а не числового, оказалась не только до сих пор не сломана, но остается одним из претендентов на стандарт пост-квантовой криптографии.

Обычный рюкзак. Имеются *n целых* чисел $h_1, \ldots, h_n$ (веса или объемы) и рюкзак объема S. Узнать имеет ли уравнение *двоичное* решение.

$$\sum_{i=1}^{n} x_i h_i = S,\ x_i \in \{0, 1\} \tag{1}$$

Это NP-полная задача. Она просто решается если $\sum_{i=1}^{j} h_i < h_{j+1}$ для любого j.

Алгебраический рюкзак. Имеется абелева группа G, в которой выбраны элементы $h_1, \ldots, h_n$. Для построения системы МакЭлиса возьмем группу $\mathbb{Z}_2^r$ r-мерных двоичных векторов со сложением по модулю 2 в качестве группы G . В уравнении (1) нас интересует двоичное решение с минимальным числом единиц.
Это NP-трудная задача, известная в теории кодирования (ТК) как декодирование линейного кода по минимуму расстояния Хэмминга (1978).

Разреженные решения системы линейных уравнений по модулю 2

Весом Хэмминга $\mathrm{a} = (a_1, \ldots, a_N)$ называется число его ненулевых координат $wt(\mathrm{a}) = |\{i : a_i \neq 0\}| = ||\mathrm{a}||_0$. Рассмотрим систему линейных уравнений над полем из двух элементов

$$\sum_{i=1}^{n} x_i \mathrm{h}_i = \mathrm{S} \mod 2,\ x_i \in \{0, 1\} \tag{2}$$

где $\mathrm{h}_1, \ldots, \mathrm{h}_n$ - r-мерные двоичные векторы и мы хотим найти решение x с минимальным весом Хэмминга. Уравнение можно переписать в матричном виде как

$$H\mathrm{x}^T = \mathrm{S} \tag{3}$$

где $r \times n$-матрица H составлена из столбцов $\mathrm{h}_1, \ldots, \mathrm{h}_n$. Это уравнение в ТК называется синдромным. Матрица H называется проверочной матрицей линейного кода C, определяемого как $C := \{\mathrm{x} : H\mathrm{x}^T = 0\}$. Известно несколько классов матриц, для которых синдромное уравнение легко решается (за полиномиальное время).

Как их замаскировать?

Маскировка по Ниедеррайтеру

Алиса берет некоторую матрицу H, для которой известно как просто решать синдромное уравнение, и держит свой выбор в секрете! Затем выбирает две случайные матрицы: $r \times r$ невырожденную матрицу A и $n \times n$ перестановочную матрицу P, выбор которых также держит в секрете, и формирует *открытый* ключ-матрицу

$$H_{pub} := AHP \tag{4}$$

Боб посылает Алисе r-мерный вектор $S = H_{pub}x^T$, а сообщением является вектор x - решение синдромного уравнения минимального веса. Все бы хорошо, но мы умеем просто решать синдромное уравнение либо для двух очень простых классов кодов (Хэмминга и им двойственные) и система нестойкая, либо, и это и есть система МакЭлиса-Ниедеррайтера ! - при дополнительном ограничении, что вес решения меньше половины минимального расстояния кода. Появляется дополнительная процедура - нумерация двоичных слов из не более чем t единиц (алгоритм Мудрова), но это небольшой недостаток.

Большой недостаток - неизвестна сложность задачи поиска решения синдромного уравнения с весом меньше половины минимального расстояния кода.

Маскировка по МакЭлису и дешифрование

В исходной системе МакЭлиса использовался код C с минимальным расстоянием $2t+1$, позволяющий просто (и однозначно) находить решение синдромного уравнения веса не более t. Сообщение m состоит из $k = n - r$ бит, оно превращается в кодовое слово $c = mG_{pub} \in C$, где $G_{pub} = A'GP$, а G называется порождающей матрицей кода C. Боб посылает Алисе вектор $y = c + e$, где e случайный вектор из t единиц.

Чтобы восстановить сообщение из синдромного уравнения $H_{pub}x^T = S$ Алиса делает следующие преобразования $AHPx^T = AH(xP)^T = AHz^T = S$, следовательно, $Hz^T = A^{-1}S = S'$, где вес z такой же как вес x,т.е. не больше t, и Алиса знает как решить такое синдромное уравнение.

Чтобы найти сообщение в системе МакЭлиса Алиса находит H_{pub} такое, что $H_{pub}c^t = 0$. Тогда $S := H_{pub}y^t = H_{pub}c^t + H_{pub}e^t = H_{pub}e^t$. Это синдромное уравнение, которое Алиса знает как решать, из него она найдет e. а затем и само сообщение m. На этом пути нетрудно доказать, что системы МакЭлиса и Нидеррайтера эквиваленты (В.М. Сидельников, 1994).

Недостатки МакЭлиса-Нидеррайтера и что кроме?

Если код слишком хороший, как, например, коды РС (богатая группа симметрий), то система взламывается (В. М. Сидельников, С. О. Шестаков, 1992).

Большие "ключи" – и открытый, и закрытый, порядка сотни тысяч бит. С этим борются, дошли до десятков тысяч, но не ниже.

Шифрованное сообщение длиннее секретного.

Плюсы – быстрая обработка и пока не знаем как взламывать на квантовом компе?

Криптосистема ГПТ (Габидулин-Парамонов-Третьяков; 1991), основанная на кодах, исправляющих ошибки решетчатой конфигурации (или в ранговой метрике). Многочисленные атаки и исправления.

Система Сидельникова

МакЭлис предложил брать коды Гоппы (а не БЧХ), потому что их много.
В.М. Сидельников (1994) предложил систему всего с одним кодом (Рида-Маллера), но повторяющимся несколько (u) раз.

Зададим $G_{pub} = GP$, где P - это $un \times un$ матрица перестановки, а $G = (G_1|G_2|\ldots|G_u)$, где G_i $k \times n$ -это разные порождающие матрицы одного и того же кода C.
Сидельников предложил брать в качестве кода C код Рида-Маллера небольшого порядка (для них известны алгоритмы декодирования, близкие к максимуму правдоподобия), а u брать не менее 3.

Система взломана, но в случае без повторений,т.е. $u = 1$! Minder, Shokrollahi 2007; М. А. Бородин, И. В. Чижов, 2014. **Стойкость по-настоящему не исследована.**

Обобщение системы - $G_1|G_2|\ldots|G_u$ - порождающие матрицы разных кодов и разной длины, но одной размерности (Егорова и др. 2017). Исправлять ошибки можно каждым из кодов, а затем сравнивать результаты. Случай $u = 2$ сломан.

Решетки

Решеткой в n-мерном евклидовом пространстве $\mathbb{R}^n$ называется дискретная подгруппа со сложению. Конструктивно любая решетка Λ задается своим базисом $\mathrm{v}_1, \ldots, \mathrm{v}_m$, $m \leq n$

$$\Lambda = \{\mathrm{v} = a_1\mathrm{v}_1 + \ldots + a_m\mathrm{v}_m : a_i \in \mathbb{Z}\}$$

Задачи, на которых стоит “ решетчатая” криптография:
– найти длину самого короткого вектора решетки;
– для заданного вектора $\mathrm{x} \in \mathbb{R}^n$ найти ближайший вектор решетки.

Λ это q-решетка если $q\mathbb{Z}^n \subset \Lambda \subset \mathbb{Z}^n$.
То есть q-решетка это линейный q-ичный код $C + q\mathbb{Z}^n$.

Известно, что для кодов эти две задачи NP-трудные.
Значит, и для решеток тоже.

Неформально - брать кодовые криптосистемы и распространять их на q-решетки. Препятствие - мы знаем классы хороших кодов, с простыми алгоритмами исправления ошибок, но они (коды) не дают хороших решеток (и, видимо, не могут дать).

Решетка хорошая, если она дает большую плотность заполнения пространства одинаковыми шарами.
Лучший класс конструктивных решеток - это решетки Барнса-Уолла, получаемые из кодов Рида-Маллера, но они плохие как решетки (как и коды Рида-Маллера).

Система LWE (обучение с ошибками) - одна из самых популярных. Она опирается на сложность исправления ошибок случайным линейным кодом в вероятностной модели, т.е. ошибки исправляются не гарантированно, а с некой очень маленькой вероятностью ложного исправления. Я не знаю является ли эта задача NP-трудной.

Р. Л. Добрушин, С. И. Ортюков, "... самокорректирующихся схем из ненадежных функциональных элементов", 1977.

Есть квантовые "чипы"на 5 кубит, но малонадежные, с вероятностью ошибки $p = 2 \times 10^{-3}$. Пусть выполняется некоторое вычисление в дискретной области из m элементов. Возьмем L схем и запустим их параллельно. Вероятность того, что правильный результат получится меньше двух раз равна $p^L + L(1-p)p^{L-1}$. Это одно нежелательное событие и его вероятность (при $L = 10$ уже меньше 10^{-24}).

Если правильно вычислили два или более раз, то оценим какова вероятность что неверные вычисления тоже совпадут. Для этого оценим вероятность того, что все они будут разные - это

$$(1 - \frac{1}{m})(1 - \frac{2}{m}) \dots (1 - \frac{L-2}{m})$$

Пусть $m >> L$, возьмем логарифм и получим, что эта вероятность $\approx \exp(-\frac{L^2}{2m})$