60 Zerosum

Задача. Найти попарно различные 128-битовые слова $X_1, \ldots, X_{128}$ такие, что сумма

$$X_1 + X_2 + \ldots + X_{128} + \texttt{Belt}_0(X_1) + \texttt{Belt}_0(X_2) + \ldots + \texttt{Belt}_0(X_{128})$$

состоит из одних нулей. Здесь + обозначает поразрядное по модулю 2 сложение двоичных слов, $\texttt{Belt}_0$ — зашифрование Belt (СТБ 34.101.31) на нулевом ключе.

Решение. Для решения задачи применим обобщенную атаку «дней рождения» [Wagner D. A generalized birthday problem. CRYPTO 2002 Proceedings. Springer, p. 288–302].

Будем выбирать случайные различные $X \in \{0,1\}^{128}$ и строить векторы $Z = X + \texttt{Belt}_0(X)$. Построенные векторы будем объединять в наборы и составлять из этих наборов *структуры*. Структура i-го уровня — это множество наборов $(Z_1, Z_2, \ldots, Z_{2^i})$ таких, что сумма $Z_1 + Z_2 + \ldots + Z_{2^i}$ начинается с $16i$ нулей.

Пусть $L = \{(Z_1, Z_2, \ldots, Z_{2^i})\}$, $L' = \{(Z'_1, Z'_2, \ldots, Z'_{2^i})\}$ — структуры i-го уровня, $|L| = T$ и $|L'| = T'$. Имеется TT' составных наборов $(Z_1, Z_2, \ldots, Z_{2^i}, Z'_1, Z'_2, \ldots, Z'_{2^i})$, сумма их элементов обязательно начинается с $16i$ нулей. Более того, в среднем для $TT'/2^{16}$ наборов сумма начинается с $16(i+1)$ нулей. Такие наборы образуют структуру $(i+1)$-го уровня. Если $T, T' \approx 2^{16}$, то новая структура также содержит $\approx 2^{16}$ наборов.

Начнем со 128 структур 0-го уровня, по 2^{16} элементов в каждой. По этим структурам построим 64 структуры 1-го уровня, затем 32 структуры второго уровня, и так далее, вплоть до единственной структуры 7-го уровня. Каждая из построенных структур содержит $\approx 2^{16}$ элементов.

Структура 7-го уровня состоит из наборов $(Z_1, Z_2, \ldots, Z_{128})$, сумма элементов которых начинается с $128-16$ нулей. В среднем для одного набора сумма будет нулевой. От векторов $Z_i = X_i + \texttt{Belt}_0(X_i)$ этого набора вернемся к векторам X_i. Набор $(X_1, X_2, \ldots, X_{128})$ является решением задачи.

Одно из решений, найденных описанным выше образом, представлено в криптографической библиотеке `bee2` (`https://github.com/agievich/bee2`). Решение задается массивом `_zerosum` модуля `test/belt-test.c` и проверяется функцией `beltZerosumTest()` того же модуля.