

ОБ ОДНОМ АЛГОРИТМЕ ВОССТАНОВЛЕНИЯ ОТКРЫТОГО ТЕКСТА

В.В. Герасименок, С.А. Гришутин

Белорусский государственный университет,
пр-т Независимости 4, 220030 Минск, Беларусь

Доклад посвящен анализу результатов применения раундового (итерационного) ключезависимого алгоритма MV2 [1, 2] формирования «ущербных текстов». Предлагается алгоритм восстановления открытого текста по значениям флагов без использования знаний ядра и раундовых ключей MV2 преобразования.

Пусть имеется некоторый открытый текст $M = x_1 \| x_2 \| \dots \| x_L$, представляющий собой конкатенацию слов $x_i \in \{0, 1\}^n$, и T_1, T_2, \dots, T_k — набор MV2 преобразований, где $T_i = (c_i, f_i)$ задается таблицей. Структура таблицы описана в [1]. Преобразования T_1, T_2, \dots, T_k применяются последовательно к тексту M и результатам последующей обработки.

После применения преобразования T_1 к тексту M получаем пару двоичных строк символов:

$$c_1(M) = c_1(x_1) \parallel c_1(x_2) \parallel \dots \parallel c_1(x_L), \quad f_1(M) = f_1(x_1) \parallel f_1(x_2) \parallel \dots \parallel f_1(x_L). \quad (1)$$

Значение $c_1(M)$ называется остатком, а $f_1(M)$ – флагом. Далее, текст $c_1(M)$ дополняется справа последовательностью из 1 до минимальной длины кратной n . Получаем текст $M_1 = x_1^{(1)} \parallel x_2^{(1)} \parallel \dots \parallel x_{L_1}^{(1)}$, к которому применяем преобразование T_2 и получаем соответственно:

$$c_2(M_1) = c_2(x_1^{(1)}) \parallel c_2(x_2^{(1)}) \parallel \dots \parallel c_2(x_{L_1}^{(1)}), \quad f_2(M_1) = f_2(x_1^{(1)}) \parallel f_2(x_2^{(1)}) \parallel \dots \parallel f_2(x_{L_1}^{(1)}).$$

После выполнения k процедур имеем:

$$c_k(M_{k-1}) = c_k(x_1^{(k-1)}) \parallel \dots \parallel c_k(x_{L_{k-1}}^{(k-1)}), \quad f_k(M_{k-1}) = f_k(x_1^{(k-1)}) \parallel \dots \parallel f_k(x_{L_{k-1}}^{(k-1)}).$$

Таким образом, на выходе наблюдаем последовательность $M_r = c_k(M_{k-1})$, которую называют ядром текста M , и $F_k = f_1(M) \parallel f_2(M_1) \parallel \dots \parallel f_k(M_{k-1})$ – флаги текста M .

Задача состоит в том, чтобы по последовательности флагов восстановить открытый текст M (возможно в вариантах) без знания M_r и ключевых таблиц T_1, T_2, \dots, T_k . В наблюдаемой последовательности F_k имеются существенные зависимости, которые присутствуют в исходном тексте. Эти зависимости заключаются в том, что расстояние между одинаковыми символами исходного и шифрованного текста (в некотором алфавите) сохраняются. На использовании этих зависимостей и построен алгоритм восстановления, который заключается в поиске по тексту F_k стандартов и в случае их наличия восстановление всего (или части) исходного текста M . Особенность рассматриваемого алгоритма заключается в том, что для восстановления текста достаточно рассмотреть не все F_k , а только его часть $f_1(M)$, полученную по формуле (1).

Алгоритм восстановления состоит из выполнения следующих шагов:

- определение стандартов и построение их маски;
- привязка стандарта к тексту и восстановление части таблицы T_1 ;
- восстановление исходного текста в вариантах;
- восстановление исходного текста методом зигзагообразного чтения.

На основе анализа переписки можно выделить часто встречающиеся слова и словосочетания S_i , $i = 1, 2, \dots, t$, которые называются стандартами. Стандарты рассматриваются с учетом прописных букв. Например, в качестве стандарта можно рассмотреть слово $S_1 = \#\text{институт}\#$ (вместо символа «пробел» будем использовать символ #).

Определение. Маской стандарта S_i называется набор чисел, который указывает на каких местах относительно первой буквы стандарта встречаются одинаковые символы.

Для простоты понимания процедуры привязки стандарта, рассмотрим случай, когда $n = 8$. В качестве элементов последовательности $f_1(M)$ выступают наборы символов $\{1\}$, $\{01\}$, $\{001\}$, $\{0001\}$, $\{00001\}$, $\{00000\}$. Рассмотрим текст F , полученный из наблюдаемого текста F_k применением следующей замены: $\{1\} \rightarrow A$, $\{01\} \rightarrow B$, $\{001\} \rightarrow C$, $\{0001\} \rightarrow D$, $\{00001\} \rightarrow E$, $\{00000\} \rightarrow F$.

Алгоритм привязки стандарта к тексту заключается в следующем:

- A1. Мaska стандарта накладывается на текст F начиная с первого места.
- A2. Если на указанных в маске местах буквы последовательности F совпадают, то стандарт может стоять на указанном месте, иначе нет.
- A3. Увеличивается номер начала приложения стандарта на 1. Переход к пункту A2.
- A4. Алгоритм заканчивает работу, когда длина стандарта не позволяет приложить его к очередному месту в тексте.

В результате работы алгоритма для каждого стандарта получаем варианты мест его привязки (при их наличии). По стандарту и месту его привязки в тексте F частично восстанавливаются варианты таблицы T_1' для всех букв, встречающихся в стандарте. Если привязка истинная, то найденные значения таблицы T_1' совпадают со значениями таблицы T_1 , в противном случае эти значения могут различаться.

Оценим вероятность привязки стандарта к конкретному месту. Если таблица T_1 составлялась случайным образом, то в этом случае символы последовательности F должны появляться с вероятностями: $p_1 = p(A) = 1/2$, $p_2 = p(B) = 1/4$, $p_3 = p(C) = 1/8$, $p_4 = p(D) = 1/16$, $p_5 = p(E) = p_6 = p(F) = 1/32$. Если маска стандарта состоит из набора k повторов с длинами l_j , то вероятность ложной привязки стандарта по маске равна

$$p_{st} = \prod_{j=1}^k \left(\sum_{i=1}^6 (p_i)^{l_j} \right).$$

В частности, для стандарта $S_1 = \#\text{институт}\# - p_{S_1} = 0.015937$, а для стандарта $S_2 = \#\text{институт}\#\text{криптографии}\# - p_{S_2} = 0.000102$. Полученное значение указывает на то, что в среднем ложная привязка стандарта S_2 происходит раз на 10 000 мест приложений.

В результате привязки стандартов частично восстанавливается таблица T_1 . Пусть в таблице практически для всех русских строчных букв восстановлены заполнения третьего столбца (т.е. известны флаги букв). Тогда для каждого значения последовательности $f_1(M)$ можно выписать возможное значение буквы открытого текста (в вариантах), что соответствует наложению некачественной гаммы. Применяя метод зигзагообразного чтения можно сделать попытку восстановить исходный текст (возможно частично и в вариантах). При восстановлении части открытого текста одновременно восстанавливаются значения третьего столбца таблицы T_1 .

Восстановление открытого текста будет эффективным, если выполнено условие:

$$H_\Gamma + H_{\text{от}} < \log_a z, \quad (2)$$

где H_Γ – энтропия гаммы, $H_{\text{от}}$ – энтропия открытого текста на знак, z – модуль алфавита, a – основание логарифма (можно положить $a = 2$). С учетом того, что $H_\Gamma < 3.125$, а $0.83 \leq H_{\text{от}} \leq 1.40$ [3], $a = 2$, $z = 32$, то формула (2) выполняется.

Литература

1. Мищенко В.А., Виланский Ю.В. Ущербные тексты и многоканальная криптография. Мн.: Энциклопедикс. 2007. 292 с.
2. Мищенко В.А., Виланский Ю.В., Лепин В.В. Криптографический алгоритм MV2. Мн.: Энциклопедикс. 2007. 176 с.
3. Алферов А.П. и др. Основы криптографии. М.: Гелиос-АРВ. 2001. 480 с.