Лекция 4. Аналитический метод

Цель лекции: рассмотреть универсальный аналитический метод криптоанализа.

План лекции:
Введение.
1 Аналитический метод
Заключение
Контрольные вопросы

Ключевые слова: [выбрать самостоятельно].

Содержание лекции:

Введение

1 Аналитический метод.

Пусть множество ключей К представимо в виде прямого произведения множеств

$$К = К_1 \times К_2 \times ... \times К_r.$$

Например, множество ключей DES К = $\{0, 1\}^{56}$. Пусть дешифровщик получил открытое сообщение $x = x_1x_2\ldots x_s$ и шифрсообщение $y= y_1y_2\ldots y_s$, где для простоты считаем, что x и y слова длины s в одном алфавите А, т.е. $x_i, y_i \in А$. Тогда можно составить систему уравнений шифрования, выписанную для Т в каком-либо базисе операций:

$$\begin{gathered} y_1 = f_1(x_1, \ldots, x_s, k_1, \ldots, k_r) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ y_s = f_s(x_1, \ldots, x_s, k_1, \ldots, k_r) \end{gathered} \qquad (1)$$

В этих уравнениях при известных f, x, y остаются неизвестными только $k_1, ..., k_r$. Отметим, что $|К| = |К_1|\times|К_2|\times...\times|К_r|$ и полный перебор требует |К|/2 опробований.

Анализ получившейся системы (1) может подсказать более быстрые способы ее решения, чем полный перебор.

Пример 1. Пусть система (1) может быть преобразована к виду:

$$\begin{gathered} g_1(x, y) = h_1(x, y, k_1) \\ g_2(x, y) = h_2(x, y, k_1, k_2) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ g_r(x, y) = h_r(x, y, k_1, \ldots, k_r). \end{gathered} \qquad (2)$$

Тогда [[1], стр. 397], опробуя $|К_1|$ вариантов часть ключа $k_1 \in К_1$ и используя для проверки правильности первое уравнение в (2), мы можем восстановить k_1. Подставив полученное значение во второе уравнение мы используем его для проверки правильности варианта для $k_2 \in К_2$, затратив при этом не более $|К_2|$ операций опробования. Найдя k_2, мы подставим его в третье уравнение и т.д. В результате весь ключ $k = (k_1, k_2, \ldots, k_r)$ мы получим, проведя не более, чем $|К_1| + |К_2| + ... + |К_r|$ опробований. Это число, как правило, значительно меньше, чем $|К|/2 = (|К_1|\times|К_2|\times\ldots\times|К_r|)/2$. Налример, при $|К_1| = |К_2|$, r = 2, трудоемкость метода ~ $2\sqrt{|K|}$. В этом случае для 56-битного ключа вместо $2^{56}/2 \approx 10^{17}/2$

[1] Шеннон К. Работы по теории информации и кибернетике, М.: Иностранная литература, 1963.

достаточно выполнить $2 \cdot 2^{28} < 10^9$ операций.

Пример 2. Значительное упрощение задачи дешифрования мы получаем, когда в системе (1) мы каким-либо образом можем выделить линейную подсистему уравнений. В этом случае можно применить известные методы решения систем линейных уравнений, которые по трудоемкости и памяти значительно меньше переборных методов. В самом деле, пусть в (1) выделена подсистема

$$g_i(x, y) = \sum_{j=1}^{r} b_{ij} k_j, \quad i = 1, \ldots, t.$$

Применим для ее решения метод Гаусса и найдем трудоемкость для случая, когда $g_i(\cdot) \in \{0, 1\}$ и линейная система рассматривается над GF(2).

Если система линейных уравнений с r неизвестными совместна и имеет вид

$$\sum_{j=1}^{r} b_{ij} k_j = c_i, \quad i = 1, \ldots, r.$$

то ее решение методом Гаусса предполагает на первом этапе выделение строк с $b_{i1} = 1$ не более r операций, вычитание из первой строки с $b_{i1} = 1$ остальных строк с $b_{m1} = 1$ не более $(r + 1)(r - 1)$ операций. Итого получаем оценку r^2 операций на первом этапе. На втором этапе делаем то же самое с подсистемой

$$\sum_{j=2}^{r} b_{ij} k_j = c_i^{(1)}{}_i.$$

Оценка трудоемкости здесь $(r - 1)^2$ операций. Продолжая этот алгоритм, в результате получим трудоемкость метода

$$\sum_{k=1}^{r} k^2 = \frac{r(r+1)(2r+1)}{6} \approx \frac{r^3}{3}.$$

Таким образом, нахождение 64-битного ключа методом полного перебора составило бы $2^{64} \approx 10^{19}$ операций, а, если задача сведена к решению линейной системы уравнений, то трудоемкость не превосходит $64^3/3 \approx 8{,}7 \cdot 10^4$ операций.

Пример 3. В книге Гилла [2] линейные рекуррентные последовательности над GF(2) рекомендуются в качестве хорошего шифра. Линейная рекуррента описывается законом

$$\gamma_{n+t} = \sum_{i=0}^{t-1} \alpha_i \gamma_{n+i}, \quad n = 0,1, \ldots, \qquad (3)$$

где сложение и умножение осуществляется в GF(2).

Иногда говорят, что такие рекуррентные последовательности порождаются линейными регистрами сдвига (регистром сдвига с линейной обратной связью). Последовательность $\{\gamma_s, s = 0, 1, \ldots\}$ может использоваться в шифре гаммирования по mod 2, ключом является начальное заполнение регистра сдвига $(\gamma_0, \gamma_1, \ldots, \gamma_{t-1}) = \gamma(0)$. Нетрудно видеть, что существует матрица A такая, что любой вектор последовательных t значений

[2] Гилл А. Линейные последовательные машины. Анализ, синтез и применение. М.: Наука, 1974.

рекурренты равен степени матрицы A, умноженной на γ(0). Если закон рекурренты известен, то задача определения ключа сводится к решению системы линейных уравнений размера t×t и поэтому данный шифр нестойкий.

Если линейная рекуррента неизвестна, то при известной γ получаем линейную систему относительно коэффициентов α_i, решив ее, можем узнать ключ γ(0) [3].

Пример 4. Пусть двоичная гамма в шифре гаммирования по mod 2 получена при помощи нелинейной рекурренты (при помощи регистра сдвига с нелинейной обратной связью или просто нелинейного регистра сдвига)

$$\gamma_{n+t} = f(\gamma_n, \gamma_{n+1}, \dots, \gamma_{n+t-1}), \quad n = 0,1, \dots . \tag{4}$$

Пусть f такова, что при любом начальном заполнении регистра последовательность периодична, тогда порождающий ее нелинейный регистр сдвига называется регулярным. Тогда оказывается, что для любого начального заполнения существует линейная рекуррента вида (3) размера ν (ν ≥ t) такая, что порождаемая ею последовательность совпадает с последовательностью, порождаемой при этом начальном заполнении регистром сдвига с нелинейной обратной связью.

Доказательство существования линейной рекурренты очевидно, так как, если последовательность $\gamma_0, \gamma_1, \dots, \gamma_T, \gamma_{T+1}, \dots$ периодична с периодом T, то линейный регистр сдвига вида $\gamma_{n+T} = \gamma_n$, n = 0, 1, ..., порождает ту же последовательность, что (4).

Определение 1. Длина минимального линейного регистра сдвига, порождающего данную периодическую последовательность (регулярного нелинейного регистра сдвига), называется линейной сложностью регулярного нелинейного регистра сдвига.

Если гамма для шифра гаммирования порождается регулярным нелинейным регистром сдвига с линейной сложностью T, и ключом является начальное заполнение регистра, то из существования линейного регистра сдвига длины T, порождающего данную гамму, следует, что нахождение ключа сводится к линейной системе размера T. Откуда сложность дешифрования не превосходит $T^3/3$. Однако построение эквивалентного минимального линейного регистра сдвига по последовательности гаммы, не является простой задачей. Вместе с тем, чем меньше линейная сложность, тем проще задача дешифрования. Для полноты картины необходимо найти конструктивные условия регулярности нелинейного регистра сдвига. Докажем следующую теорему.

Теорема 1. Нелинейный регистр сдвига (4) над GF(2) регулярен тогда и только тогда, когда

$$f(\gamma_n, \gamma_{n+1}, \dots, \gamma_{n+t-1}) = \gamma_n + g(\gamma_{n+1}, \dots, \gamma_{n+t-1}) \tag{5}$$

Доказательство. Если f имеет структуру (5), тогда для любой цепочки $x_2, \dots, x_t$

$$f(0, x_2, \dots, x_t) = 1 + f(1, x_2, \dots, x_t). \tag{6}$$

Обратно, если при всех $x_2, \dots, x_t$ выполняется (6), то имеет место представление (5). В самом деле, обозначив через $g(x_2, \dots, x_t) = f(0, x_2, \dots, x_t)$, получим (5). Всегда из (6) следует, что

$$\begin{gathered} f(x_1, x_2, \dots, x_t) = x_1 f(1, x_2, \dots, x_t) + (1 + x_1) f(0, x_2, \dots, x_t) = \\ = x_1 \big(f(0, x_2, \dots, x_t) + f(1, x_2, \dots, x_t)\big) + f(0, x_2, \dots, x_t) = \\ = x_1 + f(0, x_2, \dots, x_t). \end{gathered}$$

[3] Meyer C., Matyas S. Cryptography: a New Dimension in Computer Data Security. John Wiley & Sons, 1982.

Рекуррента определяет отображение множества t-мерных векторов в себя

$$F\colon (x_1, \dots, x_t) \longrightarrow \big(x_2, \dots, x_t, f(x_1, \dots, x_t)\big)$$

Рекуррента регулярна тогда и только тогда, когда F – взаимно-однозначное соответствие. Если верно (5) и существуют два вектора, переходящие при F в один, то они могут отличаться только в x_1 и

$$f(0, x_2, \dots, x_t) = f(1, x_2, \dots, x_t),$$

что противоречит (6). Обратно, при регулярности (4) имеем взаимно-однозначное отображение F, то есть для любых (x_2, …, x_t) векторы

$$(x_2, \dots, x_t, f(0, x_2, \dots, x_t)) \neq \big(x_2, \dots, x_t, f(1, x_2, \dots, x_t)\big).$$

Следовательно,

$$f(0, x_2, \dots, x_t) = 1 + f(1, x_2, \dots, x_t),$$

что и требовалось доказать.

Заключение

Контрольные вопросы

Смотри руководство по организации самостоятельной работы магистрантов.