Практическое задание №1

Нахождение прообраза функции сжатия MD2

В данном задании Вам необходимо реализовать алгоритм нахождения прообраза итеративной хэш-функции MD2 [1], описанный в статье [2].

Обратите внимание на ошибку в статье [2]: на Рис. 2 указано, что A_0^i вычисляется как $\phi(A_0^{i-1}, C_{15}^i + i - 1)$. В оригинальном же алгоритме $A_0^i = \phi(A_0^{i-1}, C_{15}^i + i - 2)$.

Так как вычислительная сложность данного алгоритма слишком высока для реализации на персональном компьютере, то мы рассмотрим вариант алгоритма MD2 (и, соответственно, алгоритма из [2]), работающий с 2-битными "байтами". Отличия данного варианта алгоритмов состоят в:

1. S-боксе (перестановке на $\mathbb{F}_2^2$), который в данном варианте задается следующей таблицей (здесь и далее элемент $(x_0, x_1) \in \mathbb{F}_2^2$ отождествляется с числом $x_0 + 2x_1$):

0	1	2	3
1	3	0	2

2. Padding Rule. В нашем варианте сообщение дополняется i байтами со значением (i mod 4) до длины, кратной 16 (т.е. вместо байта i используется (i mod 4)).

В качестве решения необходимо представить:

1. Программу на одном из языков программирования C, C++, Python, которая может работать в 3-х режимах:

   (a) Вычисление MD2 для заданной последовательности 16-байтных блоков. Пример вызова:
   `./your-program md2 "0 3 1 2 1 1 0 2 0 3 3 0 1 1 2 0" "3 2 2 2 2 2 1 0 0 3 3 0 1 2 3 0"`
   Ожидаемый вывод:
   `1 2 1 2 1 2 3 2 0 3 0 3 0 3 2 3`

   (b) Вычисление функции сжатия ($H_{i+1} = F(H_i, M_i)$) для заданного H_i и блока сообщения M_i. Пример вызова:
   `./your-program compress "1 3 2 2 0 2 1 0 0 3 3 0 1 2 3 0" "1 2 0 2 3 1 0 2 0 3 3 0 1 1 2 0"`
   Ожидаемый вывод:
   `3 3 0 0 1 3 1 2 0 0 2 3 3 3 0 1`

   (c) Вычисление прообраза (т.е. блока сообщения M_i) для заданных H_i и H_{i+1}. Пример вызова:
   `./your-program preimage "0 1 2 3 0 1 2 3 0 1 2 3 0 1 2 3" "0 0 3 2 2 2 1 0 0 0 3 0 3 3 1 2"`
   Ожидаемый вывод:
   `0 0 3 2 2 2 1 0 0 0 3 0 3 3 1 2`

2. Отчет (документ в формате pdf или doc), включающий в себя:

   (a) Описание алгоритма MD2 (по разобранному на лекции шаблону).

   (b) Краткое описание алгоритма атаки (своими словами).

   (c) Теоретическая оценка времени работы и требуемой Вашей программой памяти; замеры реального времени работы.

3. Ответ (т.е. любое M_i, удовлетворяющее $H_{i+1} = F(H_i, M_i)$) для Вашего варианта.

Последний день сдачи — 22 сентября 2019 года.

Варианты

1. H_i ="3 0 0 0 3 2 1 2 0 1 0 2 0 3 3 0",
   M_i ="3 1 2 0 0 3 0 0 1 0 0 1 0 0 3 2"

2. H_i ="3 2 0 3 2 0 1 0 3 2 2 3 3 0 3 0",
   M_i ="0 0 1 2 2 3 2 2 0 1 3 1 0 3 3 1"

3. H_i ="3 3 0 0 0 2 3 2 1 2 0 0 3 2 1 1",
   M_i ="1 3 0 2 1 1 3 2 3 1 2 2 1 2 0 1"

4. H_i ="2 0 2 0 0 2 3 3 1 3 0 1 1 3 2 3",
   M_i ="0 0 3 1 2 2 0 0 0 2 1 0 0 1 2 0"

5. H_i ="1 2 2 2 3 3 2 0 1 2 3 0 1 3 3 2",
   M_i ="2 0 1 3 3 1 0 0 0 3 2 0 3 1 0 1"

6. H_i ="3 0 0 1 0 1 1 2 1 1 2 3 3 0 1 0",
   M_i ="0 2 1 2 0 2 0 1 0 1 3 1 0 3 1 1"

7. H_i ="3 0 2 2 1 0 0 0 2 3 1 1 2 2 0 1",
   M_i ="3 1 0 2 0 3 1 2 2 0 2 2 0 2 0 1"

8. H_i ="1 3 3 1 3 3 3 0 1 2 2 3 0 0 1 1",
   M_i ="3 3 1 3 3 1 1 0 1 3 2 3 2 1 3 2"

9. H_i ="3 2 2 1 0 0 2 3 2 2 0 3 3 2 2 0",
   M_i ="3 3 0 3 1 2 2 0 1 3 0 3 3 3 1 1"

10. H_i ="3 1 1 2 2 1 0 2 1 0 0 2 3 0 3 1",
    M_i ="2 2 2 1 1 1 0 2 0 3 1 2 2 0 1 3"

11. H_i ="2 2 3 0 0 1 1 3 2 2 3 3 2 3 0 0",
    M_i ="2 3 0 2 1 2 1 0 2 3 0 3 2 2 2 0"

12. H_i ="0 2 2 3 2 3 0 2 1 1 0 0 2 2 2 0",
    M_i ="3 0 0 2 3 0 2 3 2 3 3 0 1 0 3 2"

13. H_i ="2 0 2 1 2 3 1 0 2 3 1 3 0 2 2 0",
    M_i ="3 3 0 3 3 1 3 0 0 2 0 2 0 2 2 0"

14. H_i ="2 0 2 1 3 2 3 2 3 3 2 3 1 0 1 1",
    M_i ="1 2 2 2 0 1 2 2 0 0 3 0 2 2 3 1"

15. H_i ="0 2 3 3 3 1 1 1 3 0 2 1 3 2 3 3",
    M_i ="2 2 0 0 2 0 2 0 0 1 3 3 0 3 3 2"

16. H_i ="3 2 2 2 2 0 0 2 0 3 2 0 1 0 3 1",
    M_i ="0 0 3 3 0 2 1 1 0 3 3 0 2 2 1 1"

17. H_i ="0 2 3 0 0 3 0 0 0 1 0 0 2 0 2 1",
    M_i ="2 2 0 3 1 1 0 0 1 0 0 3 2 1 0 2"

18. H_i ="2 3 3 0 3 0 2 2 3 0 3 1 2 0 3 1",
    M_i ="1 1 3 0 1 1 2 2 0 3 2 1 0 3 3 0"

Список литературы

[1] https://tools.ietf.org/html/rfc1319

[2] The MD2 Hash Function Is Not One-Way. Frédéric Muller