

Лабораторная работа №7

Элементы криптографии. Однократное гаммирование

Монастырская Кристина Владимировна

Содержание

Цель работы	4
Выполнение лабораторной работы	5
Выводы	9
Список литературы	10

Список иллюстраций

1	Функция <code>get_key</code>	7
2	Результат работы программы	8

Цель работы

Освоить на практике применение режима однократного гаммирования. [1]

Выполнение лабораторной работы

Разработала приложение, позволяющее шифровать и дешифровать данные в режиме однократного гаммирования.

Приложение может: 1. Определять вид шифротекста при известном ключе и известном открытом тексте. 2. Определить ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой один из возможных вариантов прочтения открытого текста.

Для реализации первого пункта реализовала функцию encryption ((рис. 1) В данной функции реализовано преобразование исходного текста и ключа в двоичный код в виде строки, а далее с этими строками происходит сложение по модулю 2. Благодаря этим действиям мы получили шифротекст в двоичном коде, который мы преобразовали в шестнадцатичную строку.

```

11 def encryption(phrase, key):
12     hex_phrase = phrase.encode('utf-8').hex()
13     bin_phrase = toBinary(hex_phrase)
14
15     hex_key = key.replace(' ', '').lower()
16     bin_key = toBinary(hex_key)
17
18     bin_cipher = ""
19     if len(bin_phrase) == len(bin_key):
20         for i in range(len(bin_phrase)):
21             if bin_phrase[i] == ' ':
22                 bin_cipher += ' '
23             elif bin_phrase[i] == bin_key[i]:
24                 bin_cipher += '0'
25             elif bin_phrase[i] != bin_key[i]:
26                 bin_cipher += '1'
27
28     bin_cipher = bin_cipher.split()
29     cipher = ''
30     for i in range(len(bin_cipher)):
31         x = int(bin_cipher[i], 2)
32         cipher += hex(x)[2:]
33         cipher += ' '
34
35     cipher = cipher.upper()
36     return cipher
37

```

Для реализации второго пункта реализовала функцию `get_key` ((рис. 2), в которой нашла ключ преобразующий шифротекст в какой-либо заданный открытый текст. Функция реализована схожим образом: открытый итоговый текст и шифротекст преобразовываются в двоичный код, далее производится операция подбора значений ключа исходя из шифротекста и итогового текста. Получается аналогичное сложение по модулю 2. После этого ключ преобразуется в шестнадцатичный формат.

```

39 def get_key(cipher, res):
40     hex_cipher = cipher.replace(' ', '').lower()
41     bin_cipher = toBinary(hex_cipher)
42
43     hex_res = res.encode('utf-8').hex()
44     bin_res = toBinary(hex_res)
45
46     bin_key = ''
47     if len(bin_cipher) == len(bin_res):
48         for i in range(len(bin_cipher)):
49             if bin_cipher[i] == ' ':
50                 bin_key += ' '
51             elif bin_cipher[i] == bin_res[i]:
52                 bin_key += '0'
53             elif bin_cipher[i] != bin_res[i]:
54                 bin_key += '1'
55
56     bin_key = bin_key.split()
57     hex_key = ''
58     for i in range(len(bin_key)):
59         x = int(bin_key[i], 2)
60         if len(hex(x)[2:]) < 2:
61             hex_key += '0'
62             hex_key += hex(x)[2:]
63             hex_key += ' '
64     return hex_key
65

```

Рис. 1: Функция get_key

Результат работы программы ((рис. 3):

```
66 |
67 | phrase = 'Штирлиц - Вы Герой!!!'
68 | key = '05 0C 17 7F 0E 4E 37 D2 94 10 09 2E 22 57 FF C8 08 82 70 54 05 0C 17 7F 0E 4E 37 D2 94 10 09 2E 22 57 FF C8 08 02 94'
69 | res = 'С Новым Годом, друзья!'
70 |
71 |
72 | print('Шифрослово: ', encryption(phrase, key))
73 | print('Ключ: ', get_key(encryption(phrase, key), res))
74 |
```

main x

C:\Users\krist\PycharmProjects\infosec\venv\Scripts\python.exe C:/Users/krist/PycharmProjects/infosec/main.py

Шифрослово: 05 A4 C6 FD DE F6 E6 52 44 AB D9 96 F3 D1 DF E8 E9 32 E3 74 25 DC 85 AE 85 6E E7 41 44 A5 D8 AE F2 E9 2F 71 91 F3 B5

Ключ: 05 05 e6 2d 43 26 58 82 f6 7a 52 46 4f f1 0f 7b 39 8c 33 c8 f5 62 55 12 a9 4e 37 f5 95 25 09 2d 22 5e fe fd 40 7c 94

Process finished with exit code 0

Рис. 2: Результат работы программы

Выводы

Освоить на практике применение режима однократного гаммирования.

Список литературы

1. Основы безопасности информационных систем : Учеб. пособие для студентов вузов, обучающихся по специальностям “Компьютер. безопасность” и “Комплекс. обеспечение информ. безопасности автоматизир. систем” / Д.А. Зегжда, А.М. Ивашко. - М. : Горячая линия - Телеком, 2000. - 449, [2] с. : ил., табл.; 21 см.; ISBN 5-93517-018-3.