

Лабораторная работа №8

Элементы криптографии. Шифрование (кодирование)
различных исходных текстов одним ключом

Монастырская Кристина Владимировна НПИБд-02-19¹

2022, 19 March, Moscow, Russian Federation

¹RUDN University, Moscow, Russian Federation

Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

Задание

Два текста кодируются одним ключом (однократное гаммирование).

$P_1 = \text{НаВашисходящийот1204}$

$P_2 = \text{ВСеверныйфилиалБанка}$

Требуется не зная ключа и не стремясь его определить, прочитать оба текста.

Необходимо разработать приложение, позволяющее шифровать и дешифровать тексты P_1 и P_2 в режиме однократного гаммирования.

Приложение должно определить вид шифротекстов C_1 и C_2 обоих текстов P_1 и P_2 при известном ключе и P_2 при известном ключе, а также должно дешифровать один из исходных текстов без использования ключа.

Функция шифровки двух текстов с известным ключом

Функция шифровки двух текстов с известным ключом

```
6 def encryption(P, K):
7     hex_P = P.encode('utf-8').hex()
8     bin_P = toBinary(hex_P)
9     hex_K = K.replace(' ', '').lower()
10    bin_K = toBinary(hex_K)
11
12    bin_C = ""
13    if len(bin_P) == len(bin_K):
14        for i in range(len(bin_P)):
15            if bin_P[i] == ' ':
16                bin_C += ' '
17            elif bin_P[i] == bin_K[i]:
18                bin_C += '0'
19            elif bin_P[i] != bin_K[i]:
20                bin_C += '1'
21    bin_C = bin_C.split()
22    C = ''
23    for i in range(len(bin_C)):
24        x = int(bin_C[i], 2)
25        if len(hex(x)[2:]) < 2:
26            C += '0'
27        C += hex(x)[2:]
28        C += ' '
29
30    C = C.upper()
31    return C
32
```

Рис. 1: Шифровка текстов при известном ключе

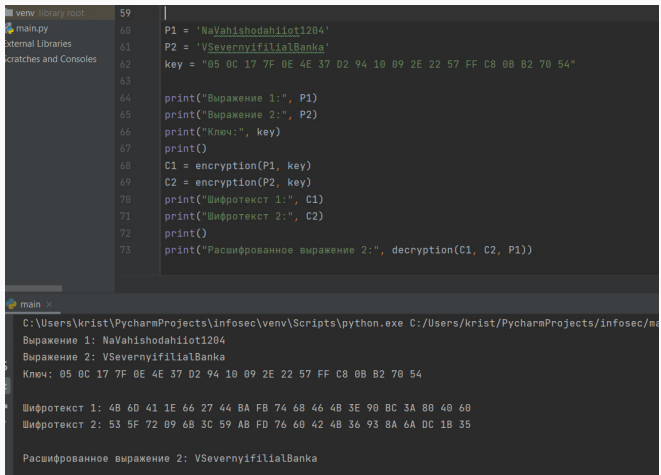
Функция дешифровки сообщений без
знания ключа.

Функция дешифровки сообщений без знания ключа.

```
34 def decryption(C1, C2, P1):
35     hex_C1 = C1.replace(' ', '').lower()
36     bin_C1 = toBinary(hex_C1)
37     hex_C2 = C2.replace(' ', '').lower()
38     bin_C2 = toBinary(hex_C2)
39     hex_P1 = P1.encode('utf-8').hex()
40     bin_P1 = toBinary(hex_P1)
41     bin_P2 = ''
42     for i in range(len(bin_P1)):
43         if bin_P1[i] == ' ':
44             bin_P2 += ' '
45         else:
46             bin_P2 += str(int(bin_C1[i]) ^ int(bin_C2[i]) ^ int(bin_P1[i]))
47
48     bin_P2 = bin_P2.split()
49     P2 = ''
50     for i in range(len(bin_P2)):
51         x = int(bin_P2[i], 2)
52         if len(hex(x)[2:]) < 2:
53             P2 += '0'
54         P2 += hex(x)[2:]
55         P2 += ' '
56     P2 = bytes.fromhex(P2).decode("ASCII")
57     return P2
58
```

Рис. 2: Дешифровка сообщений без ключа

Итог



```
59 |
60 P1 = 'NaVahishodahiiot1204'
61 P2 = 'VSevernyifilialBanka'
62 key = "05 0C 17 7F 0E 4E 37 D2 94 10 09 2E 22 57 FF C8 0B 82 70 54"
63
64 print("Выражение 1:", P1)
65 print("Выражение 2:", P2)
66 print("Ключ:", key)
67 print()
68 C1 = encryption(P1, key)
69 C2 = encryption(P2, key)
70 print("Шифротекст 1:", C1)
71 print("Шифротекст 2:", C2)
72 print()
73 print("Расшифрованное выражение 2:", decryption(C1, C2, P1))
```

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

Выражение 1: NaVahishodahiiot1204
Выражение 2: VSevernyifilialBanka
Ключ: 05 0C 17 7F 0E 4E 37 D2 94 10 09 2E 22 57 FF C8 0B 82 70 54

Шифротекст 1: 4B 6D 41 1E 66 27 44 BA FB 74 68 46 4B 3E 90 BC 3A 80 40 60
Шифротекст 2: 53 5F 72 09 6B 3C 59 AB FD 76 60 42 4B 36 93 8A 6A DC 1B 35

Расшифрованное выражение 2: VSevernyifilialBanka

Рис. 3: Вывод программы

Вывод

Освоили на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.