Отчёт по лабораторной работе

Лабораторная работа № 7

Живцова Анна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Освоить на практике применение режима однократного гаммирования.

2 Задание

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

3 Теоретическое введение

Предложенная Г. С. Вернамом так называемая «схема однократного использования (гаммирования)» является простой, но надёжной схемой шифрования данных.

Гаммирование представляет собой наложение (снятие) на открытые (зашифрованные) данные последовательности элементов других данных, полученной с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных. Иными словами, наложение гаммы — это сложение её элементов с элементами открытого (закрытого) текста по некоторому фиксированному модулю, значение которого представляет собой известную часть алгоритма шифрования.

В соответствии с теорией криптоанализа, если в методе шифрования используется однократная вероятностная гамма (однократное гаммирование) той же длины, что и подлежащий сокрытию текст, то текст нельзя раскрыть [1].

4 Выполнение лабораторной работы

1. Создана программа, шифрующая сообщение по ключу.

2. Создана программа, определяющая ключ по шифру и исходному тексту.

3. Программа проверена на данных из пособия (см. рис. 4.1)

In [103]:
```
coding_const = 1040 - 192
#Определить вид исходного текста при известном ключе и известном шифротексте
def encrypt(key, text):
    key = key.split(' ')
    text = text.split(' ')
    return  ''.join(chr((int(key[i], 16) ^ int(text[i], 16)) + coding_const) if int(key[i], 16) ^ int(text[i], 16) > 192 else chr
```

In [104]:
```
#Определить ключ при известном шифротексте и известном открытом тексте
def find_key(text1, text2):
    text1 = [hex(ord(i) - coding_const) if ord(i) > coding_const else hex(ord(i)) for i in text1]
    text2 = text2.split(' ')
    return  ' '.join(format(int(text1[i], 16) ^ int(text2[i], 16), 'X') for i in range(len(text1)))
```

In [105]:
```
key = '05 0C 17 7F 0E 4E 37 D2 94 10 09 2E 22 57 FF C8 0B B2 70 54'
text = 'DD FE FF 8F E5 A6 C1 F2 B9 30 CB D5 02 94 1A 38 E5 5B 51 75'
encrypt(key, text)
```

Out[105]:
```
'Штирлиц - Вы Герой!!'
```

In [110]:
```
text2 = 'DD FE FF 8F E5 A6 C1 F2 B9 30 CB D5 02 94 1A 38 E5 5B 51 75'
#text1 = 'С Новым Годом, Ура!!'
text1 = 'Штирлиц - Вы Болван!'
find_key(text1, text2)
```

Out[110]:
```
'5 C 17 7F E 4E 37 D2 94 10 9 2E 22 55 F4 D3 7 BB BC 54'
```

Рис. 4.1: Программа шифрования

5 Выводы

Успешно реализовали метод однократного гаммирования

Список литературы

1. А.А. Аргановский, Р.А.Хади. Практическая криптография: алгоритмы и их программирование. солон пресс, 2009.