Лабораторная работа №8

Кондрашина Мария Сергеевна[1]

18.10.2022, Moscow

[1]RUDN University, Moscow, Russian Federation

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

Цель работы

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

Выполнение лабораторной работы

Задание

Два текста кодируются одним ключом (однократное гаммирование).

Требуется не зная ключа и не стремясь его определить, прочитать оба текста. Необходимо разработать приложение, позволяющее шифровать и дешифровать тексты P1 и P2 в режиме однократного гаммирования. Приложение должно определить вид шифротекстов C1 и C2 обоих текстов P1 и P2 при известном ключе;

Необходимо определить и выразить аналитически способ, при котором злоумышленник может прочитать оба текста, не зная ключа и не стремясь его определить

Перевод открытых текстов в шестнадцатеричное представление.

1. Два текста кодируются одним ключом (однократное гаммирование). Требуется не зная ключа и не стремясь его определить, прочитать оба текста. Необходимо разработать приложение, позволяющее шифровать и дешифровать тексты P1 и P2 в режиме однократного гаммирования. Приложение должно определить вид шифротекстов C1 и C2 обоих текстов P1 и P2 при известном ключе

Ввод [19]:
```
t1 = 'НаВашисходящийот1204'
t2 = 'ВСеверныйфилиалБанка'
print("Открытые тексты: ", t1, ", ", t2)
```

```
Открытые тексты:  НаВашисходящийот1204 ,  ВСеверныйфилиалБанка
```

Ввод [20]:
```
def to_16(strM):
    str_16 = [(i.encode('cp1251')).hex().upper() for i in strM]
    return str_16
```

Ввод [21]:
```
t1_16 = to_16(t1)
t2_16 = to_16(t2)
print("Открытые тексты в шестнадцатеричном представлении: \n", t1_16, ", \n", t2_16)
```

```
Открытые тексты в шестнадцатеричной представлении:
 ['CD', 'E0', 'C2', 'E0', 'F8', 'E8', 'F1', 'F5', 'EE', 'E4', 'FF', 'F9', 'E8', 'E9', 'EE', 'F2', '31', '32', '30', '34'] ,
 ['C2', 'D1', 'E5', 'E2', 'E5', 'F0', 'ED', 'FB', 'E9', 'F4', 'E8', 'EB', 'E8', 'E0', 'EB', 'C1', 'E0', 'ED', 'EA', 'E0']
```

Создание ключа и шифрование текстов (шестнадцатеричное представление

Ввод [22]:
```
import random

key = [hex(random.randint(0,255))[2:].upper() for _ in range(len(t1))]
print("Ключ: ", key)
```

```
Ключ:  ['51', 'CF', '88', '42', 'E5', '4D', '11', '50', '39', 'D', 'F3', '10', 'F6', '40', 'D5', 'E1', '48', 'CB', '7E', '46']
```

Ввод [23]:
```
def cipher_text(strM, keyM):
    c_text = [('{:02x}'.format(int(i,16) ^ int(k, 16))).upper() for i,k in zip(strM, keyM)]
    return c_text
```

Ввод [30]:
```
t1_cipher_16 = cipher_text(t1_16, key)
t2_cipher_16 = cipher_text(t2_16, key)
print("Зашифрованные тексты в шестнадцатеричном представлении:  \n", t1_cipher_16, ", \n", t2_cipher_16)
```

```
Зашифрованные тексты в шестнадцатеричном представлении:
 ['9C', '2F', '4A', 'A2', '1D', 'A5', 'E0', 'A5', 'D7', 'E9', '0C', 'E9', '1E', 'A9', '3B', '13', '79', 'F9', '4E', '72'] ,
 ['93', '1E', '6D', 'A0', '00', 'BD', 'FC', 'AB', 'D0', 'F9', '1B', 'FB', '1E', 'A0', '3E', '20', 'A8', '26', '94', 'A6']
```

Зашиврованные тексты

Ввод [31]:
```
def cipher_text_al(strM):
    al_text = [(bytes.fromhex(el)).decode('cp1251') for el in strM]
    return al_text
```

Ввод [33]:
```
t1_cipher_al = cipher_text_al(t1_cipher_16)
t2_cipher_al = cipher_text_al(t2_cipher_16)
print("Зашифрованные тексты: \n", t1_cipher_al, ", \n", t2_cipher_al)
```

```
Зашифрованные тексты:
 ['њ', '/', 'Ј', 'ў', '\x1d', 'Ґ', 'а', 'Ґ', 'Ч', 'й', '\x0c', 'й', '\x1e', '©', ';', '\x13', 'у', 'щ', 'N', 'r'] ,
 ['“', '\x1e', 'm', '\xa0', '\x00', 'S', 'ь', '«', 'P', 'щ', '\x1b', 'ы', '\x1e', '\xa0', '>', ' ', 'Ё', '&', '”', '¦']
```

Определение открытого текста по второму отрытому тексту и зашифрованным текстам (без использования ключа)

2. Необходимо определить и выразить аналитически способ, при котором злоумышленник может прочитать оба текста, не зная ключа и не стремясь его определить.

Ввод [38]:
```
def find_t2 (t1_c16, t2_c16, t1):
    t2_f = [('{:02x}'.format(int(c1,16) ^ int(c2, 16) ^ int(p1, 16))).upper() for c1,c2,p1 in zip(t1_c16, t2_c16, t1)]
    return t2_f
```

Ввод [43]:
```
t2_f_new = find_t2(t1_cipher_16, t2_cipher_16, t1_16)
print("Поиск второго текста по известным шифрованным текстам и открытому первому тексту")
print(t2_f_new)
print(cipher_text_al(t2_f_new))

print("\nПоиск первого текста по известным шифрованным текстам и открытому второму тексту")
t1_f_new = find_t2(t1_cipher_16, t2_cipher_16, t2_16)
print(t1_f_new)
print(cipher_text_al(t1_f_new))
```

```
Поиск второго текста по известным шифрованным текстам и открытому первому тексту
['C2', 'D1', 'E5', 'E2', 'E5', 'F0', 'ED', 'FB', 'E9', 'F4', 'E8', 'EB', 'E8', 'E0', 'EB', 'C1', 'E0', 'ED', 'EA', 'E0']
['В', 'С', 'е', 'в', 'е', 'р', 'н', 'ы', 'й', 'ф', 'и', 'л', 'и', 'а', 'л', 'Б', 'а', 'н', 'к', 'а']

Поиск первого текста по известным шифрованным текстам и открытому второму тексту
['CD', 'E0', 'C2', 'E0', 'F8', 'E8', 'F1', 'F5', 'EE', 'E4', 'FF', 'F9', 'E8', 'E9', 'EE', 'F2', '31', '32', '30', '34']
['Н', 'а', 'В', 'а', 'ш', 'и', 'с', 'х', 'о', 'д', 'я', 'щ', 'и', 'й', 'о', 'т', '1', '2', '0', '4']
```

Результат

- Выполнила лабораторную работу №8.
- Освоила на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом

Список литературы

1. Методические материалы курса. “Информационная безопасность компьютерных сетей” Кулябов Д. С., Королькова А. В., Геворкян М. Н.