

Лабораторная работа №8

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

Панченко Денис Дмитриевич

Содержание

1	Цель работы	3
2	Задачи	4
3	Выполнение лабораторной работы	5
4	Контрольные вопросы	7
5	Вывод	9

1 Цель работы

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

2 Задачи

- Научиться применять режим однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

3 Выполнение лабораторной работы

1) Создадим файл с программой (рис. 3.1).

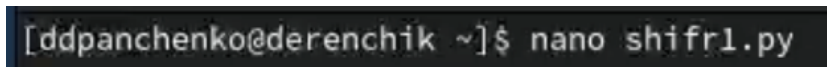


Рис. 3.1: Файл с программой

2) Напишем саму программу для шифрования (рис. 3.2).

```
GNU nano 5.6.1 shifr1.py
def encrypt(text, key):
    text_bytes = bytearray(text, 'utf-8')
    key_bytes = bytearray.fromhex(key)

    encrypted_text = bytearray(len(text_bytes))
    for i in range(len(text_bytes)):
        encrypted_text[i] = text_bytes[i] ^ key_bytes[i % len(key_bytes)]

    return encrypted_text

def decrypt(encrypted_text, key):
    key_bytes = bytearray.fromhex(key)

    decrypted_text = bytearray(len(encrypted_text))
    for i in range(len(encrypted_text)):
        decrypted_text[i] = encrypted_text[i] ^ key_bytes[i % len(key_bytes)]

    return decrypted_text.decode('utf-8')

def analyze_texts(encrypted_text1, encrypted_text2, known_text1):
    decrypted_known_text1 = bytearray(known_text1, 'utf-8')
    analyzed_text2 = bytearray(len(encrypted_text2))
    min_length = min(len(encrypted_text1), len(encrypted_text2), len(decrypted_known_text1))
    for i in range(min_length):
        analyzed_text2[i] = encrypted_text1[i] ^ encrypted_text2[i] ^ decrypted_known_text1[i]

    return analyzed_text2.decode('utf-8')

P1 = "НаВашисходящийот1204"
P2 = "ВСеверныйфилиалБанка"
K = "050C177F0E4E37D29410092E2257FFC80BB27054"

encrypted_text1 = encrypt(P1, K)
encrypted_text2 = encrypt(P2, K)

decrypted_text1 = decrypt(encrypted_text1, K)
decrypted_text2 = decrypt(encrypted_text2, K)

known_text1 = "НаВашисходящийот"
analyzed_text2 = analyze_texts(encrypted_text1, encrypted_text2, known_text1)

print("P1:", decrypted_text1)
print("P2:", decrypted_text2)
print("Analyzed P2:", analyzed_text2)
```

Рис. 3.2: Программа

3) Выполним эту программу (рис. 3.3).

```
[ddpanchenko@derenchik ~]$ python shifr1.py  
P1: НаВашисходящийот1204  
P2: ВСеверныйфилиалБанка  
AnaLyzed P2: ВСеверныйфилиалБ
```

Рис. 3.3: Выполнение программы

4 Контрольные вопросы

1. Для определения другого текста (P2, например), зная один из текстов (P1) без знания ключа, можно использовать аналитический метод, основанный на операции XOR. Если у вас есть шифротексты обоих сообщений, то можно применить операцию XOR между шифротекстами. Это даст вам результат, который, когда применен к P1, даст P2. Таким образом, можно получить P2, не зная ключа.
2. При повторном использовании ключа при шифровании текста в режиме однократного гаммирования тексты будут зашифрованы одним и тем же способом. Это значит, что при повторном использовании ключа для разных открытых текстов может возникнуть возможность провести атаку на шифротекст, используя известные свойства открытых текстов и операции XOR.
3. Режим шифрования однократного гаммирования одним ключом двух открытых текстов осуществляется путем применения операции XOR между каждым байтом открытого текста и соответствующим байтом ключа. То есть, каждый байт открытого текста складывается по модулю 2 (XOR) с соответствующим байтом ключа.
4. Недостатки шифрования одним ключом двух открытых текстов:
 - При повторном использовании ключа может возникнуть уязвимость из-за возможности атаки на шифротекст, основанной на известных открытых текстах и операции XOR.

- Сложность в управлении ключами и обеспечении их безопасного обмена, особенно если требуется использовать разные ключи для разных текстов.

5. Преимущества шифрования одним ключом двух открытых текстов:

- Экономия на вычислительных ресурсах и объеме ключа, так как для шифрования используется один ключ для двух текстов.
- Удобство при передаче и хранении ключа, так как требуется только один ключ для обоих текстов.
- Возможность эффективного использования аналитических методов для расшифровки сообщений при наличии информации об одном из открытых текстов.

5 Вывод

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