Лабораторная работа № 7

Юрченко Артём Алексеевич

2023, Москва

Цели

Освоить на практике применение режима однократного гаммирования.

Задание

Нужно подобрать ключ, чтобы получить сообщение «С Новым Годом, друзья!». Требуется разработать приложение, позволяющее шифровать и дешифровать данные в режиме однократного гаммирования. Приложение должно:

1. Определить 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.

Теоретическое введение

Гаммирование – это наложение (снятие) на открытые (зашифрованные) данные криптографической гаммы, т.е. последовательности элементов данных, вырабатываемых с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных.

Теоретическое введение

Наложение (или снятие) гаммы на блок сообщения в рассматриваемом нами стандарте реализуется с помощью операции побитного сложения по модулю 2 (XOR). То есть при шифровании сообщений каждый блок открытого сообщения ксорится с блоком криптографической гаммы, длина которого должна соответствовать длине блоков открытого сообщения. При этом, если размер блока исходного текста меньше, чем размер блока гаммы, блок гаммы обрезается до размера блока исходного текста (выполняется процедура усечения гаммы).

Выполнение лабораторной работы

Код программы на языке python

```python
import string
import random

def to_hex(text):
    return " ".join(hex(ord(char))[2:] for char in text)

def generate_key(size):
    key = "".join(random.choice(string.ascii_letters + string.digits) for _ in range(size))
    return key

def custom_encoder(text, key):
    return "".join(chr(a ^ b) for a, b in zip(text, key))

message = "С Новым годом, друзья!"
encryption_key = generate_key(len(message))
hex_key = to_hex(encryption_key)
print("Ключ: ", hex_key)

encrypted_text = custom_encoder([ord(char) for char in message], [ord(char) for char in encryption_key])
hex_text = to_hex(encrypted_text)
print("Зашифрованное сообщение: ", hex_text)

decrypted_text = custom_encoder([ord(char) for char in encrypted_text], [ord(char) for char in encryption_key])
print("Расшифрованный текст: ", decrypted_text)
```

Пример работы программы

```
Ключ:  41 63 6c 30 37 43 6a 44 77 4e 34 59 4d 6c 52 77 75 35 48 66 6b 6e
Зашифрованное сообщение:  460 43 471 40e 405 408 456 64 444 470 400 467 471 40 72 443 435 476 47f 42a 424 4f
Расшифрованный текст:  С Новым годом, друзья!
```

Выводы

Результаты выполнения лабораторной работы

Изучили алгоритм шифрования с помощью гаммирования