Лабораторная работа № 8

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

Абу Сувейлим Мухаммед Мунифович

Содержание

# 1 Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом [1].

# 2 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.

# 3 Выполнение лабораторной работы

Во-первых, нужно импортировать нужные библиотеки командой:

import os

Далее, для максимально эффективной работы лучше выполнять команды на уровне битов. Для этого мы напиши функцию xor\_bytes наложения гаммы:

def xor\_bytes(text, key):  
 return bytes([a ^ b for a, b in zip(text, key)])

Далее, напишем две функции для encoding и decoding:

def encrypt(text):  
 text\_bytes = text.encode('utf-8')  
 cipher\_text = xor\_bytes(text\_bytes, key)  
 return cipher\_text  
  
def decrypt(cipher\_text, key):  
 plain\_text\_bytes = xor\_bytes(cipher\_text, key)  
 return plain\_text\_bytes.decode('utf-8')

Выполняем пример из учебника:

# Given texts (P1 and P2)  
P1 = "НаВашисходящийот1204"  
P2 = "ВСеверныйфилиалБанка"  
  
# Display the original texts and the key  
print(f"Text P1: {P1}")  
print(f"Text P2: {P2}")  
  
# Convert texts to bytes  
P1\_bytes = P1.encode('utf-8')  
P2\_bytes = P2.encode('utf-8')

Создадим ключ дленной текста P1:

key = os.urandom(len(P1\_bytes))  
# Encrypt P1 and P2 with the same key  
C1 = encrypt(P1, key)  
C2 = encrypt(P2, key)  
  
# Display the encrypted texts and the key  
print(f"C1 (Cipher text for P1): {C1}")  
print(f"C2 (Cipher text for P2): {C2}")  
print(f"Key: {key}")  
  
# Decrypt the texts back to verify correctness  
P1\_decrypted = decrypt(C1, key)  
P2\_decrypted = decrypt(C2, key)  
print(f"Decrypted P1: {P1\_decrypted}")  
print(f"Decrypted P2: {P2\_decrypted}")  
  
  
P1\_xor\_P2 = xor\_bytes(C1, C2)  
print(f"P1 XOR P2: {P1\_xor\_P2}")  
  
  
P2\_recovered\_bytes = xor\_bytes(P1\_xor\_P2, P1\_bytes)  
P2\_recovered = P2\_recovered\_bytes.decode('utf-8')  
print(f"Recovered P2 using P1 and C1 XOR C2: {P2\_recovered}")

Получаем такой результат:

Text P1: НаВашисходящийот1204  
Text P2: ВСеверныйфилиалБанка  
C1 (Cipher text for P1): b"\x95\x0e4\x99\x8d\xef.\xff\x11\x8dp\x8e\xde\xcc\xcd\\, Yd\r\xec\xdc\x8b'p\x19\xd3\xcbo\xbdB\xe2r\xedI"  
C2 (Cipher text for P2): b"\x95\x014\x88\x8d\xc8.\xfd\x10\xb0q\xb6\xdf\xf0\xcdR,'XT\x0c\xdb\xdd\xb9'p\x19\xda\xcbj\xbcQ\x03\xf0\r\xc0"  
Key: b'E\x93\xe4)]}\xfeO\xc0\x05\xa06\x0fM\x1c\xd9\xfc\x9e\x89\xd0\xdcc\r\x02\xf7\xc8\xc9j\x1b\xd1l\xc0\xd3@\xdd}'  
Decrypted P1: НаВашисходящийот1204  
Decrypted P2: ВСеверныйфилиалБан  
P1 XOR P2: b"\x00\x0f\x00\x11\x00'\x00\x02\x01=\x018\x01<\x00\x0e\x00\x07\x010\x017\x012\x00\x00\x00\t\x00\x05\x01\x13\xe1\x82\xe0\x89"  
Recovered P2 using P1 and C1 XOR C2: ВСеверныйфилиалБан

# 4 Выводы

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

# Список литературы

1. Kulyabov D., Korolkova A., Gevorkyan M. Информационная безопасность компьютерных сетей: лабораторные работы. 2015.