МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»

Інститут комп'ютерних систем

Кафедра інформаційних систем

Протокол

Лабораторної роботи №3

За дисципліною: «Технології захисту інформації»

Тема: «Гамування (абсолютно стійкий шифр)»

Виконали:

Студенти групи АІ-181

Горбік М.М.

Козиревич Я.Р.

Перевірили:

Тьєн Нгуєн

Чан Вінь

Одеса 2021

Лабораторна робота №3

Тема: Гамування (абсолютно стійкий шифр)»

**Мета роботи** – Освоїти на практиці застосування режиму однократного гамування. Дослідити побітно безперервне шифрування даних.

Завдання

1. Ознайомитися з теоретичними відомостями щодо шифрування гаммуванням в рекомендованій літературі і конспектах лекцій.

2. Реалізувати додаток, що дозволяє шифрувати і дешифрувати дані в режимі однократного гамування. Його завдання полягають у наступному:

а) Визначити вид шифротекста при відомому ключі і відомому відкритому тексті.

б) Визначити ключ, за допомогою якого шифротекст може бути перетворений в певну частину тексту, що представляє собою один з можливих варіантів прочитання відкритого тексту.

3. Налагодити розроблену програму і показати результати роботи програми викладачеві.

4. Скласти звіт по лабораторній роботі

Хід роботи

Код програми

import random

from itertools import cycle

MAX\_NUM = 2560

CHARS\_IN\_UTF = 137994

def decrypt(text: str, gamma):

result = ''

for i in range(len(text)):

l = (ord(text[i]) ^ gamma[i]) % CHARS\_IN\_UTF

result += chr(l)

return result

def crypt(text: str, key: str):

result = ''

random.seed(key)

rand = []

lets = []

for x in text:

if not x:

break

r = random.randint(0, MAX\_NUM)

rand.append(r)

lets.append(ord(x))

result += chr((ord(x) ^ r) % CHARS\_IN\_UTF) # всё к-во символов в таблице кодировки

return result

def get\_letters\_codes(text):

result = []

for c in text:

result.append(ord(c))

return result

key = 'whatever text is here is a key 7128%\*$^#'

phrase = 'Текстовая фраза'

print("Исходный текст: '" + phrase + "'")

u = crypt(phrase, key)

print("Зашифрованый текст: '" + str(u) + "'")

print("Расшифрованый текст: '" + str(crypt(u, key)) + "'")

a = get\_letters\_codes(phrase) # коды чистого текста

b = get\_letters\_codes(crypt('Чеснок это овощ', key)) # коды зашифрованого текста (желаемый левый вариант)

c = []

for (x, y) in zip(a, cycle(b)):

c.append(x ^ y) # операция xor этих кодов даст гамму, которой нужно гамировать зашифрованный чистый текст,

# чтобы получить при расшифровке указанный желаемый левый

s = ''

for i in c:

s += "%d " % i

print('Пример гаммы для некорректной расшифровки:\n' + s[:-2])

s = decrypt(crypt(phrase, key), c)

print("Фраза, расшифрованная этой гаммой:'" + s + "'")

**Висновок**

В ході виконання лабораторної роботи була створена програма, що дозволяє шифрувати і дешифрувати дані в режимі однократного гамування. Також програма визначає вид шифротекста при відомому ключі і відомому відкритому тексті, та визначає гамму, за допомогою якої шифротекст може бути перетворений в певну частину тексту, що представляє собою один з можливих варіантів прочитання відкритого тексту.