Пшенко Артем ФИТ 3-4

Отчет по лабораторной работе № 7

Исследование блочных шифров (алгоритм DES)

Информационная безопасность

Вариант 9

Цель данной лабораторной работы состоит в изучении работы алгоритма блочного шифрования DES, его применении для шифрования и дешифрования сообщений. Необходимо продемонстрировать, как шифрование и последующее дешифрование исходного сообщения возвращает его в первоначальное состояние.

Алгоритм DES включает в себя следующие шаги:

1. Разделение сообщения: сообщение разбивается на блоки по 64 бита.
2. Начальная перестановка: каждый блок подвергается начальной перестановке согласно фиксированной таблице.
3. 16 раундов шифрования: каждый блок проходит через 16 раундов шифрования, где используются различные подблоки ключа:
4. Разделение блока: блок делится на две половины по 32 бита.
5. Функция F: правая половина блока проходит через функцию F, включающую расширение, замену и перестановку.
6. XOR с левой половиной: результат функции F объединяется с левой половиной блока с помощью операции XOR.
7. Обмен половин: половины блока меняются местами.
8. Конечная перестановка: после 16 раундов половины блока объединяются и подвергаются конечной перестановке, обратной начальной.
9. Объединение блоков: все зашифрованные блоки объединяются в окончательное зашифрованное сообщение.

Листинг 1. Исходный код алгоритма DES

|  |
| --- |
| from Crypto.Cipher import DES  from Crypto.Util.Padding import pad, unpad  from time import time  key = "Artyomps".encode()  def des\_encrypt\_decrypt(message, key, mode='encrypt'):  cipher = DES.new(key, DES.MODE\_ECB)  if mode == 'encrypt':  start\_time = time()  ct\_bytes = cipher.encrypt(pad(message.encode(), DES.block\_size))  end\_time = time()  print(f"Encryption time: {end\_time - start\_time} seconds")  return ct\_bytes  elif mode == 'decrypt':  start\_time = time()  pt = unpad(cipher.decrypt(message), DES.block\_size).decode()  end\_time = time()  print(f"Decryption time: {end\_time - start\_time} seconds")  return pt  message = "Pshenko Artyom Fyodorovich"  cipher\_text = des\_encrypt\_decrypt(message, key, 'encrypt')  print("Cipher text:", cipher\_text)  plain\_text = des\_encrypt\_decrypt(cipher\_text, key, 'decrypt')  print("Plain text:", plain\_text) |

Результат работы программы приведен на рисунке 1.

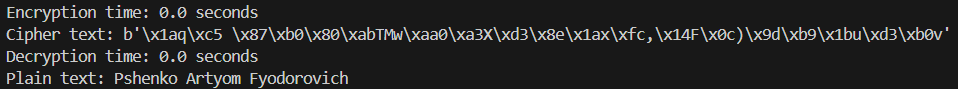


Рисунок 1. Результат работы программы

В ходе лабораторной работы была продемонстрирована работа алгоритма блочного шифрования DES. Шифрование исходного сообщения "Pshenko Artyom Fyodorovich" с последующим его дешифрованием показало, что при использовании правильного ключа и режима шифрования исходное сообщение успешно восстанавливается. Это подтверждает корректность работы алгоритма DES и демонстрирует его эффективность для шифрования и защиты информации.