Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ОТЧЕТ

О ЛАБОРАТОРНОЙ РАБОТЕ

Лабораторная работа №3
по теме:
«Применение блочных шифров»

Исполнитель, студент группы 201-361
_____ Н.А. Фельдбуш

Москва, 2023

Постановка задачи:

Напишите программу, шифрующую изображение tux.png (формат не принципиален) с помощью шифра AES. Режимы шифрования: ECB, CBC, CFB и OFB (нужно получить четыре варианта зашифрованного изображения). В учебных целях заголовочную часть файла зашифровывать не нужно. Сравните скорости выполнения алгоритмов и результаты шифрования.

Ход работы:

Выполним поставленную задачу при помощи средств языка программирования Python. Для реализации используются две основных функции (рис.1).

```
rel_path = "tux.png"


def main():
    subprocess.run('"C:/Program Files/Git/usr/bin/openssl.exe" rand -hex 16 > key.bin',
                   shell=True, stdout=subprocess.PIPE)
    start=time.time()
    subprocess.run(["C:/Program Files/Git/usr/bin/openssl.exe", "enc",
                    "-aes-256-ecb", "-in", rel_path, "-out", "new_ecb.txt","-pbkdf2", "-pass", "file:key.bin"], stdout=subprocess.PIPE)
    finish=time.time()
    print("ECB: "+str(finish-start) + "secs. Size: " + str(os.path.getsize("new_ecb.txt")) + " bytes.")
    start = time.time()
    subprocess.run(["C:/Program Files/Git/usr/bin/openssl.exe", "enc",
                    "-aes-256-cbc", "-in", rel_path, "-out", "new_cbc.txt","-pbkdf2", "-pass", "file:key.bin"], stdout=subprocess.PIPE)
    finish = time.time()
    print("CBC: "+str(finish-start) + "secs. Size: " + str(os.path.getsize("new_cbc.txt")) + " bytes.")
    start = time.time()
    subprocess.run(["C:/Program Files/Git/usr/bin/openssl.exe", "enc",
                    "-aes-256-cfb", "-in", rel_path, "-out", "new_cfb.txt", "-pbkdf2", "-pass", "file:key.bin"], stdout=subprocess.PIPE)
    finish= time.time()
    print("CFB: "+str(finish-start) + "secs. Size: " + str(os.path.getsize("new_cfb.txt")) + " bytes.")
    start=time.time()
    subprocess.run(["C:/Program Files/Git/usr/bin/openssl.exe", "enc",
                    "-aes-256-ofb", "-in", rel_path, "-out", "new_ofb.txt", "-pbkdf2", "-pass", "file:key.bin"], stdout=subprocess.PIPE)
    finish= time.time()
    print("OFB: "+str(finish-start) + "secs. Size: " + str(os.path.getsize("new_ofb.txt")) + " bytes.")


def decode(path, type):
    type = str.lower(type)
    subprocess.run(["C:/Program Files/Git/usr/bin/openssl.exe", "enc", "-d",
                    f"-aes-256-{type}", "-in", path, "-out", f"new_{type}.png", "-pbkdf2", "-pass", "file:key.bin"], stdout=subprocess.PIPE)
```

Рисунок 1 – Основные функции

Функция “main” выполняет основную поставленную задачу. С помощью библиотеки “subprocess” и “openssl.exe” происходит шифрование файла четырьмя разными способами: “aes-256-ecb”, “aes-256-cbc”, “ aes-256-cfb”, “ aes-256-ofb”. Так же в функции измеряется время необходимое для шифрования, а также размер зашифрованных файлов.

Функция “decode” принимает в себя путь к файлу, а также режим шифрования с помощью, которого файл был зашифрован. С помощью этой информации файл дешифруется и создается новый расшифрованный файл.