**ИНТЕЛЛЕКТУАЛЬНЫЕ МЕТОДЫ В ЗАДАЧАХ ЗАЩИТЫ ОТ КИБЕРУГРОЗ**

**ЛАБОРАТОРНАЯ РАБОТА № 3**
**DLP-системы.**

**Выполнил:**
**Мосолков Е.Н.**
**Преподаватель:**
**Исхаков А.Ю.**

Москва 2021 г.

## ЦЕЛЬ РАБОТЫ

Изучение механизмов работы систем класса DLP

**ЗАДАЧА ПРАКТИЧЕСКОЙ РАБОТЫ**

Разработать реализацию простейшего прототипа DLP-системы, включающего не менее 1 функции для выявления утечки данных. Требований к платформам, средам, языкам программирования не предъявляется. Примеры (приветствуются также любые варианты от студентов):

- поиск конфиденциального содержимого в файлах по заранее созданным шаблонам;
- выявление утечки на основании анализа сетевого интерфейса;
- детектирование подключения несанкционированных носителей информации / устройств вывода;
- расширение для браузера, анализирующее контент посещаемых страниц.

Записать краткое видео (не более 3 минут), демонстрирующее работу системы и приложить ссылку на данное видео.

**ОПИСАНИЕ ФУНКЦИОНАЛА**

В рамках практической работы была сделана система DLP, которая узнает и оповещает о нежелательных USB носителях. Система написана на языке Python с использованием технологии socket и библиотеки os

Система состоит из 2 составляющих:

1. Программа шпион (dlp_security_system.py)
2. Сервер (server.py)

Взаимодействие программы с сервером основано на socket соединении.

Основной алгоритм работы – следующий:

1. Запускается сервер
2. Запускается приложение, которое отслеживает подключенные USB устройства
3. При подключении/отключении устройства приложение отсылает на сервер соответствующую информацию и оповещает пользователя о подключении нежелательного устройства.

Подразумевается, что пользователи сервера не хотят, чтобы пользователь приложения каким-либо образом распространял информацию, поэтому заранее узнают было ли подключено/отключено USB устройство, что может поспособствовать устранению утечки данных во вне. Это пассивный хостовый тип DLP системы – специалист по информационной безопасности имеет полную картину, что и как происходит с носителями памяти в системе пользователя программы

**КОД**

Код файла server.py:

```
import socket

server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server.bind(('127.0.0.1', 5555))
server.listen(5)
print('Server running . . .')

while True:
    conn, addr = server.accept()
    conn.send('Connected successfully!'.encode('utf8'))
    print(f'Connection {addr} successful!')
    while True:
        print(f'Message: {conn.recv(2048).decode("utf8")}')
```

Код файла dlp_security_system.py:

```
import os.path
import socket


def diff(list1, list2):
    list_difference = [item for item in list1 if item not in list2]
    return list_difference


client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect(('127.0.0.1', 5555))
data = client.recv(2048)
print(data.decode('utf8'))

drive_names = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
drives = [f'{d}:' for d in drive_names if os.path.exists(f'{d}:')]
print(drives)
while True:
    unchecked_drives = [f'{d}:' for d in drive_names if
os.path.exists(f'{d}:')]
    x = diff(unchecked_drives, drives)
    if x:
        print(f'Warning!!! Unknown device: {x}')
        client.send(f'Unknown device detected: {x}'.encode('utf8'))
    x = diff(drives, unchecked_drives)
    if x:
        print(f'Removed drives: {x}')
        client.send(f'Device disconnected: {x}'.encode('utf8'))
    drives = [f'{d}:' for d in drive_names if os.path.exists(f'{d}:')]

%s:' % d)]
```

## СКРИНШОТЫ И ССЫЛКА НА ВИДЕО

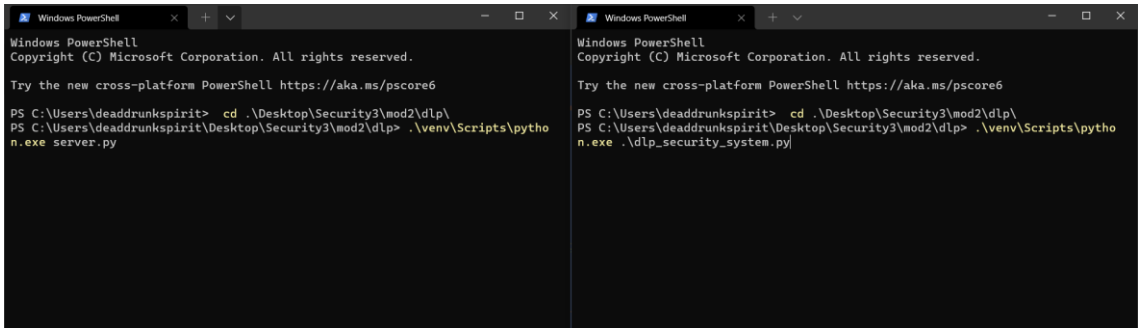


Рисунок 1. Начало работы системы

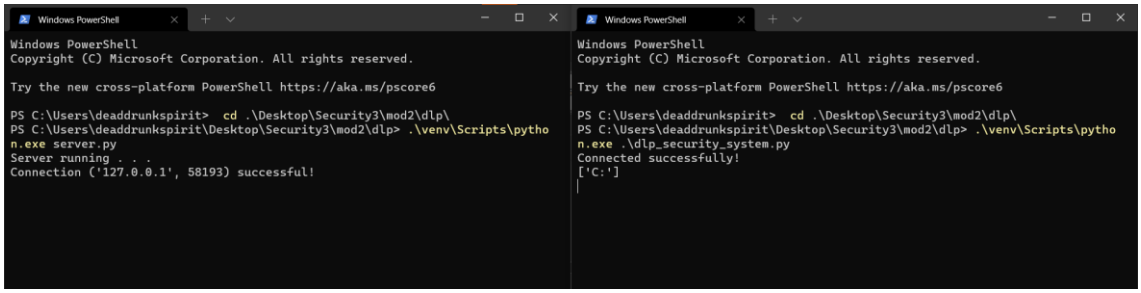


Рисунок 2. Система работает

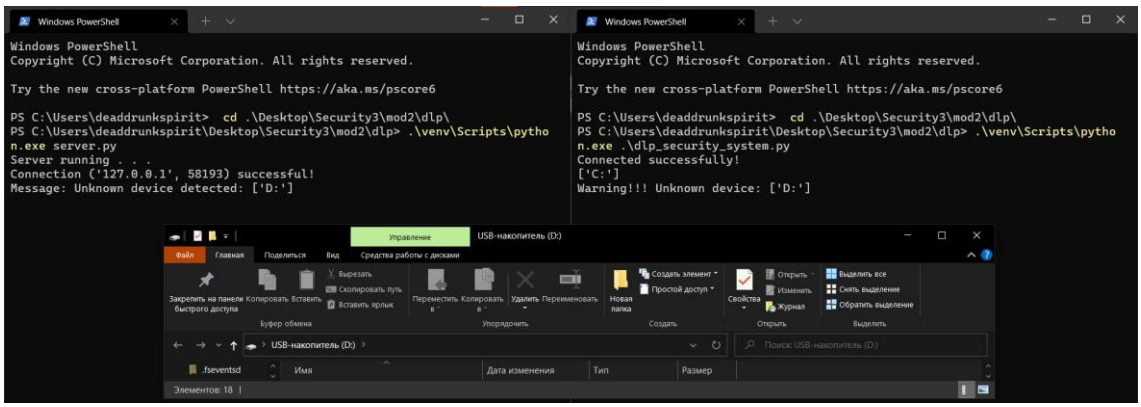


Рисунок 3. Подключили USB флэшку

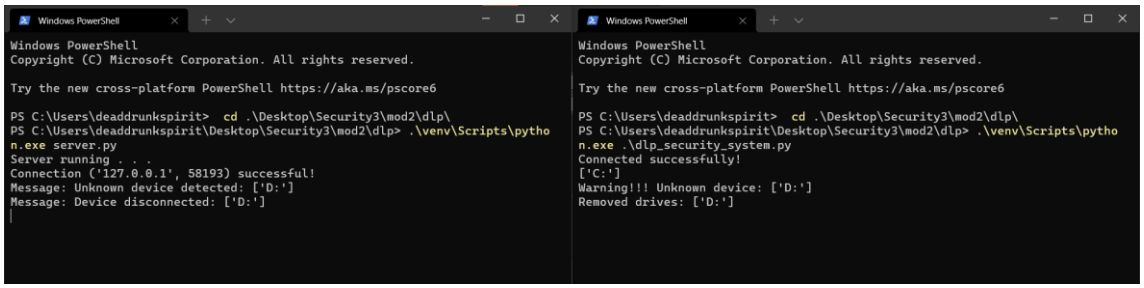


Рисунок 4. Отключили USB флэшку

Ссылка на видео:
https://drive.google.com/file/d/1kTG7bl939wZiDENUFWCYvMtihYs2qB97/view?usp=sharing

**ВЫВОД**

Я изучил механизмы работы систем класса DLP