# цель работы

Исследование принципов разработки моделей безопасности, функционирующих на уровне ядра ОС Windows.

# ход работы

Для создания нужного проекта (рисунок 1) были установлены определённые версии SDK и WDK, а именно версия 2004 для Windows 10

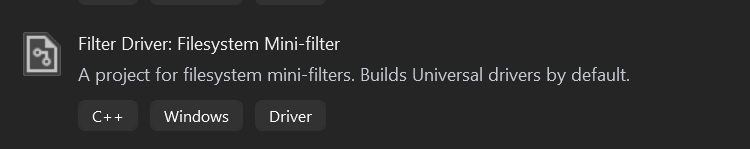


Рисунок 1 – Проект драйвер фильтра

Данный фильтр осуществлял только загрузку и выгрузку и не выполнял никакой полезной нагрузки. Для установки данного драйвера была использована виртуальная машина VMware с подготовленной Windows 10, а именно в консоли были прописаны следующие команды:

bcdedit -set TESTSIGNING ON

bcdedit /set {globalsettings} advancedoptions true

bcdedit /debug on

После этих команд операционная система запускалась в тестовом режиме. Далее был активирован дополнительный параметр загрузки “Отключение обязательной проверки подписи драйвера” (рисунок 2).

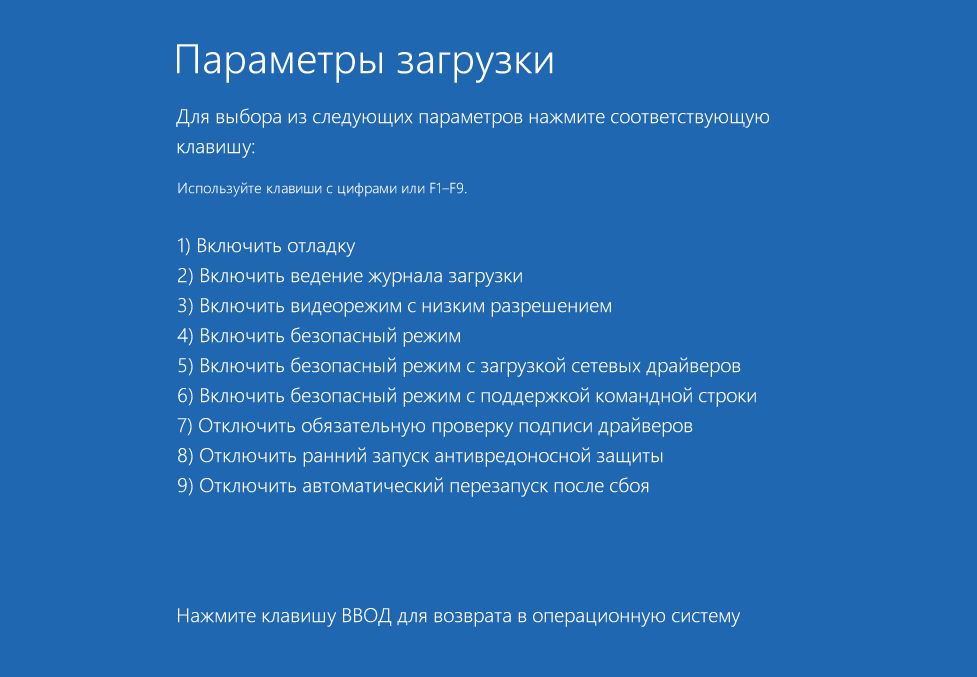


Рисунок 2 – Отключение подписи драйвера

Для установки драйвера использовалась программа “OSR Driver loader” со следующими параметрами (рисунок 3)

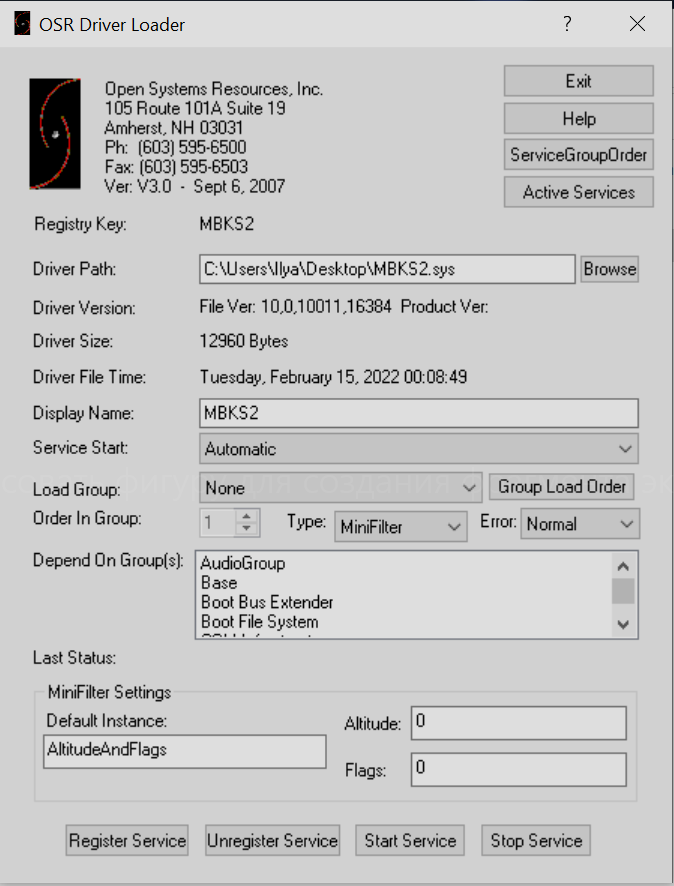


Рисунок 3 – Параметры загрузчика драйвера

Для получения отладочного сообщения была использована программа “DebugView”.

Разработанный драйвер осуществляет ограничить права доступа процессов к объектам файловой системы по ролевой модели доступности. В данной системе процессы могут читать и изменять файлы с таким же или более низким приоритетом, и только читать файлы с более высоким приоритетом. Роли для доступа к файлам и роли самих процессов прописываются в файле config.txt, который находится по пути “C:\Windows” (рисунок 4)

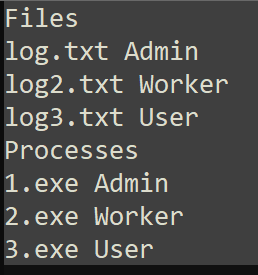


Рисунок 4 – Пример конфиг файла

Разработанный драйвер работает по следующему алгоритму (рисунок 5)

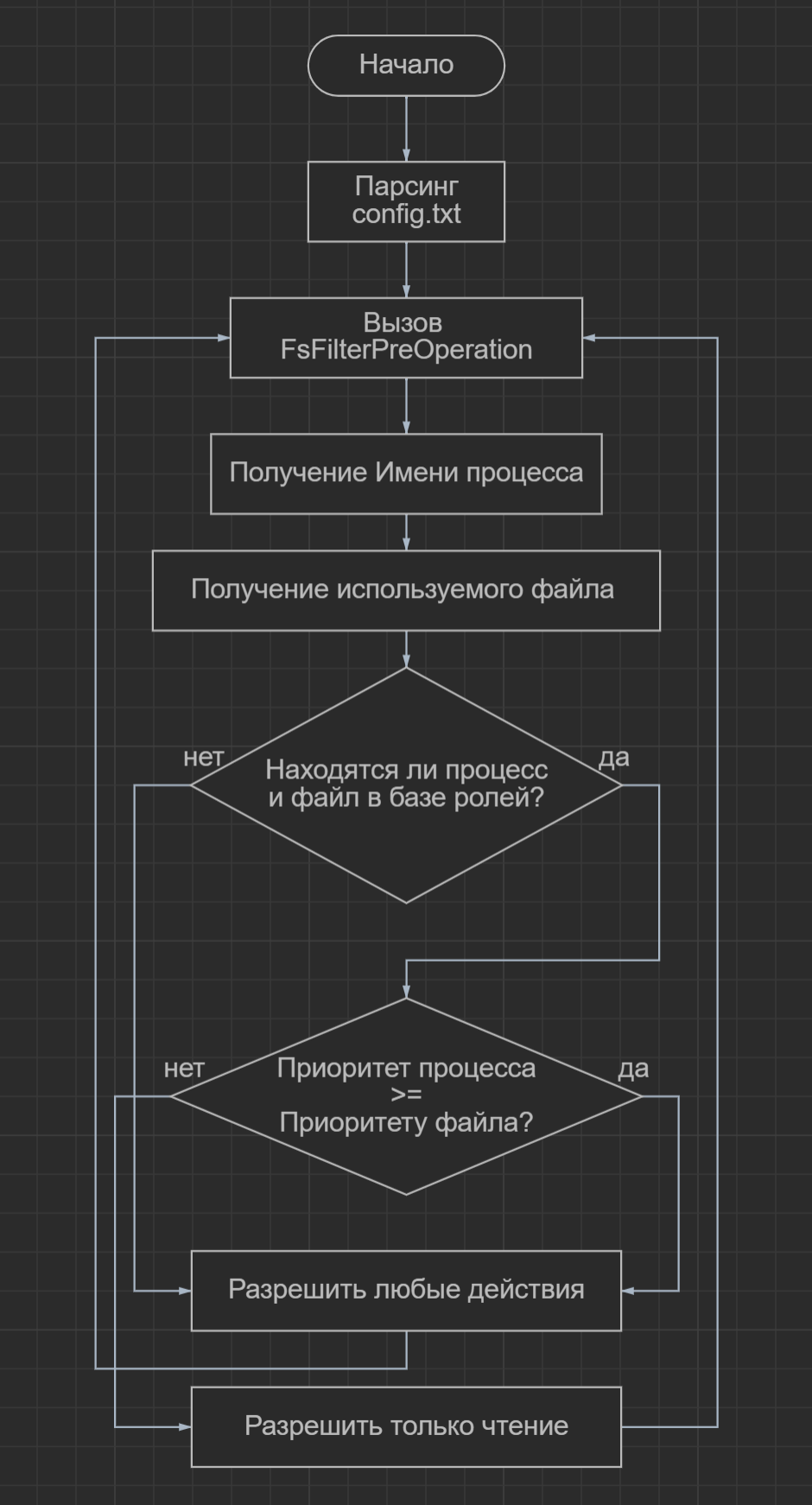


Рисунок 5 – Блок схема алгоритма работы драйвера

Для тестирования, разработанного драйверы была реализована программа, записывающая “Test Test Test”, во входной файл, а затем выводящая содержимое файла в консоль. Пример работы драйвера (рисунки 6-8)

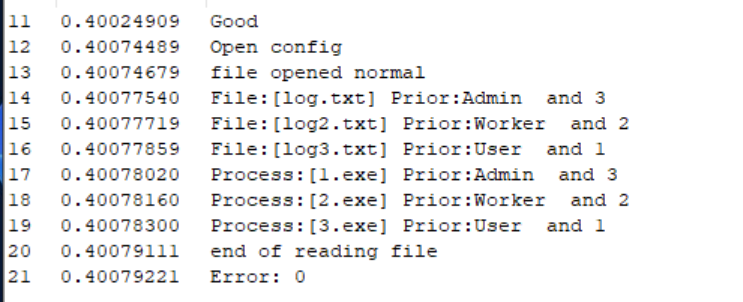


Рисунок 6 – Инициализация драйвера

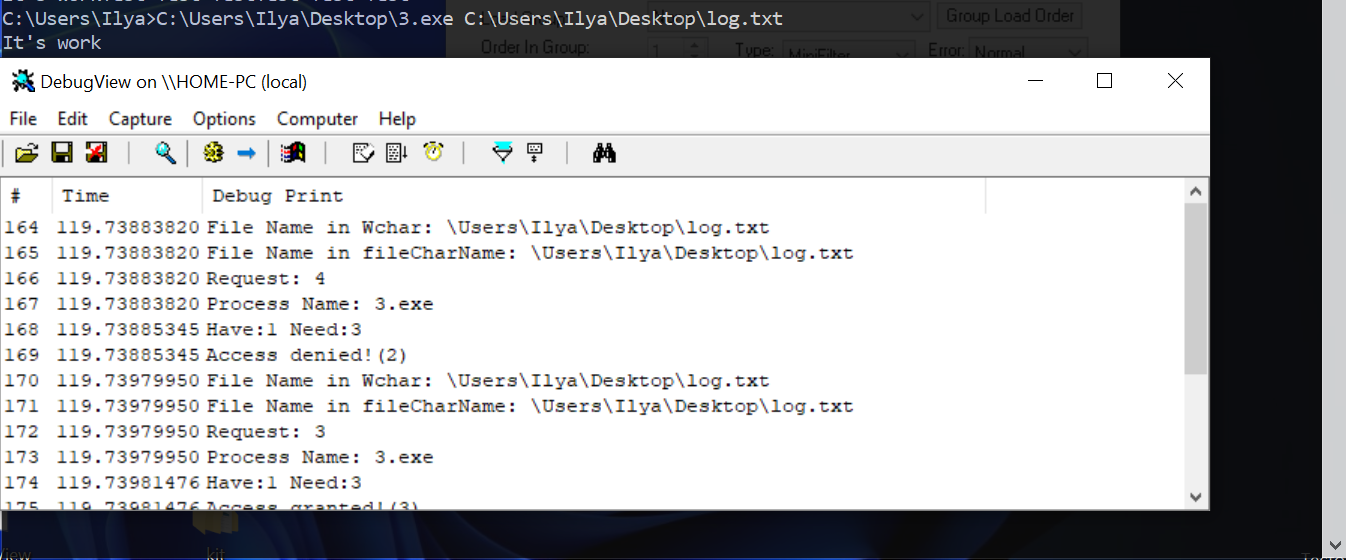


Рисунок 7 – Пример отказа доступа

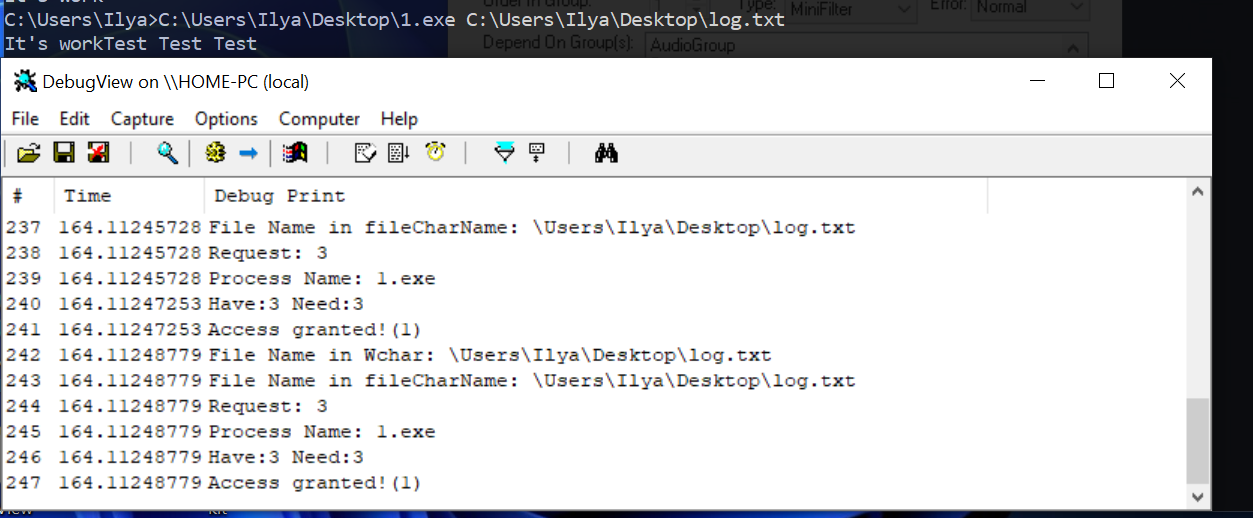


Рисунок 8 – Пример разрешения в изменении

# Изменение config файла

Для данной лабораторной работы было следующее дополнительное задание: “Сделать возможность изменения имени конфиг файла в процессе работы драйвера (без перезапуска драйвера).”

Для реализации этого задание помимо операций записи и чтения драйвер так же отлавливал операцию изменения имени файла (Рисунки 9-11)

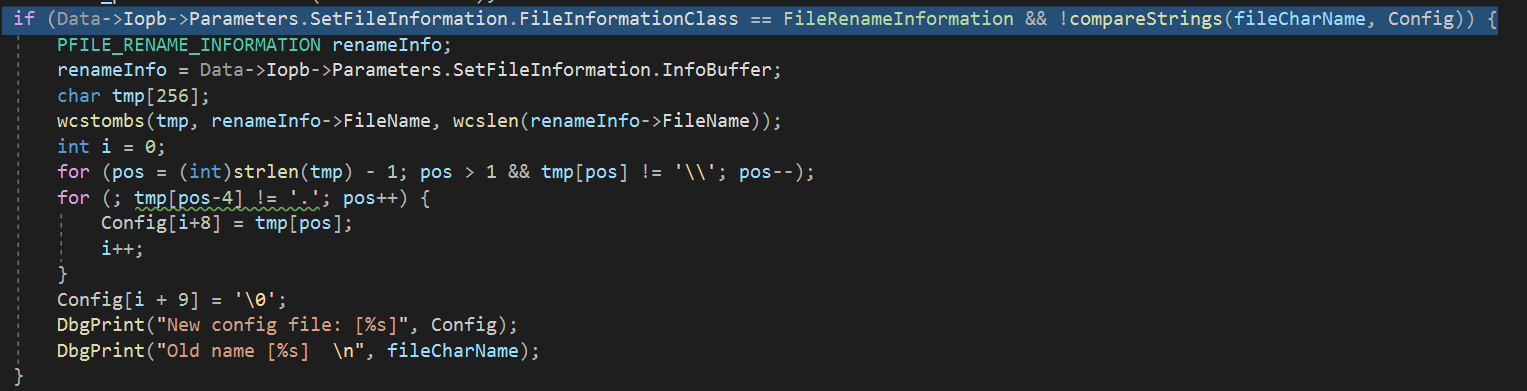


Рисунок 9 – Реализация отлавливания операции изменения имени

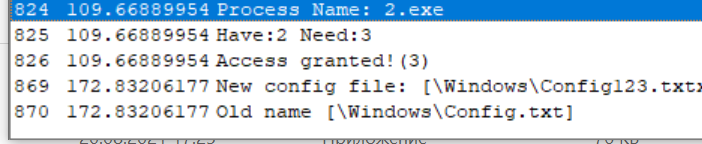


Рисунок 10 – Изменение имени конфиг файла

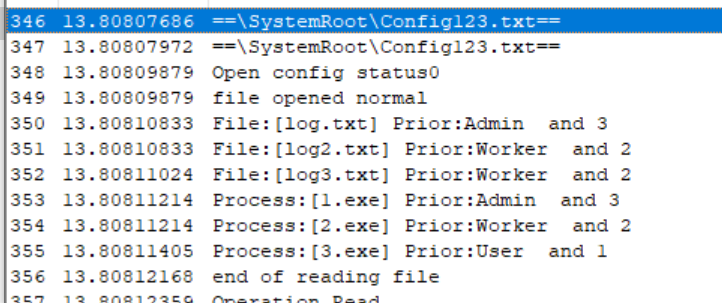


Рисунок 11 – Парсинг нового конфиг файла