Driver

목차

[프로젝트 구성 2](#_Toc91785131)

[MetisD 2](#_Toc91785132)

[MetisS 2](#_Toc91785133)

[MetisD 3](#_Toc91785134)

[주요 파일 3](#_Toc91785135)

[Init.c 3](#_Toc91785136)

[Operation.c 3](#_Toc91785137)

[UserSync.c 3](#_Toc91785138)

[행위 추적 및 기록 4](#_Toc91785139)

[기록 행위 4](#_Toc91785140)

[기록 정보 4](#_Toc91785141)

[파일 컨텍스트 4](#_Toc91785142)

[파일 매직 넘버 변경 5](#_Toc91785143)

[주요 기능 6](#_Toc91785144)

[MetisS 7](#_Toc91785145)

[주요 기능 7](#_Toc91785146)

[주요 동작 7](#_Toc91785147)

[드라이버 로그 회수 7](#_Toc91785148)

[드라이버 메시지 수신 7](#_Toc91785149)

# 솔루션 구성

## MetisD

클라이언트의 설정에 따라 센서 폴더 접근 알림, 폴더 접근 제어, 시스템 로그 기록 등을 수행하는 미니 필터 드라이버 프로그램입니다.

## MetisS

미니 필터 드라이버( MetisD.sys )를 설치/삭제/로드/언로드하고, 드라이버와 통신하여 정책 설정, 기능 제어 등의 작업이 가능한 프로그램입니다.

# MetisD

미니 필터 드라이버에 대한 기본적인 지식은 설명하지 않습니다. 아래 공식 문서를 참조하십시오.

공식 문서: <https://docs.microsoft.com/ko-kr/windows-hardware/drivers/ifs/filter-manager-concepts>

## 주요 파일

### Init.c

필터 매니저에 드라이버를 등록하거나, 콜백이 필요한 IRP 목록을 등록하는 등 전반적인 드라이버 관리가 수행되는 함수들이 있습니다. 클라이언트( MetisS )와 통신하기 위한 초기화 함수도 이 파일에서 호출됩니다.

### Operation.c

Init.c에서 특정 IRP에 대한 콜백, 혹은 프로세스 실행 콜백 등으로 등록한 함수들이 있습니다. 콜백 호출 시, 데이터를 검사하여 작업을 차단하거나, 필요 시 로깅 메모리에 기록하는 작업 등을 수행합니다.

### UserSync.c

클라이언트와 통신하여 정책을 관리하거나 기능을 제어하며, 클라이언트가 설정한 정책으로 IRP를 검사하는 기능 등을 제공합니다.

또한, 클라이언트 요청 시 로깅 메모리에 기록된 데이터를 전달하는 기능 등을 수행합니다.

**-GetInspectStatus 함수**

콜백 데이터를 검사하여 차단 필요 여부를 반환하는 함수로, PreCreate, PreSetInfo 콜백 함수에서 해당 IRP로 인해 파일이 수정될 가능성이 있다고 판단한 경우 호출합니다. 작업 종류는 백업 필요 여부를 확인하거나, 센서 알림 메시지를 전송할 때 함께 전달됩니다.

검사하는 목록은 아래와 같습니다.

* 차단 폴더
* 수정 차단 폴더
* 백업 폴더
* 센서 폴더
* 보호 폴더

위 목록은 수정이 발생하면 안 되는 폴더들이므로 감지된 경우 차단이 필요하다고 판단합니다.

차단되지 않는 작업의 경우 백업 확장자 등을 확인하여 필요한 경우 파일을 백업합니다.

**-ProcessFiltering 함수**

GetInspectStatus 내에서 사용되는 함수로, 검사 결과에 대한 유효성을 확인합니다.

예를 들어, 백업 대상인 경우 신뢰 프로세스의 작업이거나 백업 제외 경로인지 확인하며, 보호 폴더에 대한 작업인 경우 신뢰 프로세스, 혹은 허용 프로세스 여부를 확인합니다.

## 행위 추적 및 기록

드라이버는 파일 시스템에 발생하는 행위를 추적하기 위하여 파일 컨텍스트를 이용하며, 특정 행위가 발생하는 경우 로깅 메모리에 기록합니다. 클라이언트는 이러한 로그를 회수 및 분석하여 랜섬웨어 의심 행위를 탐지합니다.

### 기록 행위

* 파일의 데이터를 읽고, 파일을 삭제  
  : 랜섬웨어 프로세스가 암호화를 위하여 파일의 데이터를 읽고, 원본 파일을 삭제하는 행위를 추적하기 위함.
* 파일에 데이터를 쓰는데, 파일의 매직 넘버가 변경됨  
  : 랜섬웨어 프로세스가 암호화된 데이터를 저장하여 파일 매직 넘버가 변경되는 상황을 추적하기 위함.

### 기록 정보

* 행위 종류
* 프로세스 ID
* 파일 경로

### 파일 컨텍스트

* CreateProcessId: 파일을 생성한 프로세스 ID입니다.
* LastReadProcessId: 파일을 가장 최근에 읽은 프로세스 ID입니다.
* LastWriteProcessId: 가장 최근에 파일 데이터를 손상시킨 프로세스 ID입니다. 중복 검사 방지를 위해 사용됩니다.
* Flags: 확장자 변경, 검사 완료 등의 플래그를 설정합니다.

행위 추적

**-PostCreate**

* 파일이 생성된 경우: 파일을 생성한 프로세스 ID를 CreateProcessId에 저장합니다.
* 파일이 삭제된 경우: 프로세스가 생성한 파일이 아니고, 파일을 가장 최근에 읽었으면 의심 행위로 기록합니다.

**-PreRead, PreAcquireForSectionSynchronization**

* LastReadProcessId에 프로세스 ID를 저장합니다.

**-PreSetInformation**

* 파일이 삭제된 경우: 프로세스가 생성한 파일이 아니고, 파일을 가장 최근에 읽었으면 의심 행위로 기록합니다.

**-PostSetInformation**

* 확장자가 변경된 경우: 변경 전 확장자가 백업 대상 확장자였으면 Flags에 설정합니다.

**-PreWrite**

* 파일이 손상되는 경우: 데이터를 쓰는 위치가 일반적이지 않거나( ByteOffset > 0 ), 파일 매직 넘버가 변경되면 의심 행위로 기록합니다.

### 파일 매직 넘버 변경

알려진 확장자( ppt, pdf 등 )들은 파일 매직 넘버가 특정한 값으로 고정되어 있으며, 이 값이 바뀌는 경우 충분히 파일의 손상을 의심해 볼 수 있습니다.

다만, 랜섬웨어 프로세스가 파일을 삭제한 후 동일한 파일명으로 암호화된 데이터를 저장하는 경우 파일 매직 넘버를 정상적으로 비교할 수 없습니다.

1. Sample.pdf 데이터를 읽은 후 삭제

2. Sample.pdf 파일 생성 및 암호화된 데이터 저장

위 순서로 작업하는 경우 기존 Sample.pdf 파일의 매직 넘버를 확인하고 비교할 수 없습니다.

때문에, 드라이버는 위와 같은 몇몇 상황에서 백업 기능을 이용하여 백업되어 있는 파일과 비교를 수행하게 되며, 이러한 이유로 파일 매직 넘버 변경을 확인하는 작업은 백업 확장자를 대상으로만 진행합니다.

## 주요 기능

제공하는 기능 및 정책은 아래와 같으며, 각 정책 등록 형식과 기능 적용 방법은 C++ RansomEye 프로젝트의 Driver/UserMessage.h 헤더에 있습니다.

* 신뢰 프로세스
* 보호 프로세스
* 허용 프로세스
* 읽기 허용 프로세스
* 백업 확장자
* 백업 제외 경로
* 백업 제외 프로세스
* 프로세스 로깅 제외 경로
* 행위 로깅 제외 경로
* 파일 로깅 제외 경로
* 파일 로깅 제외 확장자
* 센서 폴더
* 차단 폴더
* 보호 폴더
* 수정 보호 폴더
* 숨김 폴더
* 보호 공유 경로
* 보호 레지스트리
* 보호 파일

# MetisS

C# RansomEye 프로그램이 드라이버를 사용하기 위한 기능을 제공하는 DLL 프로그램입니다.

## 주요 기능

MetisS를 사용하는 프로그램과 MetisD.sys는 동일한 폴더 내에 있어야 하며, 제공하는 주요 기능은 아래와 같습니다.

* 드라이버 설치
* 드라이버 삭제
* 드라이버 로드
* 드라이버 언로드
* 드라이버 기능 설정
* 드라이버 정책 설정
* 드라이버 로그 회수
* 드라이버 메시지 수신

## 주요 동작

### 드라이버 로그 회수

MetisD가 로깅 메모리에 기록한 데이터들을 클라이언트 영역으로 회수하는 기능입니다.

회수된 로그는 종류 별로 구분하여 DB에 저장하거나, 등록된 콜백 함수를 호출하여 처리합니다.

예를 들어, 파일 생성/이동 로그는 DB에 저장하고, 행위 관련 로그는 C# RansomEye 프로그램이 등록한 콜백 함수를 호출합니다.

DB에 저장하는 작업을 MetisS에서 수행하므로, 파일 및 프로세스 관련 DB 스키마 변경 시 MetisS의 작업에 영향이 없는지 확인해야 합니다.

### 드라이버 메시지 수신

드라이버로부터 센서, 보호 폴더, MBR 수정에 관한 이벤트 메시지를 수신하면 C# RansomEye가 등록한 콜백 함수를 호출합니다.