디바이스 드라이버를 활용한 랩톱 보안 소프트웨어

최지원*, 곽현석*, 박종근*
*수원대학교 정보보호학과
jwchoi1066@naver.com, ghsghs97@naver.com, dldvk9999@naver.com

Laptop Security Software using Windows Device Driver

Ji-Won Choi*, Kwak-Hyeon Seok*, Jong-Geun Park*
*Dept. of Information Security, University of Suwon

요 약

오늘 날 대부분의 대학생들은 물론 직장인들도 노트북을 많이 사용한다. 특히 카페에서 노트북을 사용해 과제를 하는 등 열린 공간에서 노트북을 사용하는 일도 많아지고 있다. 그만큼 노트북도 많은 위협에 노출되고 있다. 사용자가 잠시 자리를 비운 틈에 누군가가 노트북 화면을 몰래 볼 수도 있고, 불건전한 의도로 조작할 수도 있다. 본 논문에서는 이러한 위협에 대응하기 위한 방법을 제안하고자 한다.

1. 서론

최근 각 가정마다 노트북 및 태블릿의 보유율이 증가함에 따라 공개적인 장소에서 사용하는 사람들이 늘어나고 있다. 이에 따라 사용자가 자리에 있지 않은 경우 노트북에 대한 위협 또한 증가한다. 즉, 사용자가 노트북을 사용하다 자리를 비웠을 때 이 사이 노트북에 위협을 최소화해줄 프로그램은 필수적이다. 따라서 노트북의 내장 캠이 존재한다면 추가적인 장비를 사용하지 않고 얼굴인식 및 드라이버의 사용을 통해 외부 위협으로부터 노트북을 안전하게 보호할 수 있다[1].

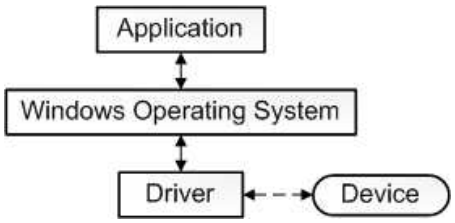
<그림1> 가구 미디어기기 보유율 변화

2. Windows Device Driver

2.1 디바이스 드라이버

드라이버는 운영 체제와 외부 장치가 서로 통신할 수 있게 해주는 소프트웨어 구성 요소다[2].

<그림 2> What is Driver?



운영체제는 디바이스 드라이버를 통해 외부 장치에 데이터를 쓰거나, 읽어 들여 응용 프로그램에게 전달한다. 윈도우에서는 Plug and Play 기능을 통해 별도의 드라이버 설치 없이도 외부 디바이스를 사용할 수 있다. 하지만 제조사들은 자신들이 개발한 드라이버를 배포해 사용자들이 기기의 확장 기능을 사용할 수 있도록 지원한다.

2.2 입출력 관리자(I/O Manager)

윈도우의 커널은 커널 모드 입출력 관리자를 통해 디바이스 드라이버가 제공하는 인터페이스와 응용 프로그램 간의 통신을 관리한다. 응용 프로그램

이 디바이스에 접근할 때 커널 모드 입출력 관리자는 접근 요청을 처리하기위해 먼저 Input/Output Request Packet(IRP)를 생성한다. IRP를 디바이스 드라이버에게 전달하면 드라이버는 IRP에 기록된 요청 내용을 보고 기기에 알맞게 처리한 후 이를 다시 커널 모드 입출력 관리자에게 반환한다. 하나의 디바이스와 통신하기 위해서는 여러 디바이스를 거쳐야 한다. USB로 연결된 마우스라 하더라도 USB 허브, USB 호스트 컨트롤러, PCI 버스 등 여러 하드웨어와 통신해야 한다. 이렇게 서로 연관된 디바이스 드라이버를 묶어 디바이스 스택을 구성한다. 커널 모드 입출력 관리자는 이 스택을 따라 IRP를 전달하며 모든 드라이버가 IRP를 처리하고 나면 그때 버스를 통해 디바이스로 데이터를 전달한다.

2.3 Input/Output Request Packet

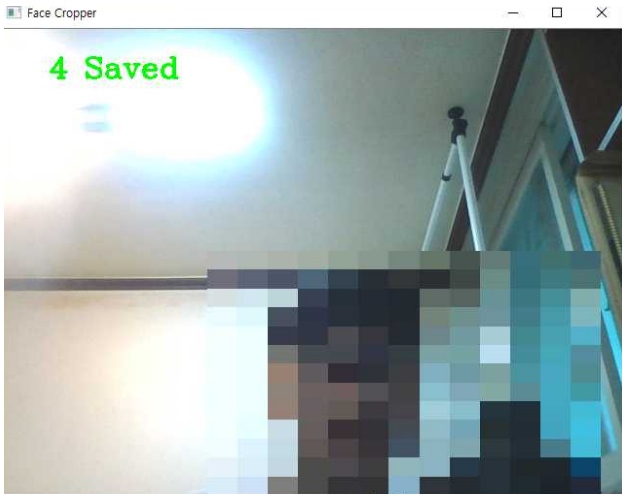
Input/Output Request Packet(IRP)는 응용 프로그램이나 운영체제가 디바이스에 접근하기 위해 드라이버로 전송하는 요청 내용을 담은 구조체다. 커널 모드 입출력 관리자는 디바이스로의 요청이 들어오면 IRP 구조체를 만들어 요청 내용에 대한 값을 넣은 후 IRP 구조체의 포인터를 디바이스 스택을 이루고 있는 드라이버에게 전달한다. IRP에는 해당 드라이버에 어떤 요청이 들어왔는지 나타내는 IRP Major Function Code라는 것이 있다. IPR_MJ라는 접두사로 시작하며, Create, Read, Write 등 주요 기능을 나타낸다.

3. 제안 시스템

3.1 얼굴인식

얼굴인식 모듈은 Face Recognition 모듈과 카메라에서 사용자의 얼굴 데이터를 받아오는 기능으로 구성되어, 카메라에서 사용자의 정보를 받아 저장한 후, 이 데이터를 이용해 Face Recognition 모듈이 사용자가 맞는지 판별하여 코어에 데이터를 넘겨 코어가 이 데이터를 기반으로 통제 여부를 결정한다. 모든 기능들은 Core에 의해서 제어된다.

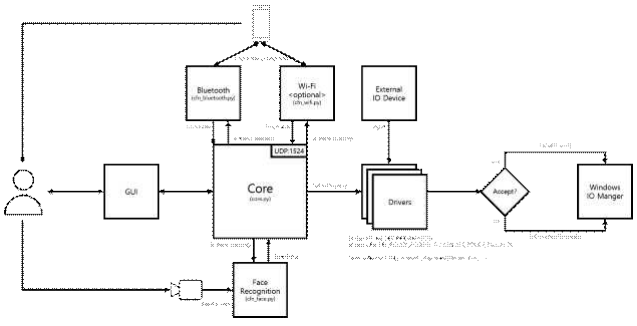
<그림 3> 얼굴인식을 적용한 예



3.2 코어

Core는 블루투스 연결 지속 여부로 사용자를 인식하는 Bluetooth 모듈, 내장된 카메라로 사용자의 모습을 촬영, 판단하는 Face Recognition 모듈, 같은 네트워크 상에 사용자가 존재하는 확인하는 Wi-Fi 모듈 등으로 구성되어 있다. 각 모듈에서의 데이터를 받아 최종적으로 노트북의 통제 여부를 결정하는 기능이다[3].

<그림 4> 프로그램의 구상도



3.3 드라이버

사용자를 인식하는 모듈로부터 사용자가 지속적으로 인식되지 않을 경우 Core는 디바이스 필터 드라이버를 로드 하게 된다. 로드 된 디바이스 필터 드라이버에 특정 값을 전달해 마우스, 키보드 등 외부로부터의 입력을 원천 차단한다. 사용자가 다시 인식된다면, Core는 디바이스 필터 드라이버를 언로드 하거나, 특정 값을 전달해 차단 상태를 해제한다.

3.4 블루투스

블루투스는 노트북과 스마트폰을 연결하여 스마트폰에서 원격으로 명령을 보낼 수 있게 하였다. 이때 블루투스는 일방향 통신 및 AES256 암호화를 적용하여 노트북과의 통신에도 보안을 고려하였다. 최근 블루투스에 관하여 취약점이 많이 보도되는 가운데 블루투스 암호화에는 Milisecond 값을 이용하여 같은 값을 암호화해도 매번 다른 암호화값이 나오도록 하여 스니핑시 보안에 더욱 강화하였다.

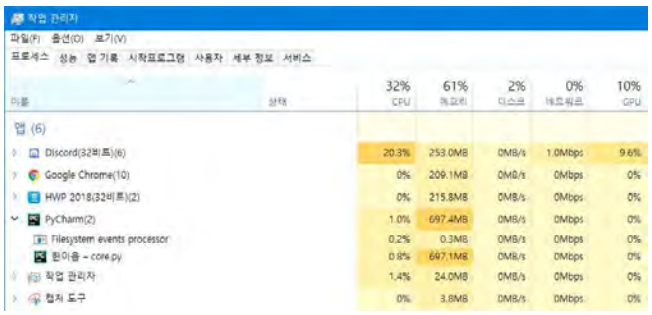
3.5 Wi-Fi

Wi-Fi 모듈도 마찬가지로 스마트폰에서 노트북의 IP를 입력하면 노트북에서 Wi-Fi를 통해 시그널을 지속적으로 보내어 노트북의 도난여부를 실시간으로 파악하게 해준다. 이때 스마트폰에서 도난여부의 시그널이 도착하지 않으면 스마트폰은 자동으로 도난으로 판단하여 사용자에게 알림 사이렌을 보낸다.

4. 성능 평가

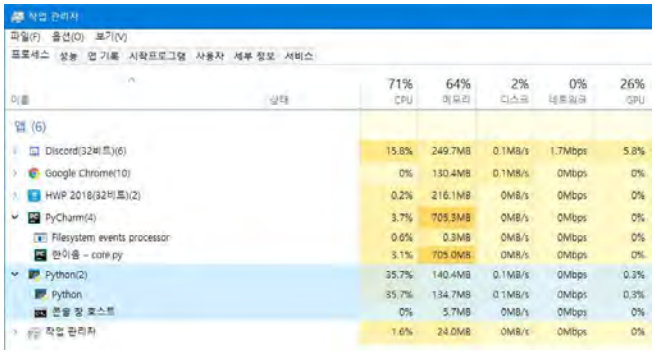
이 프로그램의 성능을 평가해보기 위해 프로그램 실행 전 총 CPU사용량과 실행 후 총 CPU사용량을 비교해보았다. 먼저 실행 전 총 CPU사용량은 다음 그림과 같이 약 25 ~ 35%를 유지하고 있다.

<그림 5> 프로그램 실행 전 CPU



그리고 프로그램을 실행했을 때 총 CPU사용량은 65 ~ 75%로 프로그램의 CPU사용량이 약 35.7% 차지함을 보였다.

<그림 6> 프로그램 실행 후 CPU



이처럼 얼굴인식 모듈을 위해 사용한 OpenCV는 사용을 하게 되면 총 CPU사용량이 기본적으로 크게 차지하게 되는데, 이는 OpenCV로 카메라 화면을 실시간으로 캡쳐하는 사이사이에 SleepTime이라는 텀을 주어 CPU의 점유율을 줄일 수 있다. 따라서 실시간으로 현재 노트북의 총 CPU사용량을 가져와 SleepTime의 값을 유동적으로 바뀌게 한다면 CPU 과부하시엔 점유율을 줄이고, 회복시 다시 원활하게 작동되게 할 수 있을 것으로 기대된다.

5. 기대 효과

사용자의 얼굴 인식 및 지문 인식 등의 기능으로 사용자가 자리를 비웠을 때 효과적으로 노트북을 보호합니다. 특히 사용자의 얼굴인식과 키 제어를 통해 허가 없이는 노트북을 사용할 수 없게 만듭니다. 이로 인해서 카페나 공개된 장소에서 일어날 수 있는 범죄 (ex. 도난, 무단으로 사용 등)로부터 효과적으로 보호할 수 있고, 범죄 발생률 또한 줄어드는 효과를 기대할 수 있다.

“본 논문은 과학기술정보통신부 정보통신창의인재양성사업의 지원을 통해 수행한 ICT멘토링 프로젝트의 결과물입니다.”

출처

[1] 한국미디어패널조사 연구팀, "2019년 한국미디어패널조사 결과 주요 내용", 2020-1-15, KISDI STAT Report 20-01

[2] What is a driver?, 2017.04.20., https://docs.microsoft.com/ko-kr/windows-hardware/drivers/gettingstarted/what-is-a-driver-

[3] Windows Kernel-Mode I/O Mananger,

2018.10.17.,
https://docs.microsoft.com/en-us/windows-hardware/drivers/kernel/windows-kernel-mode-i-o-manager