KIPS Tr. Comp. and Comm. Sys.
Vol.6, No.7 pp.307~314 pISSN: 2287-5891

https://doi.org/10.3745/KTCCS.2017.6.7.307

A Study on Image Acquisition and Usage Trace Analysis of Stick-PC

Han Hyoung Lee† · Seung Gyu Bang† · Hyun Woo Baek† · Doo Won Jeong†† · Sang Jin Lee†††

ABSTRACT

Stick-PC is small and portable, So it can be used like a desktop if you connect it to a display device such as a monitor or TV anytime and anywhere. Accordingly, Stick-PC can related to various crimes, and various evidence may remain. Stick-PC uses the same Windows version of the operating system as the regular Desktop, the artifacts to be analyzed are the same. However, unlike the Desktop, it can be used as a meaningful information for forensic investigation if it is possible to identify the actual user and trace the usage by finding the traces of peripheral devices before analyzing the system due to the mobility. In this paper, We presents a method of collecting images using Bootable OS, which is one of the image collection methods of Stick-PC. In addition, we show how to analyze the trace of peripheral connection and network connection trace such as Display, Bluetooth through the registry and event log, and suggest the application method from the forensic point of view through experimental scenario.

Keywords : Peripheral Device, Digital Forensic, Stick-PC, Connection Trace, Image Acquisition

Stick-PC의 이미지 수집 및 사용흔적 분석에 대한 연구

이 한 형† · 방 승 규† · 백 현 우† · 정 두 원†† · 이 상 진†††

요 약

스틱-PC(Stick-PC)는 크기가 작고 휴대가 용이하여 언제 어디서든 모니터나 TV 등의 디스플레이 장치에 연결하면 데스크탑 PC(Desktop PC)처럼 사용이 가능하다. 이에 따라 스틱-PC(Stick-PC)도 일반 PC처럼 각종 범죄로 연결될 수 있으며 다양한 증거들이 남아 있을 수 있다. 스틱-PC(Stick-PC)는 일반 데스크탑 PC(Desktop PC)와 같은 윈도우즈(Windows) 버전의 운영체제를 사용하고 있어 분석해야 할 아티팩트들은 동일하다. 하지만 데스크탑 PC(Desktop PC)와 달리 이동성이 있어 시스템 분석 전에 주변 기기 연결 흔적을 찾아 실 사용자 확인 및 사용 흔적을 파악하는 것이 이루어지면 포렌식 조사 시 상당히 의미 있는 정보로 사용될 수 있다. 따라서 본 논문은 스틱-PC(Stick-PC)의 이미지 수집 방법 중 하나인 Bootable OS를 이용하여 이미지를 수집하는 방법을 제시한다. 또한 레지스트리와 이벤트로그를 통해 디스플레이, 블루투스(Bluetooth) 등의 주변기기 연결 흔적과 네트워크 연결 흔적을 분석하는 방법을 제시하고 실험 시나리오를 통해 포렌식 관점에서 활용 방안을 제시한다.

키워드 : 주변기기, 디지털 포렌식, 스틱-PC, 연결 흔적, 이미지 수집

1. 서 론

스틱-PC(Stick-PC)는 인텔에서 2014년 11월 Intel Compute Stick 이라는 이름으로 출시하기 시작하면서 구글(Google), ASUS, 한성컴퓨터 등의 여러 제조사에서도 비슷한 제품을 출시하면서 형성된 하나의 제품군을 의미한다.

인텔에서는 당시 가장 최신 운영체제 버전인 Windows 8.1을 설치하여 판매하였으며, 최근엔 사양을 높이고 USB 3.0 포트를 추가한 제품을 출시[1]하면서 지속적으로 시장에 나오고 있다.

스틱-PC(Stick-PC)의 인터페이스를 살펴보면 디스플레이 장치에 연결 가능한 HDMI 단자, 키보드, 마우스 등의 장치를 연결 가능한 USB포트, 그리고 블루투스(Bluetooth) 기능을 제공한다. 또 저장 용량이 작기 때문에 추가 저장장치인 SD카드를 사용할 수 있도록 Slot이 존재한다.

스틱-PC(Stick-PC)는 데스크탑 PC(Desktop PC)에 비해 매우 작고 휴대가 용이하여 이동성을 가지면서 데스크탑 PC(Desktop PC)의 모든 기능을 수행할 수 있는 것이 특징이다. 이러한 특징으로 인해 스틱-PC(Stick-PC) 또한 일반 데스크탑 PC(Desktop PC)와 같이 디지털 범죄에 사용될 수 있다. 일반 데스크탑 PC(Desktop PC)와 동일하게 윈도우즈(Windows) 운영체제를 사용하고 있기 때문에 디지털 포렌식

※ 이 논문은 한국대학교 자유과제 학술연구비(2년)에 의하여 연구되었음.
† 준 회 원 : 고려대학교 정보보호대학원 정보보호학과 연구원
†† 비 회 원 : 고려대학교 정보보호대학원 정보보호학과 연구원
††† 종신회원 : 고려대학교 정보보호대학원 교수
Manuscript Received : March 31, 2017
Accepted : April 12, 2017
* Corresponding Author : Sang Jin Lee(sangjin@korea.ac.kr)

조사는 용이하다고 볼 수 있다. 하지만 스틱-PC (Stick-PC)의 특성상 디지털 포렌식 조사를 위해서는 먼저 사용한 사람이 누구인지 확인하는 것과 스틱-PC (Stick-PC)에 연결한 주변기기들의 정보를 찾아 언제, 어디에서 사용하였는지 행위를 추적하는 것이 디지털 포렌식 관점에서 매우 중요하다.

본 논문에서는 스틱-PC(Stick-PC) 제품군 중 인텔에서 처음 출시한 Windows 8.1이 설치된 Intel Compute Stick라는 이름의 제품을 이용하여 데이터의 원본을 훼손, 변형 없이 수집하는 방법을 제시한다. 또한 실사용자 확인 및 사용흔적을 확인할 수 있는 주변 기기를 분류하고 데이터를 분석한 결과를 바탕으로 스틱-PC(Stick-PC)의 디지털 포렌식 조사 시 활용 방안을 제시한다.

2. 관련 연구

윈도우즈(Windows) 운영체제는 이벤트로그에 다양한 이벤트들을 기록한다. 또한 윈도우즈(Windows) 운영체제에서 행해지는 모든 작업은 레지스트리를 참고하고 기록한다. 따라서 스틱-PC(Stick-PC)의 이벤트로그와 레지스트리를 통해 주변기기의 연결 흔적을 찾아 사용자의 사용흔적을 확인할 수 있다.

2.1 이벤트로그

이벤트로그는 윈도우즈(Windows) 운영체제에서 발생하는 하드웨어, 소프트웨어 및 시스템 문제에 대한 다양한 동작(이벤트)들이 바이너리 형태로 기록된 것이며, EVTX파일[2]을 수집하여 윈도우즈(Windows) 운영체제에서 기본적으로 제공하는 이벤트 뷰어를 통해 이벤트 발생 날짜, 시간, ID, 세부 정보 등을 확인할 수 있다.

2.2 레지스트리

윈도우즈(Windows)에서 수행되는 거의 모든 작업은 레지스트리를 참고하며 또한 레지스트리에 기록된다. 레지스트리 분석은 활성 레지스트리 분석과 비활성 레지스트리 분석으로 구분 된다. 활성 시스템에서의 레지스트리 분석은 RegEdit을 통해 확인 가능하고, 비활성 시스템에서의 레지스트리 분석을 위해서는 레지스트리 하이브(Hive) 파일을 수집하여 분석해야 한다[3]. Harlan Carvey는 레지스트리에 시스템 정보, 사용자 정보, 자동실행, 네트워크 정보 등의 포렌식 조사 시 유용하게 사용될 가치 있는 많은 정보들이 존재한다고 하였다[4].

3. Stick-PC의 이미지 수집 방법

스틱-PC(Stick-PC)의 경우 출력단자가 HDMI밖에 존재하지 않는다. 따라서 기존의 하드웨어 이미징 장비인 Falcon, TD3 등을 이용하는 방법으로는 정상적인 이미지 획득이 불가능하다. 따라서 스틱-PC(Stick-PC)의 사본 이미지를 생성하는 방법은 Chip-Off, 소프트웨어를 이용한 이미지 수집, Bootable OS를 이용한 이미지 수집 세 가지 방법이 존재한다. 본 장에서는 각각의 스틱-PC(Stick-PC) 이미지 수집 방법과 그에 따른 장단점에 대해 설명하고자 한다.

3.1 Chip-off

데이터를 복사하는 방법 중 첫 번째 방법으로 하드웨어를 직접 분리하여 이미지를 수집하는 방법이다. 하지만 Chip이 매우 작아 분리하기 어려울 뿐만 아니라 이미지 수집 후 재결합 시에 결함이 생길 수 있어 주의가 필요하다. 또한 Chip을 분리하는 것에 성공하였다 하더라고 저장된 데이터를 읽기 위해서는 고가의 장비를 필요로 한다.

3.2 Software

소프트웨어를 이용한 이미징 방법의 경우 스틱-PC(Stick-PC)를 구동한 뒤, FTK Imager[5], EnCase[6] 등의 소프트웨어를 사용하여 이미지를 수집하는 방법이다. 이는 Stick-PC에 직접 설치하여 이미지를 수집해야 한다. 따라서 구동 및 소프트웨어 사용에 의한 원본 데이터의 변화가 생기기 때문에 훼손은 불가피하다.

3.3 Bootable OS

Bootable OS를 이용한 이미지 수집 방법은 스틱-PC(Stick-PC)에 기본적으로 Windows 8.1이 설치되어 있으나, BIOS에서 추가적으로 리눅스(Linux) OS를 선택하여 부팅 할 수 있게 되어 있다. 따라서 이를 이용해 리눅스(Linux) OS가 설치된 USB[7]로 부팅을 한 후에 윈도우즈(Windows) OS가 설치된 드라이브를 마운트하여 리눅스(Linux) OS에서 제공하는 이미지 수집 명령어(dd)를 사용하여 이미지를 수집하는 것이다.

앞서 말한 두 가지의 이미지 수집 방법의 단점 때문에 본 논문에서는 세 번째 방법인 Bootable USB를 사용하여 이미지 수집을 진행하였다. 이 때 원본 데이터의 무결성을 확인하기 위해 총 10번 반복하여 이미지를 수집해 얻은 복사본의 해시 값을 비교하였다.

그 결과 Table 1과 같이 해시 값이 모두 동일함을 확인함으로써 원본 데이터의 무결성을 확인할 수 있었다.

Table 1. Compare Hash Values of Collected Images

Count	Hash(MD5)
1	2c7da27f2b884b1d5f6cbabe49f15b
2	2c7da27f2b884b1d5f6cbabe49f15b
3	2c7da27f2b884b1d5f6cbabe49f15b
4	2c7da27f2b884b1d5f6cbabe49f15b
5	2c7da27f2b884b1d5f6cbabe49f15b
6	2c7da27f2b884b1d5f6cbabe49f15b
7	2c7da27f2b884b1d5f6cbabe49f15b
8	2c7da27f2b884b1d5f6cbabe49f15b
9	2c7da27f2b884b1d5f6cbabe49f15b
10	2c7da27f2b884b1d5f6cbabe49f15b

4. 주변 기기 종류와 연결 흔적 분석

스틱-PC(Stick-PC)는 휴대가 용이하기 때문에 모니터나 TV등의 디스플레이 장치에 연결하여 언제 어디서든 사용할 수 있다. 따라서 스틱-PC(Stick-PC)에 대한 디지털 포렌식 분석을 하는 것과 동시에 주변 기기에 대한 연결 흔적을 확인함으로써 분석관은 사용자가 스틱-PC(Stick-PC)를 사용한 장소를 파악할 수 있고 이후 수사에 도움이 될 수 있다. 본 장에서는 스틱-PC(Stick-PC)의 특징에 따라 스틱-PC (Stick-PC)를 사용할 모니터 등과 같은 주변 기기를 사용하였을 경우 확인할 수 있는 흔적을 분류하고 분석 방법을 제시한다.

4.1 디스플레이 장치 사용흔적

스틱-PC(Stick-PC)를 사용하기 위해서 반드시 모니터, 프로젝터 등의 디스플레이 장치에 연결되어야만 한다. 이렇게 디스플레이 장치에 연결하였을 경우 연결된 디스플레이 장치에 대한 정보는 레지스트리에 모두 기록된다.

레지스트리에 남는 정보를 살펴보면 Fig. 1과 같이 HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Enum\DISPLAY 하위키로 모니터의 모델명으로 된 레지스트리키가 생성된다. 그리고 인스턴스 ID로 하위키가 생성되어 모델별로 고유한 컨테이너 ID를 포함한 정보들이 저장된다.

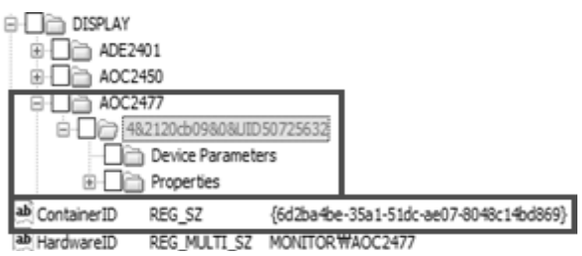

Fig. 1. Monitor Instance ID and Container ID Information

다음의 예시에서 모델명 AOC2477로 레지스트리키가 생성되었고 장치 인스턴스 ID가 4&2120CB09&0&UID50725632로 하위키가 생성되어 그 안에 Container ID {6d2ba4be-35a1-51dc-ae07-8048c14bd869}가 생성된 것을 확인할 수 있다.

단, 동일한 모델의 다른 제품에 연결할 경우에는 가장 마지막에 연결한 제품의 정보가 저장된다.

또한 Fig. 1에서는 추가적으로 HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\DISPLAY\모델명\장치 인스턴스 ID가 생성되며 하위키로 Properties와 DeviceParameters 키가 생성된다.

DeviceParameters키에는 Fig. 2와 같이 EDID가 존재하며 내용에 모니터의 고유 일련번호가 저장된다. 위의 예시에서 제품의 고유 일련번호는 AAKF39A001639임을 확인할 수 있다.

Properties는 Fig. 3과 같이 바이너리 값으로 장치 설치 시간, 마지막 연결 시간이 저장된다. 시간 값은 Little Endian 방식의 16진수로 저장되어 별도의 변환 과정을 통해 시간을 확인할 수 있다.

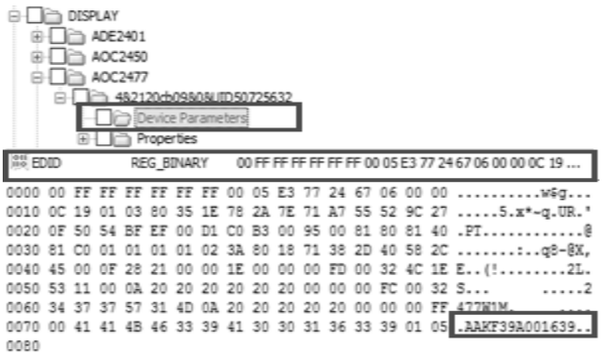

Fig. 2. The Serial Number of the Monitor

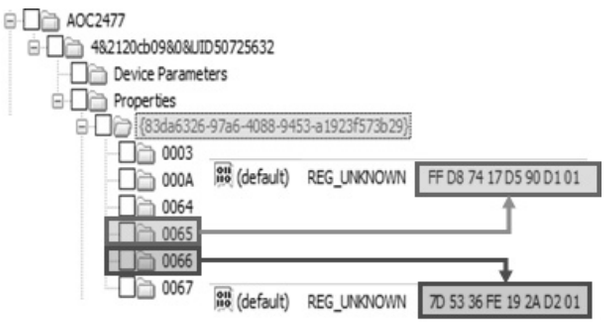

Fig. 3. Monitor Connection Time Information

다음의 예시에서는 {83da6326-97a6-4088-9453-a1923f573b29} 키 하위에 0065키 안에 설치 시간이 FFD87417D590D101의 16진수 값으로 저장되어 있고 이를 변환하면 2016-04-07 22:55:06인 값이 저장되어 있음을 확인할 수 있다. 또한 0066 키 안에 마지막 연결 시간이 7D5336FE192AD201의 16진수 값으로 저장되어 있고 이를 변환하면 2016-10-20 00:03:47인 값이 저장되어 있음을 확인할 수 있다.

레지스트리 키와 저장되는 정보를 정리하면 Table 2와 같다.

Table 2. Registry Keys and Information of Monitor

Registry Key	Information
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\DISPLAY\ModelName	ModelName, Container ID
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\DISPLAY\ModelName\Device Instance\DeviceParameters\EDID	Serial Number
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\DISPLAY\ModelName\DeviceInstance\Properties	First Connection Time, Last Connection Time

레지스트리를 통해 연결된 모니터의 모델명, 설치 시간, 마지막 연결 시간을 확인할 수 있으므로 이 정보를 확인하여 사용자가 언제 어떤 장치에 연결하여 사용하였는지 흔적을 확인할 수 있으며 연결했던 장치의 위치를 안다면 어디에서 사용하였는지도 확인할 수 있다.

또한 이벤트로그에서도 디스플레이 장치에 대한 연결 흔적을 확인할 수 있다. C:\Windows\System32\Winevt\Logs\Microsoft-Windows-DeviceSetupManager%4Admin.evtx의 이벤트로그 파일에 모델명, Container ID, 연결 시간 등이 저장되므로 레지스트리에서 확인한 설치 시간과 마지막 연결 시간 사이의 기간 동안 사용 흔적을 추가적으로 살펴 볼 수 있다[8].

Fig. 4는 로그파일을 이벤트뷰어로 확인한 그림이다. 예시에서 Event ID는 112, 모델명은 2477W1M, Container ID는 {6d2ba4be-35a1-51dc-ae07-8048c14bd869}로 시간과 함께 저장된 것을 확인할 수 있다.

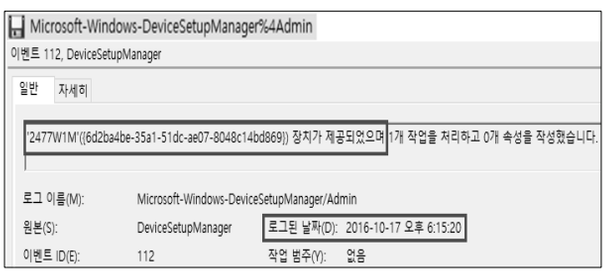

Fig. 4. Viewing Log Files with Event Viewer

4.2 블루투스(Bluetooth) 기기 연결 흔적

스틱-PC(Stick-PC)는 입력 장치연결 등을 위하여 블루투스(Bluetooth) 기능을 지원한다. 블루투스(Bluetooth)로 장치를 연결하였을 경우 디스플레이 장치와 마찬가지로 레지스트리에 정보가 저장된다.

레지스트리에서 블루투스(Bluetooth) 정보를 살펴 보면 먼저 Fig. 5와 같이 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\DeviceAssociationFramework\Store 키에 연결했던 Bluetooth 기기의 MAC 주소로 하위키가 생성되므로 해당 기기의 MAC 주소를 확인할 수 있다.

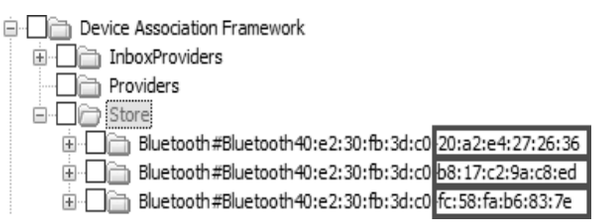

Fig. 5. MAC Address of Bluetooth Devices

Fig. 6을 보면 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\BTHENUM 하위키로 Dev_기기의 MAC 주소가 생성되어 Container ID, 이름, HardwareID 등의 정보가 저장된다.

다음의 예시에서는 Container ID는 {2d5a0bae-281a-5547-86d9-2a1a4b867391}, 이름은 Administrator의 iPhone, HardwareID는 BTHENUM\Dev_20A2E4272636로 저장되어 있는 것을 확인할 수 있다.

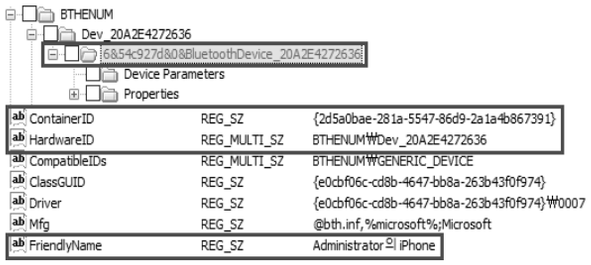

Fig. 6. Bluetooth Container ID and Hardware ID Information

또한 하위 키로 DeviceParameters 키와 Properties 키가 하위키로 생성되어 각 기기의 Unique ID와 설치 시간, 마지막 연결 시간 등의 정보가 저장된다.

DeviceParameters 키에는 Fig. 7과 같이 UniqueID가 존재하며 내용에 기기의 MAC 주소가 저장된다. 다음의 예시에서 Unique ID는 20A2E4272636임을 확인할 수 있다.

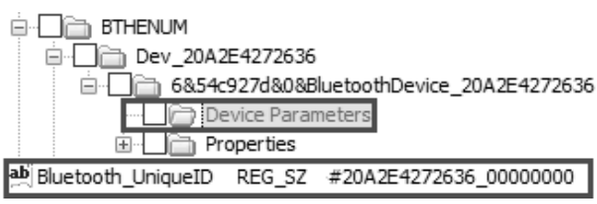

Fig. 7. Bluetooth Unique ID

이어서 Properties도 Fig. 8과 같이 바이너리 값으로 장치 설치 시간, 마지막 연결 시간이 저장된다.

다음 예시에서는 {83da6326-97a6-4088-9453-a1923f573b29} 키의 하위키인 0065 키 안에 설치 시간이 50F948E12628D201의 16진수 값으로 저장되어 있고 이를 변환하면 2016-10-17 12:31:00인 값이 저장되어 있음을 확인할 수 있다. 또한 0066키 안에 마지막 연결 시간이 B6F92DFD192AD201의 16진수 값으로 저장되어 있고 이를 변환하면 2016-10-20 00:03:45인 값이 저장되어 있음을 확인할 수 있다.

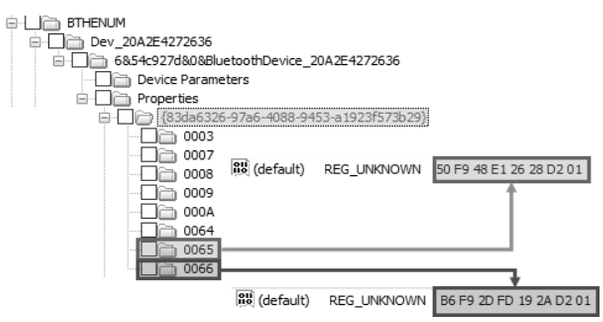

Fig. 8. Bluetooth Connection Time Information

또 기기의 이름을 확인할 수 있는 레지스트리키가 존재하는데 Fig. 9와 같이 HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\BTHPORT\Parameters\Devices 하위에 MAC 주소 키로 생성되어 기기의 이름이 저장된다.

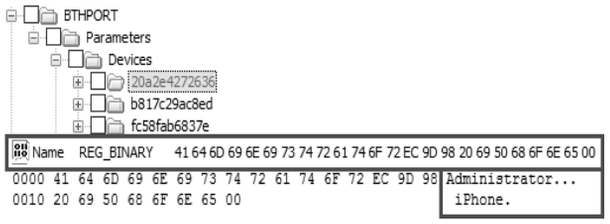

Fig. 9. Name of Bluetooth Device in the Registry

다음 예시에서는 MAC 주소인 20A2E4272636키에 Name이라는 필드가 존재하며 내용은 Administartor iPhone임을 확인할 수 있다.

위의 내용을 Table 3에 요약하였다.

Table 3. Registry Keys and Information of Bluetooth

Registry Key	Information
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\DeviceAssociationFramework\Store	Device MAC address
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\BTHENUM\Dev_MAC	Device MAC address, ContainerID, Name, HardwareID
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\BTHPORT\Parameters\Devices\MAC	DeviceName, ServiceName
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\BTHENUM\Dev_MAC\Properties	First Connection Time, Last Connection Time

레지스트리에서 얻은 정보를 활용해 이벤트로그에서 추가적인 연결 시간을 확인할 수 있는데 디스플레이 섹션에서 언급했던 로그파일과 동일한 파일에서 해당 정보를 확인할 수 있다.

연결했던 블루투스(Bluetooth)기기의 이름, MAC주소, 시간을 확인하여 사용자를 특정할 수 있고 스틱-PC(Stick-PC)의 소유주인지 확인하는 정보로 사용할 수 있다.

4.3 USB 탈부착 흔적

스틱-PC(Stick-PC)의 USB 등 이동식 저장매체 연결 흔적은 레지스트리에서 확인할 수 있다. HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Enum\USB 키에 Fig. 10과 같이 마우스, 키보드, USB허브 등의 하드웨어 ID, 컨테이너 ID 등의 정보가 저장된다.

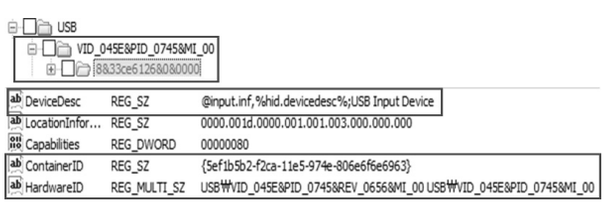

Fig. 10. Hardware ID, Container ID Information of USB

USB의 HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Enum\HID\{하드웨어 ID}키에서도 Fig. 11과 같이 하드웨어 ID, 컨테이너 ID와 장치의 종류(키보드, 마우스 등)를 나타내는 정보들이 저장된다.

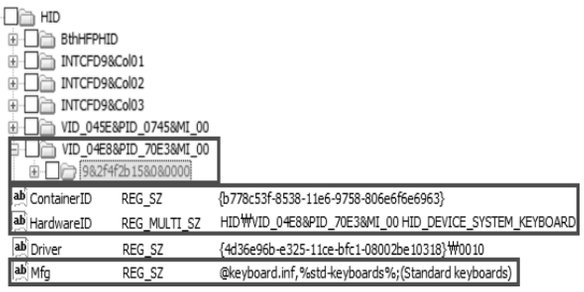

Fig. 11. Hardware ID, Container ID Information of USB

USB의 HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Enum\HID\{하드웨어 ID}\Properties 키에는 Fig. 12와 같이 장치의 최초 연결 시간과 마지막 연결 시간 정보가 저장된다.

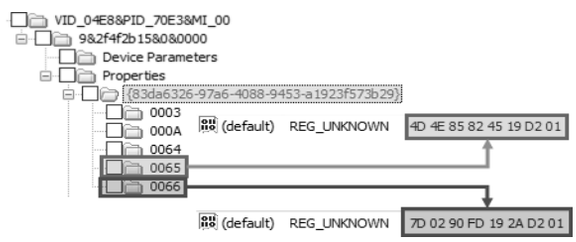

Fig. 12. USB(keyboard) Connection Time Information

이 정보를 활용해 앞에서 언급한 이벤트로그 파일에서 추가적인 연결 시간정보를 확인할 수 있다.

위의 내용을 Table 4에 요약하였다.

Table 4. Registry Keys and Information of USB

Registry Key	Information
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Enum\USB\{Hardware ID}	Hardware ID Container ID
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Enum\HID\{Hardware ID}	Hardware ID Container ID DeviceDescription
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Enum\HID\{Hardware ID}\Properties	First Connection Time, Last Connection Time

4.4 네트워크 연결 흔적

네트워크에 연결한 다양한 흔적이 레지스트리에 저장되므로 레지스트리를 통해 네트워크 연결 흔적들을 찾음으로써 사용자의 사용 흔적을 알아낼 수 있다[9].

먼저 스틱-PC(Stick-PC)에 내장된 네트워크 인터페이스 카드(NIC)관련 정보를 레지스트리 키를 통해 찾을 수 있다.

Fig. 13을 보면 해당 키는 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkCards 이다. 이 키에서 Description에 네트워크 인터페이스 카드 이름이 저장되어 있고, ServiceNameKey에 할당된 GUID(Globally Unique Identifier)가 저장되어 있다.

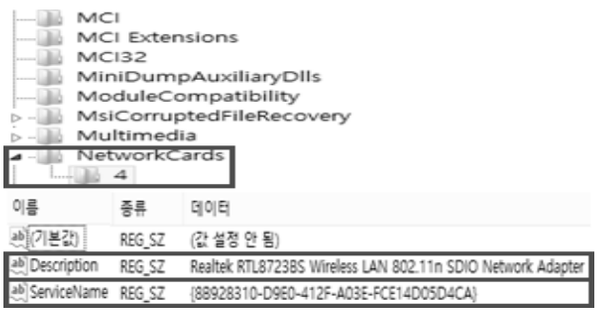

Fig. 13. Network Interface Card Information

위의 예시에서 네트워크 인터페이스 카드 이름이 Realtek RTL8723BS Wireless LAN 802.11n SDIO Network Adapter 이고, 할당된 GUID는 {8B928310-D9E0-412F-A03E-FCE14D05D4CA}임을 확인할 수 있다.

이어서 인터페이스에 할당된 GUID를 통해 무선 네트워크 IP 주소와 그와 관련된 정보들 찾을 수 있다.

Fig. 14를 보면 HKEY_LOCAL_MACHINE\SYSTEM\ControlSet00X\Services\Tcpip\Parameters\Interface\{GUID} 키에 IP 주소와 게이트웨이 주소, 도메인 이름 등을 찾을 수 있다. 단, 이 정보는 가장 마지막에 연결했던 네트워크의 정보만 남아 있고 이전에 연결했던 네트워크의 정보는 남아있지 않다.

Fig. 14. Assigned IP Address Information

위의 예시에서 IP 주소 10.16.22.168가 할당되었고, 게이트웨이 주소는 10.16.22.1이며, 도메인 korea.ac.kr이 할당된 것을 확인할 수 있다.

추가적으로 확인할 수 있는 정보는 연결했던 무선 네트워크 이름과 최초 연결 시간, 마지막 연결 시간이다.

해당 정보는 Fig. 15에서 보이는 것처럼 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\NetworkList\Profiles\ProfileGUID키에서 찾을 수 있다. Description은 무선 네트워크 이름이 저장되고, DateCreated에는 최초 연결 시간이, DateLastConnected에는 마지막 연결 시간이 저장된다. 시간 값은 바이너리 구조로 되어 있으며 4바이트 부분으로 나누어 각 리틀 엔디언 방식의 16진수로 연, 월, 요일, 일, 시, 분, 초, 밀리초를 나타낸다. 따라서 16진수의 4바이트 값을 10진수로 변환하면 시간을 해석할 수 있다.

Fig. 15. Connected AP Information

위의 예시에서는 무선 네트워크 이름은 KoreaUniv AP이고, 최초 연결 시간은 E007090003001C000D0035000E006E00 이고 이 값을 해석하면 2016-09-28 13:53:14.110이 된다. 마지막 연결 시간도 동일한 방법으로 구하면 2016-10-26 16:45:54.932임을 알 수 있다.

위의 내용을 Table 5에 요약하였다.

Table 5. Registry Keys and Information of Network

Registry Key	Information
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkCards	NIC Information, Interface GUID
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interface\{GUID}	IP address, Gateway address, Domain name
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles	Wireless Network Name, First Connection Time, Last Connection Time

네트워크 연결 흔적을 통해 사용했던 위치를 확인할 수 있어 알리바이에 유용한 정보로 활용할 수 있다.

5. 포렌식 관점에서 활용 방안 검증

4장의 스틱-PC(Stick-PC) 고유의 디지털 포렌식 조사 대상 및 방법을 검증하고, 실제 스틱-PC(Stick-PC)가 사용된 위치를 특정할 수 있는지 확인하기 위하여 시나리오를 세우고 실험을 진행하였다.

5.1 실험 개요 및 시나리오

실험은 스틱-PC(Stick-PC)를 다양한 디스플레이 장치에 연결하였을 경우 스틱-PC(Stick-PC) 디지털 포렌식 분석을 통하여 스틱-PC(Stick-PC)가 사용된 위치를 특정할 수 있는지 판단하기 위하여 실험을 진행하였다. 실험의 시나리오는 다음과 같다.

OO기업에 다니고 있는 A씨는 권한이 없음에도 불구하고 기업의 기밀문서에 접근하기 위하여 스틱-PC(Stick-PC)를 사용하였다. A씨는 사람들의 눈을 피해 회사 내의 빈자리들을 찾아다니면서 다른 직원의 모니터에 자신이 보유하고 있는 Stick-PC를 연결한 후 네트워크 접속을 시도하였으나 실패하였다. 이후 퇴근 중 보안경보기에 스틱-PC(Stick-PC)가 포착되어 압수되었다. 이에 기업의 보안담당자는 A씨가 스틱-PC(Stick-PC)의 용도를 확인하기 위한 조사가 시작되었다.

이 시나리오를 바탕으로 본 논문에서 확인한 조사 대상을 통해 A씨의 범죄행위를 입증하려고 한다.

시나리오에 사용된 디스플레이 장치는 모두 모니터로 AOC 제품 2대, SAMSUNG 제품 4대, LG 제품 1대로 총 7대의 모니터에 스틱-PC(Stick-PC)를 연결하였다 해제하였다. 보안담당자가 획득한 스틱-PC(Stick-PC)의 정보는 Table 6과 같다.

Table 6. Sepecification of Stick-PC

Model Name	STCK1A32WFC
Serial Number	GEFC524005D2
Size	21.9 GB
OperatingSystem	Windows 8.1 K with Bing

5.2 스틱-PC(Stick-PC) 분석

보안담당자는 먼저 획득한 Stick-PC의 분석용 사본 이미지를 생성하기 위하여 Bootable OS를 이용하여 스틱-PC(Stick-PC)를 이미징하였다. 생성된 이미지 파일의 상세 정보는 Table 7과 같다.

Table 7. Information of Stick PC's Image File

File Name	Image
Size	21.9 GB
Hash	ca479d4279b26d4eea3c74dbeb5ed0f8

다음으로 연결된 디스플레이를 확인하기 위하여 사본 이미지에서 레지스트리를 추출하여 Table 2의 경로에서 연결된 디스플레이의 정보를 확인하였다. 총 7개의 연결 흔적을 확인할 수 있었으며 이에 대한 정보는 Table 8과 같다.

따라서 A씨는 총 7개의 모니터가 설치되어 있던 장소에서 스틱-PC(Stick-PC)를 사용하였다는 것을 확인할 수 있다. 따라서 분석관은 A씨가 다른 직원의 자리에서 스틱-PC(Stick-PC)를 연결하여 동작시켰다는 사실을 확인할 수 있으며 모니터의 시리얼 번호를 조회하면 실제 모니터를 사용하고 있던 담당자의 성명까지 확인이 가능하다.

6. 결론 및 향후계획

본 논문에서는 스틱-PC(Stick-PC)의 BIOS에서 제공하는 부팅 설정 방법을 활용하여 Bootable USB를 이용한 이미지 수집을 진행하였고, 수집한 이미지의 해시 값 비교를 통해 원본 데이터의 무결성을 확인하였다.

또한 레지스트리에서 디스플레이장치, 블루투스, USB 등의 주변 기기의 고유 일련 번호, 최초 연결 시간 정보와 마지막 연결 시간 정보 등과 연결했던 네트워크 이름, 연결 시간 정보를 확인하였다. 이 정보를 토대로 이벤트로그에서 최초 연결 시간과 마지막 연결 시간 사이의 추가적인 연결 시간 정보를 확인하였다. 따라서 이동성을 지닌 Stick-PC의 시스템 분석 전에 레지스트리와 이벤트로그에서 주변기기의 고유 일련 번호 및 연결 시간 정보를 추출하여 실사용자 확인 및 언제, 어디서 사용 하였는지 사용 흔적을 파악하고 타임라인을 구성함으로써 디지털 포렌식 조사 시 도움이 될 수 있을 것이다.

향후 분석시간을 단축하기 위하여 연결 기기들의 정보가 저장된 레지스트리와 이벤트로그 파일을 추출하여 타임라인을 구성하는 도구를 제작을 진행할 예정이다.

References

[1] Intel [internet], http://www.intel.co.kr/content/www/kr/ko/compute-stick/intel-compute-stick.html.

[2] Yonghak Shin, Junyoung Cheon and Jongsung Kim, “Study on Recovery Techniques for the Deleted or Damaged Event Log (EVTX) Files,” *Journal of the Korea Institute of Information Security & Cryptology*, Vol.26, No.2, pp.387-396, 2016.

Table 8. Device Lists and Information Connected to the Display Unit

Model Name	Serial Number	First Connection Time	Last Connection Time
AOC 2450	DENB8HA000284	2017-03-29 16:35:24 (+09:00)	2017-03-29 16:37:37 (+09:00)
AOC 2477	AAKF39A001670	2017-03-29 16:43:43 (+09:00)	2017-03-29 16:46:46 (+09:00)
SAMSUNG S23B350	S23B350	2017-03-29 16:02:49 (+09:00)	2017-03-29 16:04:25 (+09:00)
SAMSUNG S23C340	S23C340	2017-03-29 16:19:56 (+09:00)	2017-03-29 16:20:43 (+09:00)
SAMSUNG S24C340L	S24C340	2017-03-29 16:14:01 (+09:00)	2017-03-29 16:16:01 (+09:00)
SAMSUNG T24D390	T24D390	2017-03-29 16:25:22 (+09:00)	2017-03-29 16:30:06 (+09:00)
LG	24EA55	2017-03-29 17:00:13 (+09:00)	2017-03-29 17:03:05 (+09:00)

[3] Sang Jin Oh and Kyu Ho kim, “A Study on The Procedure Analysis Vulnerability for Security Incidents using The Registry Parsing,” *Conference Workshop of The Institute of Electronics Engineers of Korea*, pp.287-290, 2016.
[4] Harlan Carvey, “The Windows Registry as a forensic resource,” *Digital Investigation*, Vol.2, Issue 3, pp.201-205, 2005.
[5] Forensic Toolkit (FTK) [internet], http://accessdata.com/solutions/digital-forensics/forensic-toolkit-ftk.
[6] EnCase Forensic [internet], https://www.guidancesoftware.com/encase-forensic?cmpid=nav_r.
[7] Ubuntu(Linux OS) [internet], https://www.ubuntu.com/download.
[8] kali-km_Security Study [Internet], kali-km.tistory.com/entry/Windows-Event-Log-2---주요-이벤트-로그.

이 한 형
e-mail : ykjocn@naver.com
2015년 강원대학교 전기전자공학부(학사)
2015년～현 재 고려대학교 정보보호대학원 정보보호학과 연구원
관심분야 : Digital Forensic, Reverse Engineering

방 승 규
e-mail : bangkert89@naver.com
2015년 목포대학교 정보보호학과(학사)
2017년 고려대학교 정보보호대학원 정보보호학과 연구원
관심분야 : Digital Forensic, Reverse Engineering

백 현 우
e-mail : hyunwoob.24@gmail.com
2015년 경희대학교 전자전파공학과(공학사)
2017년 고려대학교 정보보호대학원 정보보호학과 연구원
관심분야 : Digital Forensic, Reverse Engineering

정 두 원
e-mail : dwjung77@korea.ac.kr
2011년 고려대학교 산업경영공학과(학사)
2011년～현 재 고려대학교 정보보호대학원 정보보호학과 연구원
관심분야 : Digital Forensic, Information Security, Big Data Analysis

이 상 진
e-mail : sangjin@korea.ac.kr
1987년 고려대학교 수학과(학사)
1989년 고려대학교 수학과(석사)
1994년 고려대학교 수학과(박사)
1989년～1999년 ETRI 선임연구원
1999년～현 재 고려대학교 정보보호대학원 교수
2008년～현 재 고려대학교 디지털포렌식연구센터 센터장
관심분야 : Digital Forensic, Steganography, Hash Function