ISSN(Print) : 1976-5304, ISSN(Online) : 2713-931X
https://kdfs.jams.or.kr, DOI : 10.22798/kdfs.2022.16.1.24

Bootice의 은닉 영역에 대한 안티포렌식 및 안티포렌식 대응에 대한 연구

홍 표 길*, 김 도 현**
부산가톨릭대학교 컴퓨터공학과 (대학원생)*, (조교수)**

A Study of an anti-forensics and counter anti-forensics of Bootice's hidden area

Pyo-Gil Hong*, Dohyun Kim**
Dept. of Computer Engineering, Catholic University of Pusan (Graduate Student)*, (Assistant Professor)**

요 약

USB 메모리는 대표적인 이동식 저장 매체 중 하나이며 휴대성과 가용성이 용이하여 여러 분야에서 널리 사용되고 있다. 그러나 USB 메모리가 널리 사용되면서 악의적인 사용자가 USB 메모리에 데이터를 은닉하는 등의 안티포렌식 행위를 하여 디지털 포렌식 조사를 방해할 가능성이 있다. 본 논문에서는 저장매체의 파티션을 관리하는 프로그램인 Bootice에 숨겨진 영역을 생성하는 기능이 존재하여 악용할 경우 디지털 포렌식 조사를 방해할 가능성이 있기 때문에 이 영역을 분석하여 구조를 파악하고 은닉된 파티션을 탐지하는 방안을 제안한다. 또한 은닉 영역에 파일을 저장하고 디지털 포렌식 도구로부터 탐지를 회피하기 위한 방법을 제안한다. 이 연구는 은닉 영역을 생성하는 기능이 존재하는 Bootice 도구를 분석하여 안티포렌식 대응 방안과 안티포렌식 기법 발전에 기여한다.

주제어 : 디지털 포렌식, 안티포렌식 기법, 은닉 파티션, 데이터 은닉, USB 메모리

ABSTRACT

USB flash drive is one of the representative removable storage media and is widely used in various fields due to its ease of portability and availability. However, there is a possibility that malicious users may interfere with digital forensic investigation by performing anti-forensic actions such as hiding data in USB flash drive. In this paper, Bootice, a program that manages partitions of storage media, has a function of creating hidden areas, and if abused, it may interfere with digital forensic investigation, so we propose a plan to analyze this area to identify the structure and detect hidden partitions. It also proposes a method for storing files in hidden areas and avoiding detection from digital forensics tools. This study contributes to the development of anti-forensics countermeasures and anti-forensics techniques by analyzing Bootice tools that have the function of generating hidden areas.

Key Words : Digital Forensics, Anti-Forensics Technique, Hidden Partition, Data Hiding, USB Flash Drive

Ⅰ. 서 론

USB 메모리는 반도체 기술의 발전으로 속도, 용량 그리고 휴대성이 증가하면서 가격이 점차 저렴해지고 있기에 클라우드 스토리지 서비스가 등장한 이후에도 널리 사용되고 있다. 그러나 인터넷의 발전으로 안티포렌식에 대한 정보를 손쉽게 접할 수 있게 되면서 악의적인 사용자가 USB 메모리에 안티포렌식 기법을 적용하여 디지털 포렌식 조사를 방해할 가능성이 있다.

본 논문에서 분석하는 대상인 Bootice는 파티션 관리 및 편집하는 도구다[1]. 이 도구에는 저장매체에 은닉 영역을 생성하는 기능이 존재하는데 이 기능을 디지털 포렌식 조사를 방해하는 안티포렌식으로 악용할 수 있기

※ 이 논문은 2021년도 부산가톨릭대학교 교내연구비에 의하여 연구되었음
• Received 04 December 2021, Revised 07 December 2021, Accepted 31 March 2022
• 제1저자(First Author) : Pyo-Gil Hong (Email : kinvyrlfl744@gmail.com)
• 교신저자(Corresponding Author) : Dohyun Kim (Email : dohyun@cup.ac.kr)

때문에, 우리는 본 논문을 통해 Bootice의 이 기능과 은닉 영역과 관련된 파일 시스템을 분석하여 은닉 영역을 탐지하는 방법을 제안한다. 이를 위해 은닉 영역과 관련된 파일 시스템 구조를 분석하여 기존에 널리 사용되는 디지털 포렌식 도구의 은닉 영역을 탐지 방법을 파악한다. 또한, 분석 결과를 통해 파일 시스템의 데이터를 인위적으로 변경하여 디지털 포렌식 도구의 은닉 영역 탐지 기능을 회피하는 방법과 그 영역에 은닉할 파일을 저장하는 방안을 제안함으로써 Bootice가 안티포렌식으로 어떻게 활용될 수 있는지 확인하고 이렇게 은닉된 은닉 영역을 탐지하기 위한 절차와 방법도 제안한다.

현재까지 Bootice로 생성한 은닉 영역의 구조를 분석한 내용과 이에 대한 안티포렌식 대응에 관한 연구는 진행되지 않았기 때문에 기존의 디지털 포렌식 조사 과정에서는 이 영역을 탐지하기 어려울 것이다. 본 논문은 디지털 포렌식 조사를 방해하는 안티포렌식 방법 중 은닉 영역 생성에 대한 안티포렌식 대응 연구에 이바지할 수 있을 것이다.

Ⅱ. 관련 연구 및 동향

안티포렌식 기법 및 도구에 대해 기존의 연구를 소개하거나 분석한 논문들이 있었다. M Gül외 1명은 다양한 안티포렌식 기법에 대해 조사하였으며 이러한 기법들이 사용되었는지 확인하는 방안에 대해 연구했다[2]. TR Etow은 안티포렌식 기법과 도구, 안티포렌식 기술 대응 기법들을 조사하고 디지털 포렌식 도구에 대해 실험과 연구를 진행하였다. 연구 결과, 확장자 조작과 같은 안티포렌식 기법은 조사에 실질적인 영향을 미치지 않으며 디지털 포렌식 도구로 쉽게 감지가 가능한 것을 발견했다. 그러나 스테가노그래피와 같은 안티포렌식 기법은 조사 과정에 영향을 미치며 가장 효율적인 안티포렌식 기법은 볼륨 암호화인 것을 확인했다[3]. JPA Yaacoub 외 3명은 각종 저장 매체에 대한 디지털 포렌식 기법과 도구들을 조사하고 연구 방향성들에 대해 논의했으며 안티포렌식 도구와 기법 그리고 이에 대한 탐지 및 대응 방안을 조사했다. 그들은 안티포렌식에 대해 머신 러닝 접근 방식을 기반으로 안티포렌식 행위를 탐지하는 방안과 증거 변경을 방지하기 위한 추가적인 정보 보호 솔루션 사용을 제안했으며 이러한 교육의 필요성을 강조했다[4]. H Majed 외 2명은 안티포렌식 공격과 기법들을 조사하고 이 기법과 공격들에 대한 한계점과 대응 방안을 조사했으며 보안 정책 준수 및 디지털 포렌식 조사관의 안티포렌식 기법 연구 필요성을 강조했다[5]. Mohamad 외 2명은 파일 시스템에 대한 안티포렌식 유형, 기법과 도구들에 대해 조사하고 디지털 포렌식 도구만으로는 한계가 존재하기 때문에 디지털 포렌식 조사관들은 파일 시스템에 대한 안티포렌식 기법에 대한 조사의 필요성을 강조했다[6].

USB 메모리 또는 포트를 이용한 안티포렌식 기법에 대한 논문들이 있었다. Jewan Bang 외 1명은 보안 USB 메모리 컨트롤러의 명령을 사용하여 USB 메모리의 보안 기능을 우회하는 기법을 제시하고 이 기법을 토대로 보안 USB 메모리의 우회를 설계 및 구현했다[7]. Jaein Kim 외 5명은 USB 메모리의 내부 데이터 보호 기법 중 하나인 하드웨어 기반 파티션 분할 방식에 사용되는 SM3254AE 칩셋에 보안 영역과 통신하기 위한 명령어에서 취약점을 발견했다. 그 결과 보안 영역에 저장된 데이터를 인가되지 않은 사용자가 접근이 가능한 것을 확인했다[8]. Stéphanie Blanchet은 2014년에 발견된 USB 메모리 컨트롤러에 펌웨어를 이용한 취약점에 대해 논의하고 해결 방안을 제안했다[9]. 김소희 외 2명은 USB 메모리에 기존의 이미징 도구로 탐지되지 않는 은닉 영역인 CD 영역을 생성하고 CD 영역에 패킷을 보내는 방법을 통해 존재 유무 및 파일 데이터를 확인하는 방안을 제안했다[10]. Nir 외 2명은 최근 USB 메모리 및 프로토콜에 기반한 공격에 대해 조사 및 분류를 제시하고 체계화했다. 그리고 기존 방식의 탐지 방법에 대한 한계점을 제시했다[11]. Tyler Thomas 외 3명은 특수 제작된 공격용 USB 메모리인 Hak5의 USB Rubber Ducky와 Bash Bunny를 사용하여 USB 기반 공격 플랫폼에서 생성되는 메모리 포렌식 아티팩트를 연구했다. 그 결과, Windows 진단 이벤트를 활용하여 공격용 USB 메모리가 분리된 후 최대 11시간까지 USB 메모리 사용과 관련된 메타 데이터를 수집할 수 있었으며 메모리 분석을 통해 수행한 스크립트를 평문으로 찾을 수 있었다[12]. Marian TICU는 USB 기반 사이버 공격의 일부를 소개하고 이러한 공격을 방지하기 위해 실시간 USB 트래픽 분석기를 구현하는 것을 제안했다[13].

기존의 데이터 은닉은 파일 시스템 구조상으로 낭비되는 영역 및 예약된 영역에 데이터를 숨기는 기법이다. 은닉 영역을 생성하여 데이터를 은닉하는 경우에는 기존의 파일 시스템으로 생성한 후에 할당된 드라이브 문자를 제거하여 윈도우 탐색기에서 보이지 않게 하거나 MBR의 파티션 엔트리의 일부 값을 수정하는 방법을 사용한다. 그러나 Bootice로 생성한 은닉 영역은 기존의 파일 시스템 구조와 상이하며 현재까지 이 영역에 대한 논문 및 연구가 존재하지 않았기 때문에 이 영역을 처음 접할 경우 분석이 지연될 가능성이 존재한다. 또한 은닉 파티션에 대한 MBR의 파티션 엔트리를 제거했기 때문에 디지털 포렌식 도구로부터 잘 탐지되지 않아 해당 파티션을 발견하는 데에도 시간이 지연될 가능성이 존재한다.

본 논문의 연구 결과를 통해 디지털 포렌식 조사 과정에서 은닉 영역 분석과 은닉 영역 내부에 존재하는 데이터 추출에 기여를 할 것으로 기대된다.

Ⅲ. Bootice의 은닉 영역

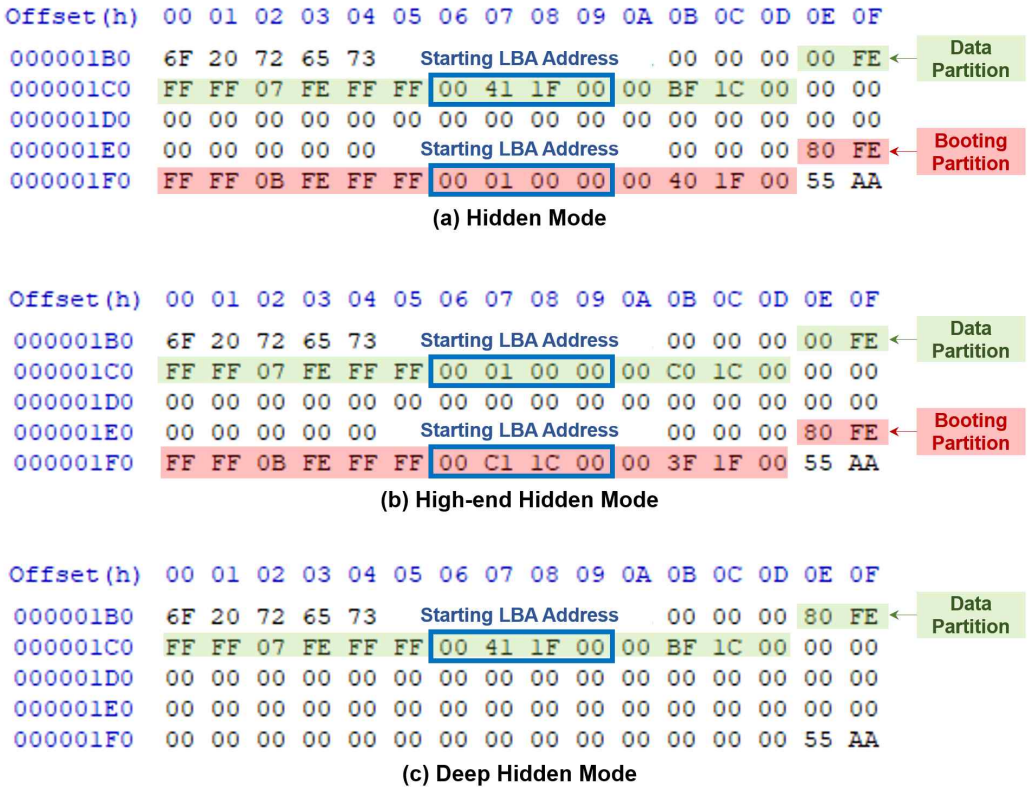
3.1 은닉 영역 종류와 특징

Bootice(ver. 1.3.4)는 저장 매체의 MBR 및 PBR에 설치, 백업, 복원이 가능하며 저장 매체의 파티션을 관리 및 포맷이 가능한 무료 도구다. 이 도구의 파티션 관리 기능 중 하나인 'U+ V2(2)' 기능을 사용하면 파티션을 부팅 파티션과 데이터 파티션 두 개로 나눠서 생성하고 이 중 부팅 파티션을 Hidden, High-end Hidden, Deep Hidden 3가지 모드로 은닉할 수 있다. 각 Hidden 모드는 MBR의 파티션 엔트리에 부팅 파티션과 데이터 파티션을 기록하는 방식이 상이하며 전체적인 구조는 〔그림 1〕과 같다.

Hidden 모드는 〔그림 1〕의 (a)와 같이 첫 번째 파티션 엔트리에 데이터 파티션의 정보를 기록하고 네 번째 파티션 엔트리에 부팅 파티션의 정보를 기록한다. 특이점은 첫 번째 파티션 엔트리에 있는 데이터 파티션의 시작 위치는 0x1F4100(2048256) 섹터고 네 번째 파티션 엔트리에 있는 부팅 파티션의 시작 위치는 0x100(256) 섹터로, 네 번째 파티션 엔트리에 기록된 부팅 파티션이 첫 번째 파티션 엔트리에 기록된 데이터 파티션보다 파일 시스템의 앞부분에 위치한다. 정상적인 파티셔닝과는 달리 파티션 엔트리가 순서대로 기록되지 않지만 두 파티션 모두 MBR의 파티션 엔트리에 기록되므로 실질적으로는 은닉되지 않기 때문에 윈도우 운영체제의 디스크 관리자를 통해 두 파티션 모두 접근 가능하다.

High-end Hidden 모드는 〔그림 1〕의 (b)와 같이 MBR의 파티션 엔트리에 각 파티션 정보를 기록하며 Hidden 모드와는 다르게 데이터 파티션의 시작 위치가 0x100(256) 섹터, 부팅 파티션의 시작 위치는 0x1CC100(1884416) 섹터로 첫 번째 파티션 엔트리에 기록된 데이터 파티션이 네 번째 파티션 엔트리에 기록된 부팅 파티션보다 파일 시스템의 앞에 위치한다. 하지만 이것도 각 파티션의 정보가 MBR에 저장돼있기 때문에 실질적으로 은닉되지 않으며 윈도우 운영체제의 디스크 관리자를 통해 두 파티션 모두 접근 가능하다.

마지막으로 Deep Hidden 모드는 〔그림 1〕의 (c)와 같이 데이터 파티션만 MBR의 파티션 엔트리에 기록하여 그 위치만 0x1F4100(2048256) 섹터임을 알 수 있고, 부팅 파티션의 파티션 정보는 기록하지 않아 이것을 성공적으로 은닉한다.

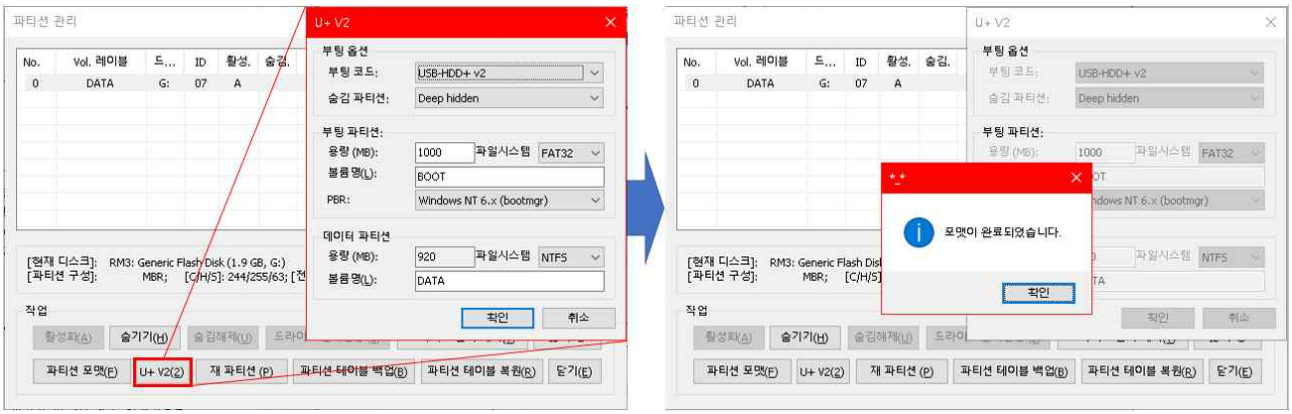


〈Figure 1〉 Current status of partition entry of MBR according to Hidden, High-end Hidden, and Deep Hidden modes

3.2 Deep Hidden 은닉 영역 생성

우리는 Bootice의 Deep Hidden 모드로 생성한 은닉 영역을 분석하기 위해 2GB 크기의 USB에 'U+V2(2)' 기능의 Deep Hidden 모드를 적용하여 연구를 진행했다. 〔그림 2〕 은 부팅 파티션의 크기는 1,000MB, 파일 시스템의 종류는 FAT32, PBR을 Windows NT 6.x bootmgr로, 볼륨명을 BOOT로 지정하고 데이터 파티션의 크기는 920MB, 파일 시스템의 종류는 NTFS로 지정하여 은닉 파티션을 생성하는 과정이다. 이 기능을 사용하면 앞에서 살펴본 것과 같이 1,000MB 크기의 부팅 파티션을 은닉할 수 있다.

우리는 윈도우의 디스크 관리 기능과 탐색기 그리고 EnCase, FTK Imager, Autopsy, X-ways Forensics와 같은 디지털 포렌식 도구들을 사용하여 이 은닉 영역의 탐지 여부를 확인했고 그 결과는 〔표 1〕과 같다. 〔그림 3〕은 〔표 1〕의 표기된 4가지의 디지털 포렌식 도구들을 통해 은닉 영역을 확인한 결과다. 오직 FTK Imager만 1,000MB 크기의 은닉된 부팅 파티션을 탐지했으며 나머지 도구들은 탐지하지 못했다. 특히 EnCase의 경우 내장된 Partition Finder 기능을 통해서도 은닉된 파티션 탐지에 실패했다.



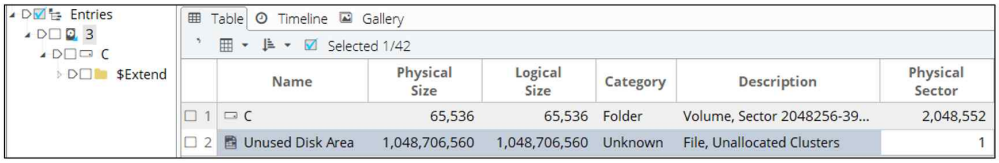
〈Figure 2〉 The process of creating a hidden area(Booting Partition) using Bootice's Deep Hidden mode.

〈Table 2〉 Result of detecting the hidden area created by Bootice using Windows programs and digital forensic tool.

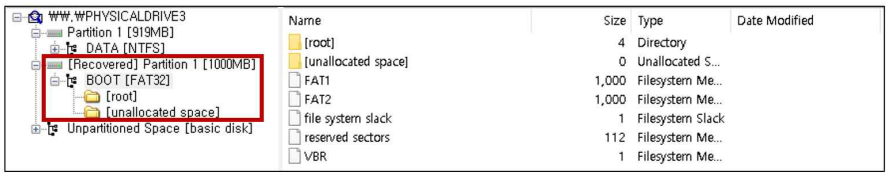
	Windows Tools		Digital Forensics Tools			
Tools	Disk Management	Windows Explorer	EnCase	FTK Imager	Autopsy	X-ways Forensics
Detect	x	x	x	o	x	x

3.3 Deep Hidden 은닉 영역 분석

Bootice로 생성한 은닉영역은 0x00(0)번 섹터부터 0x3E(62)번 섹터까지 0x1C-0x1D, 0x1BC, 파티션 엔트리의 0x1C6-0x1C9를 제외한 63개의 MBR이 반복된다. 0x3F(63)번 섹터부터 0x5F(95)번 섹터까지는 0x0E, 0x1BC, 파티션 엔트리의 0x1C6-0x1C9를 제외한 23개의 FAT32 VBR이 반복된다. 그리고 0x60(96)번 섹터부터 0x62(98)번 섹터까지는 Bootice에 내장된 커스텀 부트코드가 존재한다. 0x63(99)번부터 0xF8(248)번 섹터까지 총 150개의 FAT32 VBR이 다시 반복된다. 249번 섹터부터 251번 섹터까지는 총 3개의 Bootice에 내장된 커스텀 부트코드가 존재한다. 0x100(256)번 섹터부터 0x1F40FF(2048255)번 섹터까지는 은닉 영역의 파일 시스템이 존재하고 0x1F4100(2048256)번 섹터부터는 정상적인 NTFS 파일 시스템이 존재한다. 〔그림 4〕은 Bootice로 생성한 은닉 영역의 구조를 그림으로 표현했다.

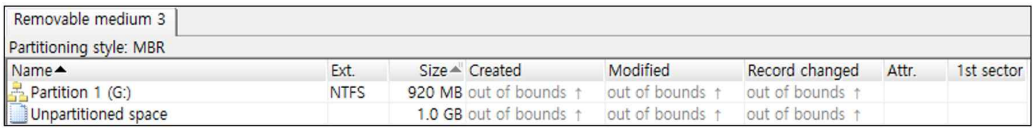


(a) EnCase



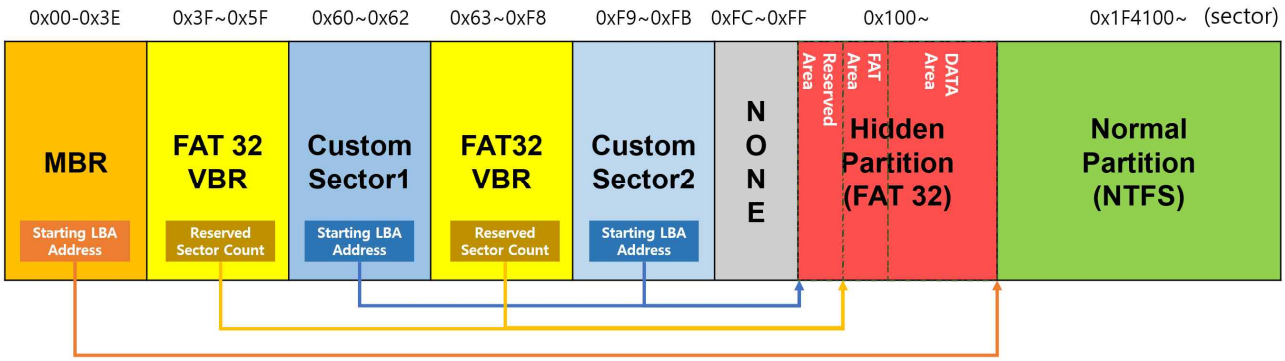
(b) FTK Imager

(c) Autopsy



(d) X-ways Forensics

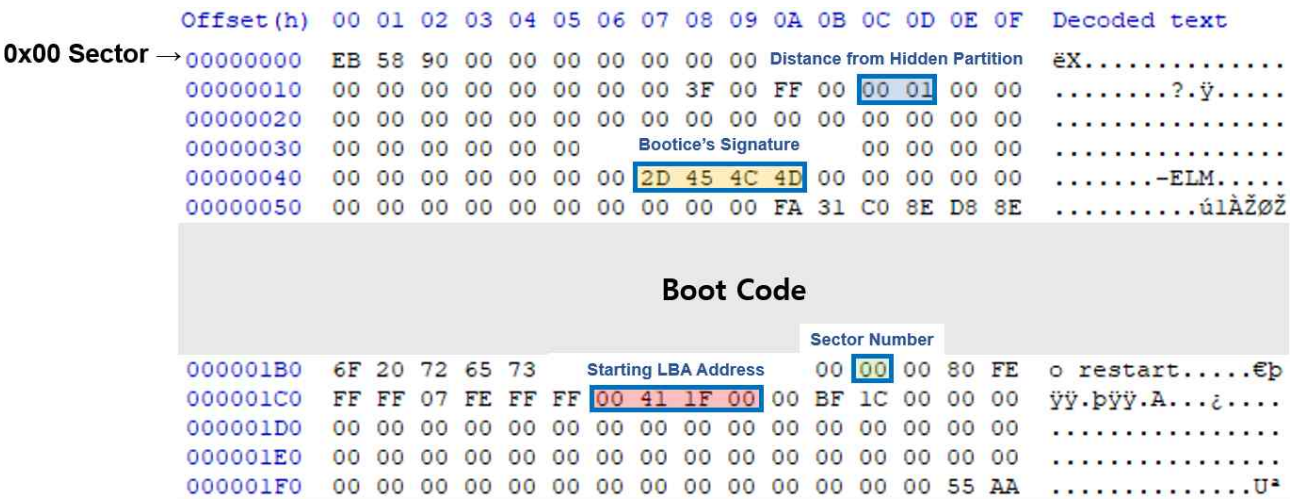
〈Figure 3〉 Results of attempting to detect hidden areas made of Bootice using digital forensics tools.



〈Figure 4〉 File system structure of hidden area created in deep hidden mode of Bootice.

3.3.1 MBR 영역

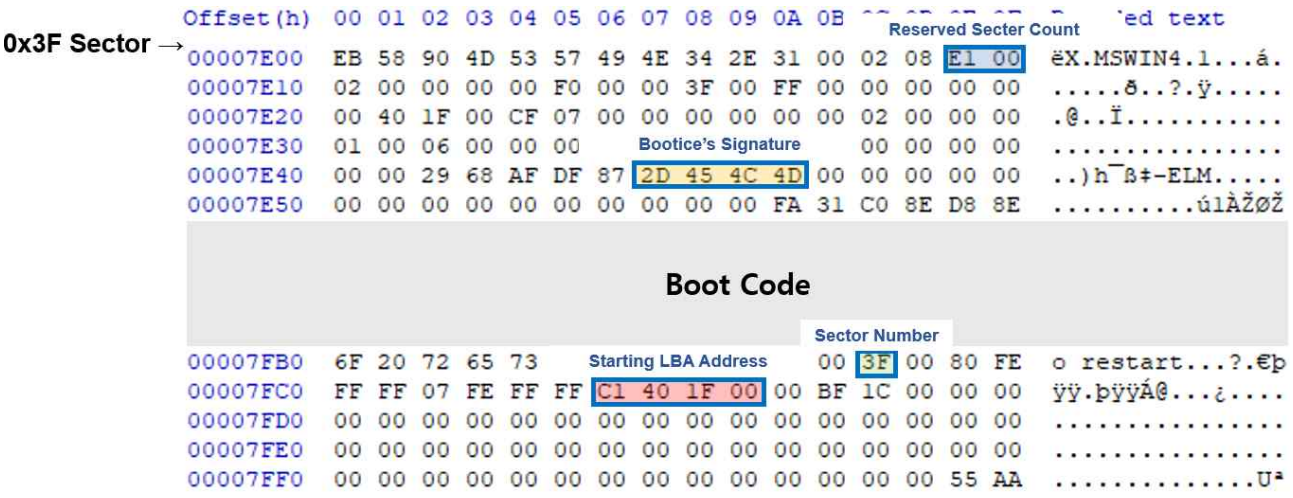
MBR 영역은 0x00(0)번 섹터부터 0x3E(62)번 섹터까지 0x1C-0x1D, 0x1BC, 0x1C6-0x1C9를 제외한 모든 부분이 동일하게 반복 저장돼 있다. 0x1C-0x1D는 해당 섹터부터 은닉 영역의 파일 시스템까지 거리다. 0x47-0x4A는 Bootice로 생성한 은닉영역의 시그니처이다. 0x1BC는 섹터 번호의 16진수 값이다. 0x1C6-0x1C9는 파티션 엔트리에서 은닉되지 않은 정상 파티션(데이터 파티션)의 위치를 섹터 단위로 표시한 값이다. 0x1C-0x1D와 0x1C6-0x1C9 값은 섹터 번호와 반비례하고 0x1BC는 비례한다. 〔그림 5〕는 Bootice로 생성한 은닉 영역의 MBR 영역이다.



〈Figure 5〉 MBR Area

3.3.2 FAT32 VBR 영역

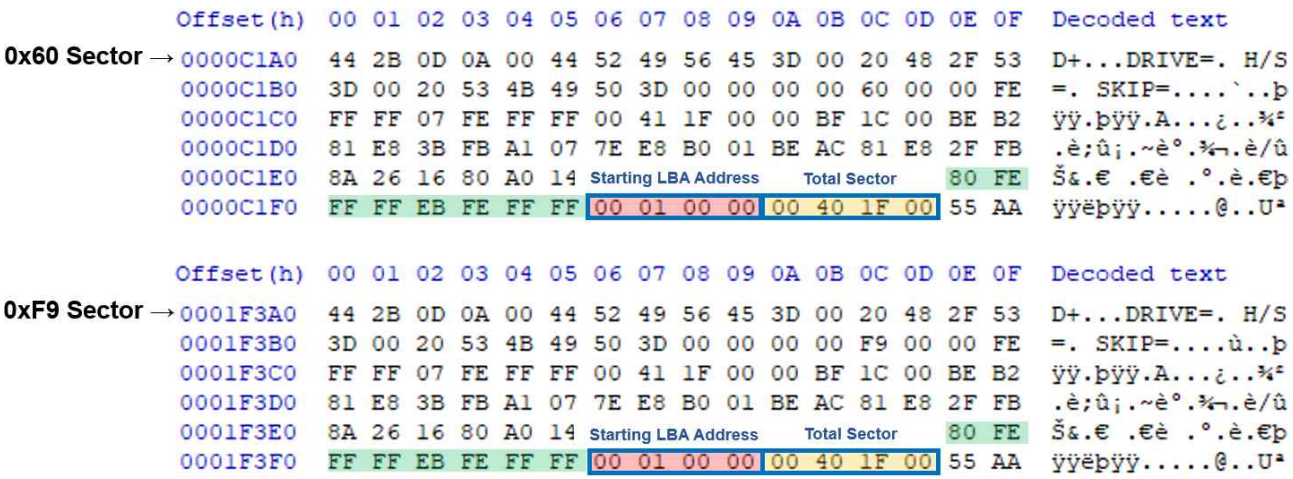
FAT32 VBR 영역에는 은닉된 파티션인 부팅 파티션에 대한 VBR의 BPB(BIOS Parmameter Block) 정보가 존재하고 파티션 엔트리에는 은닉되지 않은 데이터 파티션 정보만 저장돼 있다. 0x3F(63)번 섹터부터 0x5F(95)번 섹터와 0x63(99)번 섹터부터 0xF8(248)번 섹터까지 0x0E-0x0F, 0x1BC, 0x1C6-0x1C9를 제외한 모든 부분이 동일하게 반복 저장돼 있다. 0x7E0E-0x7E0F는 FAT 파일 시스템의 Reserved Area의 섹터 값이며 이 값을 통해 은닉된 파티션의 FAT Area의 시작 위치를 계산할 수 있다. 또한, FAT Area의 크기(0x7E24-0x7E27)와 개수(0x7E10)의 값을 통해 Data Area의 시작 위치도 계산할 수 있다. 0x7E47-0x7E4A는 Bootice로 생성한 은닉 영역의 시그니처이다. 0x7FBC는 해당 섹터 번호를 16진수로 저장한 것이며 값이며 0x7FC6-0x7FC9는 은닉되지 않은 정상 파티션(데이터 파티션) 위치를 섹터 단위로 표시한 값이다. 그리고 0x0E-0x0F 값과 0x1C6-0x1C9 값은 섹터 번호와 반비례하고 0x1BC 값은 비례한다. 〔그림 6〕 Bootice로 생성한 은닉 영역의 FAT32 VBR 영역이다.



〈Figure 6〉 FAT32 VBR Area

3.3.3 Custom Sector 영역

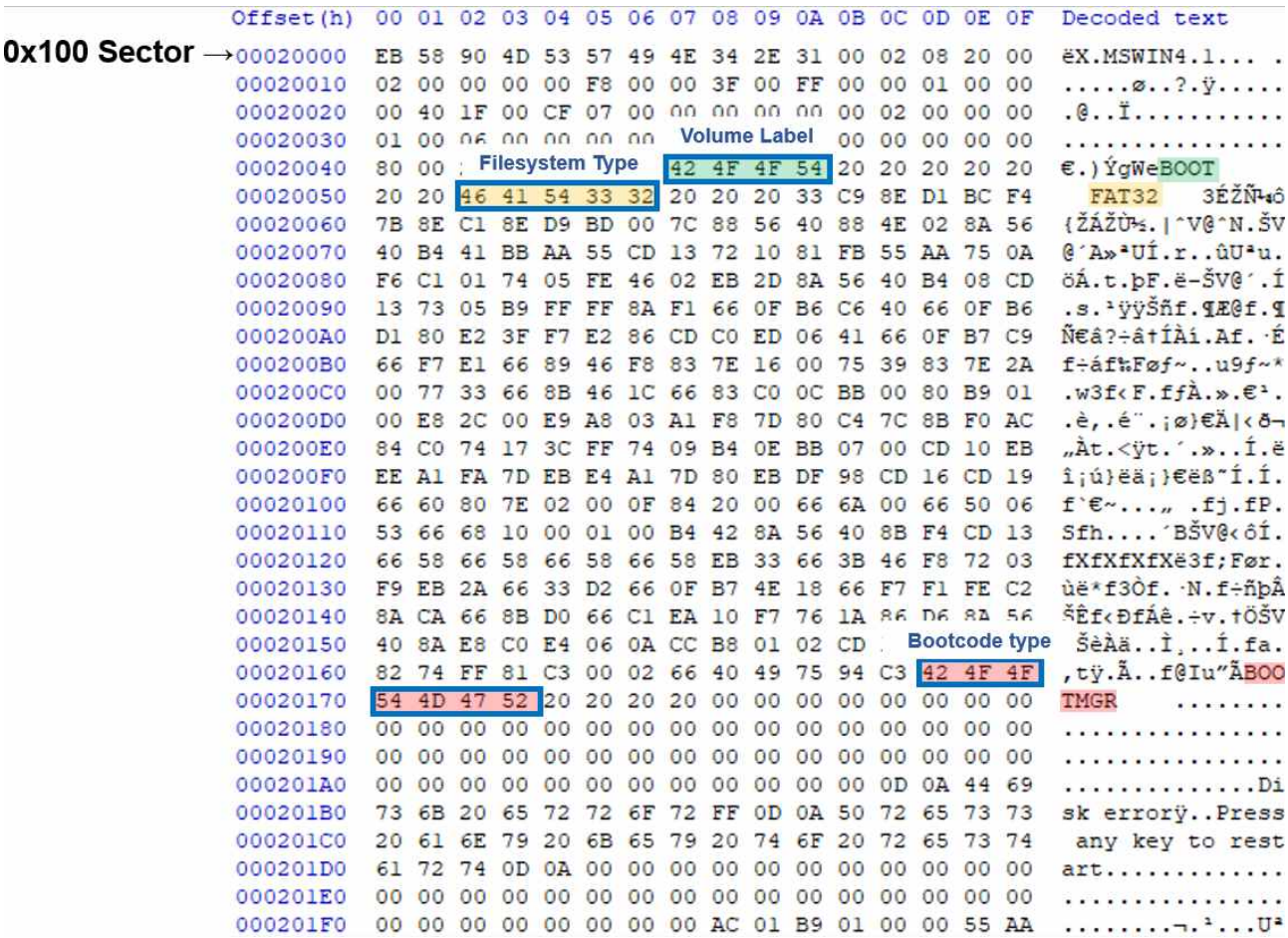
Custom Sector 영역에는 은닉된 파티션인 부트 파티션의 위치를 계산하기 위한 커스텀 부트코드와 부트 파티션에 대한 파티션 테이블 정보가 저장돼 있다. 0x60(96)번 섹터부터 0x62(98)번 섹터까지 그리고 0xF9(249)번 섹터부터 0xFB(251)번 섹터까지는 Bootice에 내장된 커스텀 부트코드가 존재하는 영역이다. 96번 섹터의 경우 0xC1EE부터 0xC1FD까지, 249번 섹터의 경우 0x1F3EE부터 0x1F3FD까지 은닉 영역의 파티션 정보(위치, 크기)가 존재하며 구조는 일반적인 MBR의 파티션 엔트리와 동일하다. 〔그림 6〕은 0x60(96)번 섹터와 0xF9(249)번 섹터를 확인한 결과이다. 은닉된 영역의 위치는 0x100(256)번 섹터이며 크기는 0x1F4000(2048000)개 섹터다. 〔그림 7〕은 Bootice로 생성한 은닉 영역의 커스텀 섹터 영역이다.



〈Figure 7〉 Custom Sector Area

3.3.4 Hidden Partition 영역

이 영역은 은닉된 부트 파티션이 저장돼 있다. 0x100(256)번 섹터에는 기존의 FAT32 VBR이 존재하며 일반적인 VBR과 동일하다. 0x100(256)번 섹터부터 0x1F40FF(2048255)번 섹터에는 은닉 영역의 파일 시스템이 존재한다. 이 영역과 0x3F(63)번 섹터의 FAT32 VBR과의 차이점은 파티션 엔트리 영역이 존재하지 않는다는 점이다. 〔그림 8〕은 0x100(256)번 섹터를 확인한 결과이며 〔그림 2〕에서 사용자가 지정한 부팅 파티션 값이 반영된 것을 확인할 수 있다.



〈Figure 8〉 Hidden Partition Area

Ⅳ. Bootice의 은닉 영역을 사용한 안티포렌식 기법

Bootice의 은닉 영역은 앞에서 살펴본 것과 같이 FTK Imager에서만 정상적으로 탐지하고 그 외의 디지털 포렌식 도구들은 탐지하지 못하기 때문에 안티포렌식 기법으로 사용될 수 있다. 하지만 Bootice는 은닉 영역을 생성만 하고 그 영역에 접근하기 위한 인터페이스는 제공하지 않기 때문에 이 영역을 안티포렌식을 위해 사용하기 위해서는 안티포렌식 행위자가 파일 시스템 도구, Hex Editor 등의 도구를 사용하여 메뉴얼한 방법으로 은닉 영역에 데이터를 저장해야 한다.

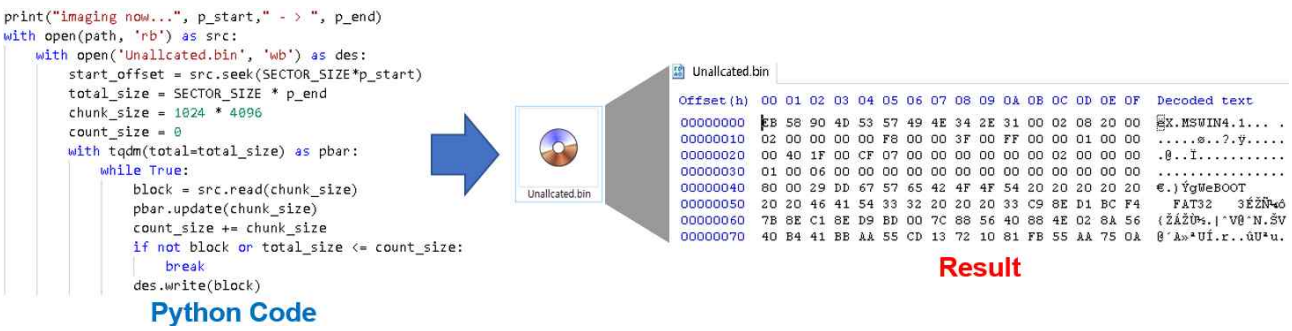
따라서 우리는 이 은닉 영역에 쉽게 파일을 저장할 수 있는 안티포렌식 기법을 개발했다. 우선 USB에서 이 은닉 영역의 파일 시스템을 추출하고 마운팅한다. 그 후 윈도우의 탐색기를 통해 쉽게 이 은닉된 영역에 파일들을 저장한 후 USB에 이 수정된 파티션 영역을 적용한다. 또한, Bootice의 Deep Hidden 기능으로 생성한 은닉 영역을 FTK Imager 도구에서도 탐지하지 못하도록 탐지 회피 기법을 개발했다.

이 방법들을 사용하면 Bootice의 Deep Hidden 기능으로 기존의 모든 디지털 포렌식 도구에서 탐지를 우회하는 은닉 영역을 생성하고 은닉된 영역에 파일을 쉽게 저장하는 안티포렌식을 수행할 수 있다.

4.1 은닉 영역에 파일 저장 기법

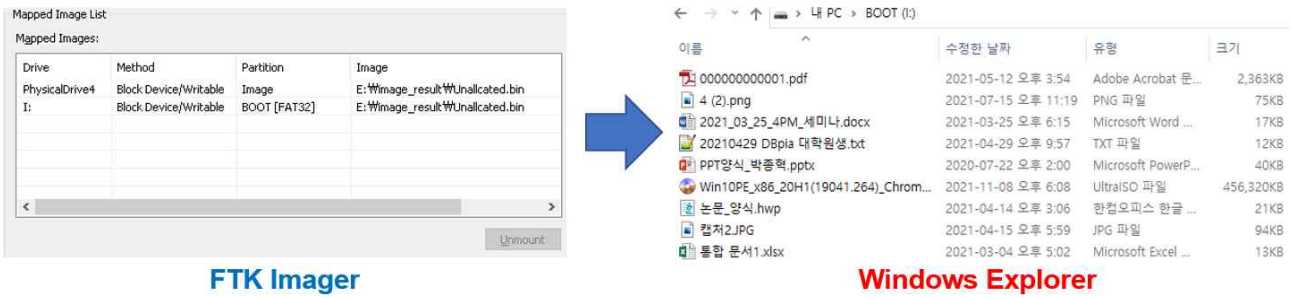
우리는 은닉 영역의 파일 시스템에 파일을 저장하기 위해 파이썬 코드를 이용하여 0x100(256)번 섹터부터 0x1F40FF(2048255)번 섹터까지 추출하고 FTK Imager의 image mounting 기능을 사용해 추출한 파일 시스템을 마운트한다. 그 후 윈도우 탐색기를 통해 파일을 저장하고 FTK Imager를 통해 이미징 한 다음 파이썬 코드를 이용하여 다시 0x100(256)번 섹터부터 0x1F40FF(2048255)번 섹터까지 덮어쓴다.

Bootice를 사용해 생성한 은닉 영역의 파일 시스템은 0x100(256)번 섹터부터 0x1F40FF(2048255)번 섹터까지다. 〔그림 9〕 해당 영역을 추출하기 위해 파이썬의 pytsk 라이브러리를 이용하여 부팅 파티션을 탐지하고 시작섹터를 0x100(256)번 섹터로 지정하여 일정한 크기만큼 추출하는 코드의 일부와 코드 실행 결과이다.



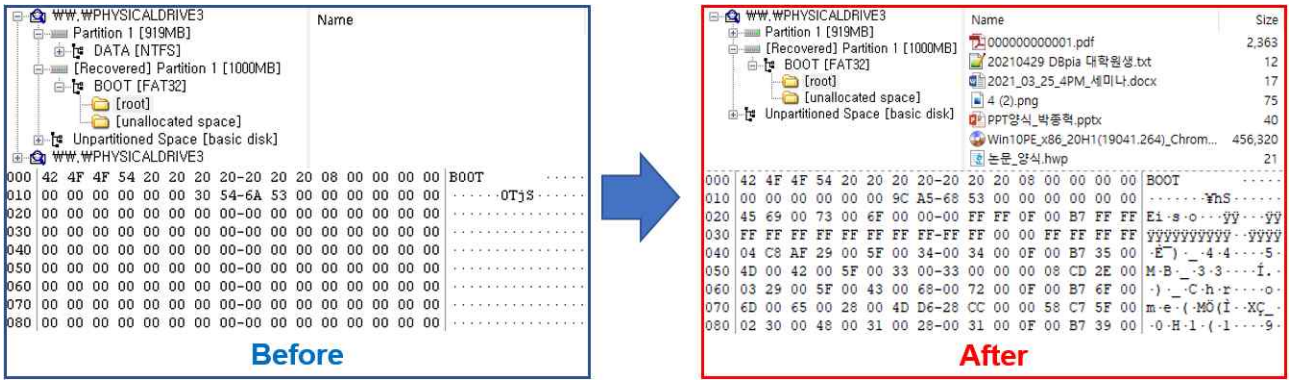
〈Figure 9〉 Python code that extracts hidden areas and the result of executing that code

〔그림 10〕은 파이썬 코드를 통해 추출한 은닉 영역의 바이너리를 FTK Imager의 image mounting 기능을 통해 마운트한 후에 윈도우 탐색기를 통해 테스트 파일을 마운트 영역에 저장했다. 그러나 FTK Imager로 마운트한 영역은 변경사항을 저장하지 않기 때문에 변경 사항이 저장된 이미지 파일을 생성한다.



〈Figure 10〉 Mount the extracted image file using FTK Imager and store the test file in the mount area

〔그림 9〕의 파이썬 코드와 거의 동일한 코드를 사용하여 USB에 변경사항이 저장된 이미지 파일의 내용을 0x100(256)번 섹터부터 0x1F40FF(2048255)섹터까지 덮어쓴다. 〔그림 11〕은 변경사항이 저장된 이미지 파일을 덮어쓰기 전과 후를 FTK Imager를 통해 확인한 결과이다.

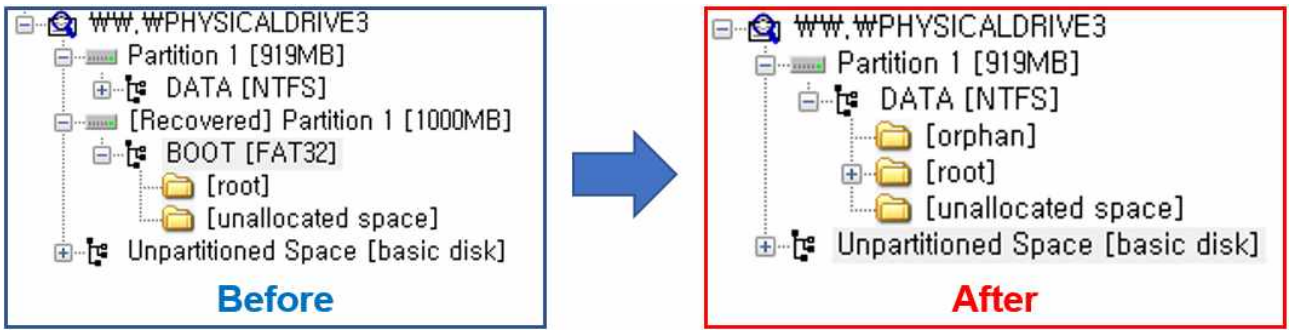


〈Figure 11〉 Overwrite the extracted image file to the USB flash drive

4.2 은닉 영역 탐지 회피 기법

앞서 Bootice를 이용해 생성한 은닉 영역은 FTK Imager를 통해 탐지 및 복구가 가능한 것을 확인했다. 그래서 우리는 FTK Imager가 은닉 영역의 탐지를 방해하기 위해 파이썬 코드를 통해 은닉 영역의 MBR값 일부를 변경하여 탐지를 회피하는 방법을 개발했다.

FTK Imager는 Bootice로 생성한 은닉 영역을 탐지할 때 은닉 영역에 0x3F(63)번 섹터를 인식하여 파티션 복구하는 것으로 추정된다. 그래서 우리는 0x3F(63)번 섹터부터 0x5F(95)번 섹터까지 차례로 BPB(BIOS Parameter Block)의 일부 영역을 수정하는 실험을 통해 0x3F(63)번과 0x40(64)번 섹터의 BPB(BIOS Parameter Block)영역에 0x0B~0x0C 값을 0x00으로 수정할 경우 FTK Imager는 은닉 영역의 파일 시스템을 인식하지 못하는 것을 발견했다. 〔그림 12〕은 0x0B~0x0C 값을 0x00으로 수정하기 전과 후를 FTK Imager를 통해 탐지한 결과이다.



〈Figure 12〉 The result of detecting the hidden area in the FTK Imager when the value of 0x0B to 0x0C is modified to 0x00.

Ⅴ. Bootice의 은닉 영역 탐지 방법

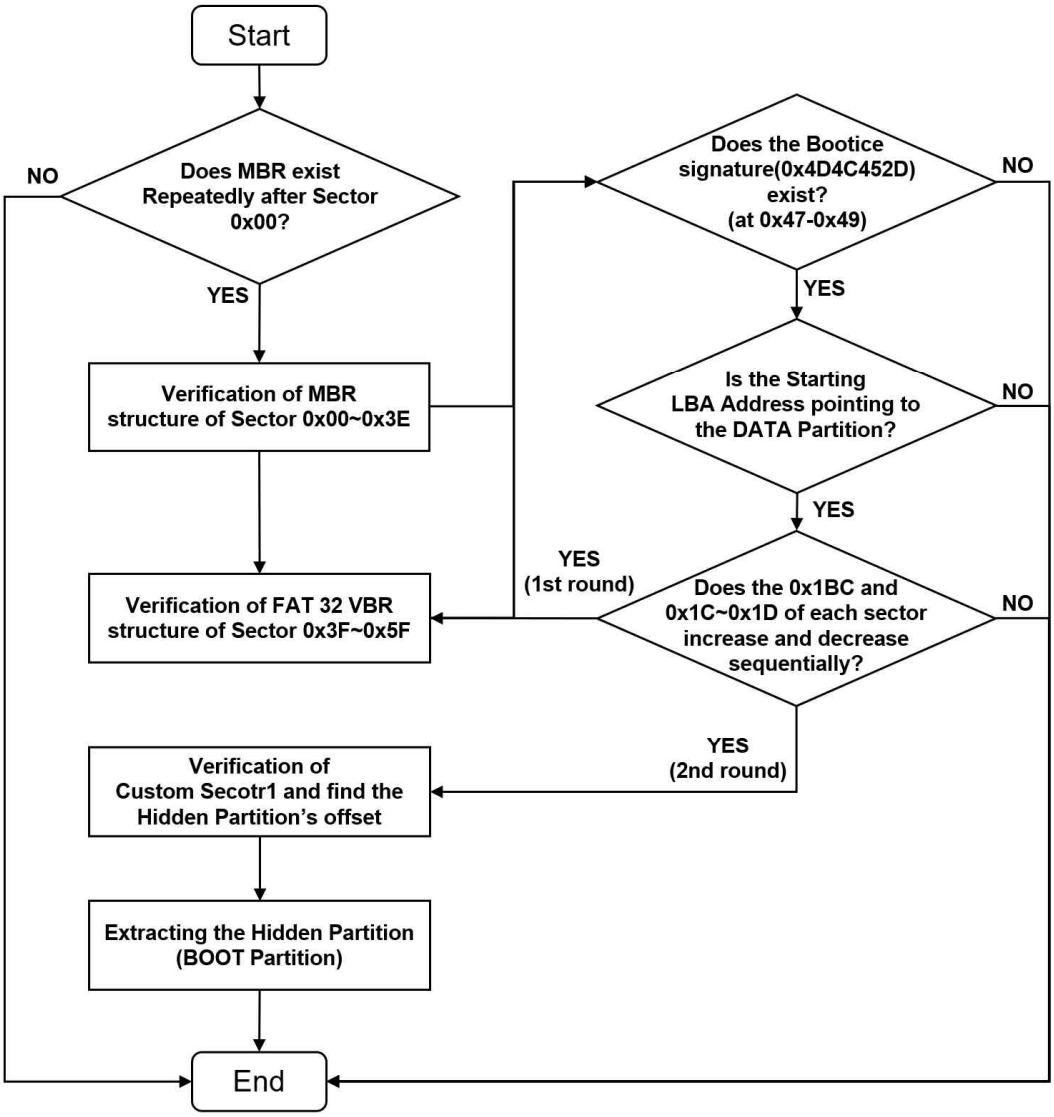
우리는 Bootice로 생성한 은닉 영역을 분석한 결과를 토대로 〔그림 13〕과 같은 은닉 영역 탐지 방안을 제안한다. 우선, 0x00(0)번 섹터 이후에 MBR 영역이 반복적으로 존재하는 경우 은닉영역 생성 도구를 사용 여부를 의심해야 한다. 은닉 영역을 생성하는 또 다른 도구인 FbinstTool도 MBR 영역이 반복적으로 존재한다〔14〕.

그 후 반복되는 MBR 영역의 구조체에서 3가지 값을 확인한다. 우선 0x47-0x4A에서 Bootice의 시그니처(0x4D4C452D)의 존재 유무를 확인한다. 그리고 파티션 엔트리의 Starting LBA Address에서 정상적인 DATA 파티션의 위치를 확인한다. 마지막으로 0x1BC에 저장된 해당 파티션의 위치 값이 각 섹터마다 1씩 순차적으로 증가하는지, 0x1C-0x1D에 저장된 은닉된 파티션의 상대 위치 값이 각 섹터마다 1씩 순차적으로 감소하는지 확인한다.

이후 0x3F-0x5F 위치에 반복적으로 존재하는 FAT32 파일 시스템으로 구성된 은닉된 파티션의 VBR 구조체에서도 3가지 값을 확인한다. 우선 MBR과 마찬가지로 0x47-0x4A에서 Bootice의 시그니처(0x4D4C452D)의 존재 유무를 확인하고 파티션 엔트리의 Starting LBA Address에서 정상적인 DATA 파티션의 위치를 확인한다. 그리고 이 위치 값이 각 섹터마다 1씩 감소하는지 확인한다. 마지막으로 0x0E-0x0F에서 FAT32의 Reserved Area 영역의 섹터 수를 확인함으로써 은닉된 파티션의 존재 유무를 확실하게 확인한다.

위의 2가지 과정을 확인함으로써 USB에 Bootice의 은닉 영역이 존재함을 알 수 있다. 따라서 Custom Sector1에 존재하는 은닉 영역의 숨겨진 파티션 엔트리의 값을 확인하여 은닉된 파티션을 추출한다. 이러한 과정을 통해 Bootice의 은닉 영역을 탐지할 수 있다.

또한, 반복적인 실험을 한 결과 은닉된 파티션은 무조건 0x100(256)번 섹터에 생성됨을 알 수 있었다. 따라서 MBR과 정상적인 파티션의 사이인 0x100번 섹터에서 FAT32의 VBR이 발견되는 경우 Bootice의 은닉 파티션이 존재함을 쉽게 탐지할 수 있다.



〈Figure 13〉 Hidden area detection algorithm created with Bootice

Ⅵ. 결 론

본 논문에서는 Bootice로 생성한 은닉 영역을 분석하여 구조를 파악하고 분석한 결과를 토대로 은닉 영역에 데이터를 은닉하는 방법을 연구했다. 그리고 은닉 영역을 디지털 포렌식 도구의 탐지로부터 회피하는 방안을 연구했다. 그 결과, 은닉 영역의 구조를 파악하여 은닉 파티션의 위치 및 크기에 대한 정보를 얻을 수 있었으며 이 파티션을 추출하여 마운트한 후 원하는 데이터를 저장하고 다시 은닉 파티션의 위치에 덮어쓰는 방식으로 은닉 영역에 데이터를 저장할 수 있었다. 그리고 디지털 포렌식 도구의 탐지로부터 회피하기 위해 은닉 영역의 0x3F(63)번 섹터와 0x40(64)번 섹터의 BPB 영역의 일부 값을 수정하여 탐지로부터 회피할 수 있었다. Bootice로 생성한 은닉 영역은 FAT과 같은 알려진 파일 시스템을 사용하였으나 기존에 알려진 MBR, VBR 구조와 다르기 때문에 기존의 조사 방식으로는 이 은닉 영역을 발견하지 못하거나 발견을 하더라도 분석에 어려움이 있을 것으로 예상된다.

본 연구 결과는 디지털 포렌식 조사 과정을 늦추거나 방해하는 목적으로 설계된 안티포렌식 도구 및 기법 연구에 대해 기여할 것으로 기대된다. 그러나 본 논문은 Bootice로 생성한 은닉 영역에만 해당되며 이 영역을 이용한 안티포렌식 행위는 디지털 포렌식에 대한 지식이 없는 사용자는 구현하기 어렵다는 단점이 존재한다. 향후 우리는 USB 메모리에 은닉 영역을 생성하는 또 다른 도구들을 조사하고 이것들을 탐지하는 디지털 포렌식 도구를 개발할 예정이다.

참 고 문 헌 (References)

〔1〕 Bootice 1.3.4 version〔internet〕 Available : https://bootice.softonic.kr/

〔2〕 M. Gül and E. Kugu, "A survey on anti-forensics techniques," 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), pp. 1-6, 2017.

〔3〕 T. R. Etow, "IMPACT OF ANTI-FORENSICS TECHNIQUES ON DIGITAL FORENSICS INVESTIGATION," Dissertation, 2020.

〔4〕 Jean-Paul A. Yaacoub, Hassan N. Noura, Ola Salman, Ali Chehab, "Digital Forensics vs. Anti-Digital Forensics: Techniques, Limitations and Recommendations.", CoRR, abs/2103.17028, 2021.

〔5〕 H. Majed, H. N. Noura and A. Chehab, "Overview of Digital Forensics and Anti-Forensics Techniques," 2020 8th International Symposium on Digital Forensics and Security (ISDFS), pp. 1-5, 2020.

〔6〕 Mohamad Ahtisham Wani, Ali AlZahrani, Wasim Ahmad Bhat, "File system anti-forensics - types, techniques and tools", Computer Fraud & Security, Volume 2020, Issue 3, Pages 14-19, 2020.

〔7〕 Jewan Bang, Byeongyeong Yoo, Sangjin Lee, "Secure USB bypassing tool", Digital Investigation, Volume 7, Supplement, pp. S114-S120, 2010.

〔8〕 Jaein Kim, Youngjun Lee, Kyungroul Lee, Taeyoung Jung, Dmitry Volokhov, and Kangbin Yim, "Vulnerability to Flash Controller for Secure USB Drives", J. Internet Serv. Inf. Secur., 3(3/4), pp. 136-145, 2013.

〔9〕 Stéphanie Blanchet, "BadUSB, the threat hidden in ordinary objects.", 2018

〔10〕 So-Hee Kim, Jaehyok Han, Sangjin Lee, "A Study on Detecting Data Hiding Area of Removable Storage Device Based on Flash Memory", Digital Forensics Research, 12(2), pp. 21-29, 2018

〔11〕 Nir Nissim, Ran Yahalom, Yuval Elovici, "USB-based attacks", Computers &Security, Volume 70, pp. 675-688, 2017.

〔12〕 Tyler Thomas, Mathew Piscitelli, Bhavik Ashok Nahar, Ibrahim Baggili, "Duck Hunt: Memory forensics of USB attack platforms", Forensic Science International: Digital Investigation, Volume 37, Supplement, 301190, 2021.

〔13〕 Marian, T. I. C. U. "Raising awareness of cyber security concerns regarding the use of USB peripherals.", Romanian Cyber Security Journal, No. 1, vol. 1, Spring 2019

〔14〕 Pyo-Gil Hong, Dohyun Kim, "A study on counter anti-forensics for hidden areas of removable media", Journal of digital froensics, vol.15, no.3, pp.72-84, 2021.

저 자 소 개

홍 표 길 (Pyogil Hong)
준회원

2021년 2월 : 부산가톨릭대학교 졸업
2021년 3월~현재 : 부산가톨릭대학교 일반대학원 컴퓨터공학과 석사과정
관심분야 : 디지털 포렌식, 취약점 분석 등

김 도 현 (Dohyun Kim)
정회원

2019년 8월 : 고려대학교 정보보호대학원 공학박사
2017년 7월 ~ 2019년 8월 : 한국전자통신연구원 지능화융합연구소 정보보호연구본부 연구원
2019년 9월 ~ 2020년 3월 : 고려대학교 정보보호연구원 디지털포렌식연구센터 연구교수
2020년 4월 ~ 현재 : 부산가톨릭대학교 컴퓨터공학과 조교수
2020년 7월 ~ 현재 : 부산가톨릭대학교 융합보안공학센터 센터장
관심분야 : 디지털 포렌식, 취약점 분석, 사이버보안 등