Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.27, NO.4, Aug. 2017 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2017.27.4.753

외장형 USB 매체의 작업이력 점검 방법에 관한 연구

이 성 재,[1†] 노 봉 남[2‡]
[1]ETRI 부설연구소, [2]전남대학교

A Study of Checking the Job History of External USB Media

Seongjae Lee,[1†] Bongnam Noh[2‡]
[1]The Attached Institute of ETRI, [2]Chonnam National University

요 약

최근 각 분야에서 외장형 USB 매체를 활용한 악성코드 감염 및 비밀문건의 유출 사고가 빈번하게 발생하고 있다. 외장형 USB 매체를 이용해서 발생한 사건들을 조사하기 위해서 매체들에 대한 수사를 진행하지만, 분실 및 훼손이 발생할 수 있다는 점에서 많은 어려움이 있다. 결국 외장형 USB 매체에 대한 사건 조사를 위해서는 외장형 USB 매체뿐만 아니라 해당 매체가 연결되었던 시스템에 대한 직접적인 분석을 진행해야만 한다.

본 논문은 외장형 USB 매체가 연결되었던 Windows 시스템의 아티팩트에 대한 분석으로 매체에서의 작업이력을 점검하는 방법을 제안한다. 이를 통해 외장형 USB 매체를 확보하지 못한 상황에서도 시스템에서 USB 매체를 통해서 수행되었던 작업이력을 분석하는데 활용될 수 있을 것으로 기대된다.

ABSTRACT

Recently, malicious code infiltration and leakage of confidential documents using external USB medium are frequently occurring in each field. We investigate the media to investigate incidents using external USB media, but there are many difficulties in that they can be lost or damaged. Ultimately, in order to investigate cases of external USB media, it is necessary to conduct a direct analysis of the external USB media as well as the system to which the media is connected.

This paper describes an analysis of the artifacts of Windows systems to which external USB media is connected, and how to check the job history on the media. Therefore, it is expected that the system can be used to analyze the job history of the USB medium even if the external USB medium is not secured.

Keywords: USB Storage, USB Media, USB Media Forensic

I. 서 론

외장형 USB 매체는 소형화 그리고 대용량화를 목적으로 꾸준히 발달해 왔으며, 편의성을 바탕으로 매우 높은 성장세를 보여 왔다[1]. 더불어 USB 매체가 제공하는 편리한 자료 보관 및 전달, 대중성은 산업전반에 걸쳐 업무 효율성의 증대를 가져왔다. 하지만, 이와 함께 USB 매체를 이용한 악성코드의 감염 및 기밀자료의 유출 등 부적정인 사례들이 끊임없이 발생하고 있다. USB 매체를 이용하여 발생되는 사건·사고를 조사하기 위해서 가장 먼저 선행되어야 할 작업은 해당 USB 매체의 특정과 확보이다. 이 과정에서 USB 매체 확보에 실패하거나 확보했더라도 매체에 대한 훼손이 발생할 수 있다는 점에서 아직까지 극복해야할 문제점이 있다. 따라서 USB 매체에 대한 분석과 함께 해당 매체가 연결되었던 시스템에 대한 분석이 요구된다.

본 논문에서는 USB 매체가 연결되었던 시스템에 매체의 작업이력과 관련된 아티팩트를 설명하고, 아티팩트들 간의 상관관계를 분석하여 추가적으로 작업

Received(03. 07. 2017), Modified(05. 30. 2017), Accepted(05. 30. 2017)
† 주저자, dreistar3@nsr.re.kr
‡ 교신저자, bbong@jnu.ac.kr(Corresponding author)

이력를 추적할 수 있는 방법에 대해서 제안한다.

논문의 구성은 2장에서 USB 매체를 점검하기 위해 참조될 수 있는 시스템 아티팩트에 대한 소개와 단일 아티팩트 분석으로 확인할 수 있는 작업이력을 서술한다. 3장에서는 아티팩트에서 획득한 정보들의 상관관계를 분석하여 추가적으로 작업이력을 점검하는 방안을 제안한다. 4장에서는 현재 논문이 갖는 한계와 이를 극복하기 위한 향후 연구방향에 대해서 서술한다.

II. USB 매체 점검을 위한 시스템 아티팩트

〔표 1〕은 USB 매체의 작업이력을 분석하기 위해서 참조될 수 있는 시스템 아티팩트와 이들에서 확인할 수 있는 정보를 나타낸다. 단일 아티팩트 분석으로 확인할 수 있는 USB 매체의 작업이력을 파악하고, 아티팩트들 간의 상관관계를 증명하기 위해서 참조될 수 있는 정보를 확인한다.

2.1 Windows Registry

USB 매체가 시스템에 연결되면 버스 드라이버는 PnP 관리자에게 제조사, 일련번호 등 매체의 고유 식별정보를 전달하여, 매체의 연결을 알린다. 커널 모드 PnP 관리자는 식별정보를 바탕으로 Device Class ID를 구성하고, 레지스트리에서 해당 ID 값에 부합하는 드라이버가 시스템에 설치되어 있는지를 확인한다. 드라이버가 설치되어 있지 않았다면 커널 모드 PnP 관리자는 장치 펌웨어에게 드라이버를 요청하고 전달받은 드라이버를 사용자 모드 PnP 관리자에게 전달한다. 사용자 모드 PnP 관리자는 전달받은 드라이버를 설치한다. 드라이버의 설치가 완료되면, USB 매체를 시스템에 마운트하고 관련 정보를 레지스트리와 이벤트 로그에 기록한다〔2〕. USB 매체가 시스템에 연결되어 변경되거나 혹은 새롭게 생성된 레지스트리 값은 매체의 작업이력을 점검하기 위해서 반드시 분석해야한다.

USB 매체의 작업이력을 점검하기에 앞서 시스템에 연결되었던 USB 매체에 대한 식별이 선행되어야 한다. HKLM\SYSTEM\ControlSet00x\Enum의 하위에 존재하는 USB 그리고 USBSTOR 레지스트리 키에는 지금까지 시스템에 연결되었던 USB 매체의 정보가 기록되어 있다. 해당 레지스트리 키의 하위에는 USB 매체에 대한 각각의 정보가 목록화되어 있어서, 제조사 및 모델명, 드라이버 정보, 매체의 종류, 장치 설명 정보, 장치식별 정보 등을 확인할 수 있다. 특히, CompatibleIDs, HardwareID, ClassGUID로 기록되는 장치식별 번호는 개별 USB 매체의 작업이력을 판별하기 위해서 필히 참조된다〔3〕.

다음으로 USB 매체가 시스템에 마운트 되어 할당 받은 드라이브 문자를 확인한다. 생성, 삭제, 수정 등의 작업이력 관련 데이터를 갖는 Link 파일, Jump 파일 및 Prefetch 파일 등에서는 대상 객체

Table 1. System artifacts for job history analysis of USB media

Analysis Target / Information		A_1 Event	A_2 Registry	A_3 Link · Jump	A_4 Browser History	A_5 Prefetch	A_6 IconCache
D_1	First connected time	○	○	-	-	-	-
D_2	Connected time	○	-	-	-	-	-
D_3	Released time	○	-	-	-	—	-
D_4	Last released time	○	○	-	-	-	-
D_5	Identification number	○	○	-	-	-	-
D_6	Mounted drive letter	-	○	-	-	-	-
D_7	Absolute path of file	-	-	○	○	-	○
D_8	File name	-	-	○	○	○	○
D_9	Time Info of the file	-	-	○	○	-	-
C_1	Data accumulation	○	○	-	○	○	○
C_2	Updating data	-	-	○	-	○	-
C_3	Elimination possibility	○	○	○	○	○	-

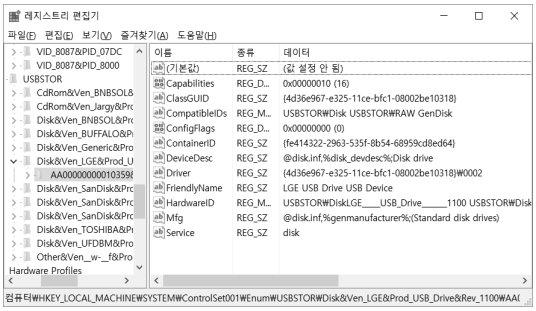

Fig. 1. Analysis about USB media identification and device identification number

의 절대경로와 함께 생성시간, 수정시간, 접근시간 등 다양한 정보를 갖는다. USB 매체가 삽입된 시간 동안 임의의 작업이력이 확인된 객체의 절대경로에서 USB 매체의 드라이브 문자가 포함되어 있는 경우, 해당 매체에의 작업이력으로 판단한다. USB 매체에 할당 드라이브 문자를 확인하기 위해서 확인하는 레지스트리를 참조하는 과정을 보인다. HKLM\SYSTEM\MountedDevices 레지스트리 키는 드라이브 문자를 기준으로 매체의 식별정보를 포함한다. 드라이브 문자에 매핑된 매체 식별정보는 현재 또는 가장 최근에 마운트 되었던 매체 정보만을 보여주기 때문에, 동일한 드라이브 문자를 마운트 받았던 이전의 USB 매체 정보는 확인할 수 없다.

HKLM\System\CurrentControlset\Enum\SWD\WPDBUSENUM 레지스트리 키는 휴대용 매체의 enumerator 서비스를 지원하기 위해서 참조되고 있으며, 원활한 서비스 지원을 위해서 매체의 정보를 포함하고 있다. 이들 중에서 FriendlyName 정보는 사용자 혹은 매체의 제조사가 지정한 볼륨 레이블이 기록된다. 하지만 임의의 설정이 이루어지지 않았을 경우에는 마운트 되어 할당받은 드라이

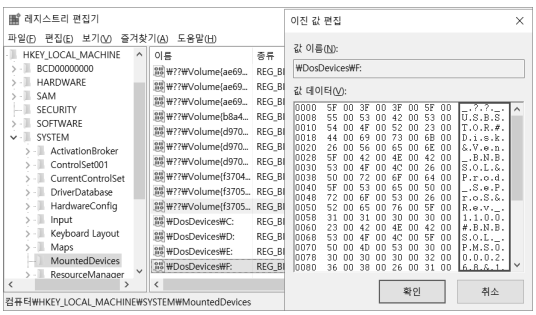

Fig. 2. Analysis about drive letter assigned to the USB media_(1)

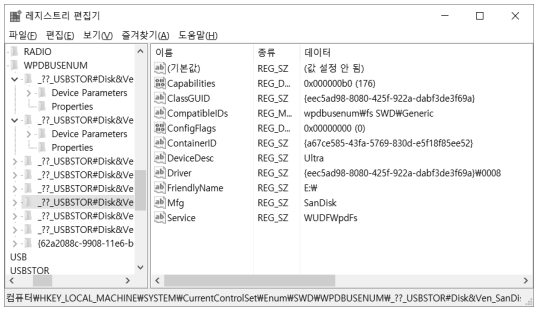

Fig. 3. Analysis about drive letter assigned to the USB media_(2)

브 문자가 기록된다. 해당 레지스트리 키를 점검할 경우에는 현재 혹은 최근에 연결되었던 매체의 드라이브 문자뿐만 아니라 과거에 동일한 드라이브 문자를 할당받았던 장치의 정보를 확인할 수 있다.

2.2 Windows Event

USB 매체가 시스템에 연결되거나 해제될 경우, 커널모드 PnP 관리자는 버스 드라이버에 전달받은 장치 식별번호를 이용하여 USB 장치의 동작을 지원하는 드라이버를 로드하거나 언로드한다. 이를 처리하기 위해서 Windows 시스템은 USB 매체와 관련된 시스템 이벤트를 발생시킨다[4]. USB 매체가 시스템에 연결되는 경우 2003, 2004, 2005, 2010, 2100, 2101, 2102, 2105, 20001, 20003 값의 ID를 갖는 이벤트들이 순차적으로 호출되며, USB 매체가 해제될 경우 2100, 2102, 2100, 1008 값의 ID를 갖는 이벤트들이 호출된다. 이렇게 호출된 이벤트들은 Windows 시스템에서 제공하는 이벤트 뷰어 프로그램을 이용해서 점검할 수 있다. Microsoft\Windows\DriverFrameworks-UserMode\Operational.evtx에 기록되어 있는 이벤트들에서 각 이벤트가 발생한 시간 정보와 이벤트를 처리되는 과정에서 참조되는 레지스트리 정보를 함께 확인할 수 있다[5]. USB 매체와 관련된 보다 상세한 정보를 확인하기 위해서 instance 값을 참조할 수 있다. 이벤트에 기록되어 있는 instance 값은 개별 이벤트들이 처리되는 과정에서 참조하는 레지스트리 키의 정보를 갖는다. HKLM\System\ControlSet00x\Enum을 기준으로 작성되며, instance 값을 통해서 확인한 레지스트리 키에서 USB 매체의 제조사, 모델명, 드라이버 정보 등을 확인할 수 있다.

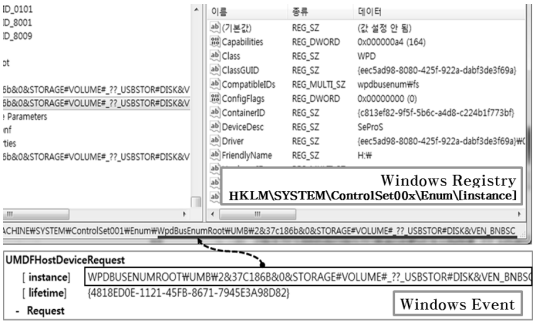

Fig. 4. Analysis the USB media using Windows event

2.3 Link 파일 및 Jump 파일 [6]

Link 파일과 Jump 파일은 사용자에게 파일 및 프로그램에 대한 접근 편의성을 제공하기 위해서 도입되었다. 대상 파일 및 프로그램이 위치하지 않는 경로에서 접근이 가능하도록 주소 정보를 갖고 있으며, Volume Serial Number, 최초 생성시간, 마지막 수정시간, 마지막 접근시간 등 다양한 정보를 갖기 때문에 작업이력 분석을 위해서 참조된다.

Link 파일과 Jump 파일에서 기록하는 작업이력들이 USB 매체가 시스템에 연결되었던 시간동안 발생하였고 링크 대상 객체의 주소가 USB 매체의 마운트 드라이브 문자를 포함한다면, USB 매체를 이용한 사용자의 작업이력으로 판단할 수 있다.

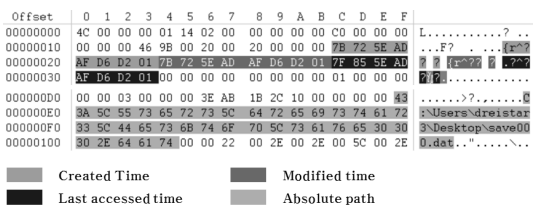

Fig. 5. Analysis the job history using link file

2.4 Browser History 파일

웹 브라우저에서 기록하는 히스토리 파일은 사용자가 이용한 웹 서비스 목록을 확인할 수 있기 때문에, 사용자의 작업이력 및 서비스 이용 패턴을 점검하기 위해서 참조된다. 히스토리 파일은 방문한 웹 사이트의 URL 주소, 방문한 시간, 방문한 횟수 등을 기록한다. 특히 방문한 사이트의 URL 주소에는 http, https 등과 같이 서비스 이용을 위해서 사용되었던 프로토콜 정보가 함께 기록된다.

웹 브라우저는 웹 서비스 이용을 위한 렌더링 작업뿐만 아니라 시스템에 위치한 txt, xml, pdf 등의 문서파일과 jpg, png, bmp, gif 등 이미지 파일에 대한 열람 기능을 지원한다. 시스템에 위치한 파일을 웹 브라우저를 통해서 열람한 경우에도 관련 내용들이 히스토리 파일에 기록되는데, URL 주소에는 프로토콜 정보를 대신하여 'file:///'이라는 문자열이 주소 정보와 함께 기록된다. 히스토리 파일에서 확인한 시간정보가 USB 매체가 시스템에 연결되었던 시간이고 URL 주소에서 확인한 열람한 객체의 주소가 USB 매체의 마운트 드라이브 문자를 포함한다면, 저장매체를 이용한 사용자의 작업이력으로 판단할 수 있다.

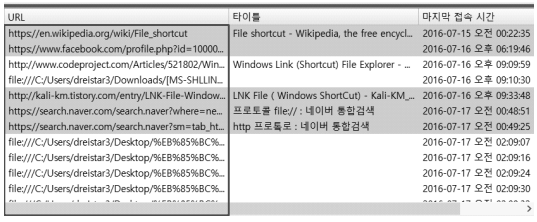

Fig. 6. Analysis the job history using browser history file

2.5 Prefetch 파일 [7]

프리패치 파일은 Windows 시스템에서 운영하는 메모리 관리기법 중 하나이다. 시스템의 부팅 혹은 프로그램의 실행 중에 접근하는 코드와 시스템 자원을 프로그램의 종류에 따라서 개별 파일로 기록한다. 프리패치 파일에 기록되어있는 자원은 시스템이 부팅되는 중 메모리에 로드되기 때문에, 부팅 시간과 프로그램의 실행시간을 단축시킨다. 프리패치 파일의 개수는 일반적으로 128개로 한정되어 있으며, 최대 개수를 초과할 경우 오래된 프리패치 파일을 삭제하고 새로 생성하기를 반복한다. 이 때문에 오래된 작업이력을 확인하기보다는 현재 혹은 최근에 수행되었던 작업이력을 판별하기 용이하다. 프리패치 파일은 일반적으로 %SystemRoot%\Prefetch에 위치한다. 프로그램마다 별개의 프리패치 파일이 존재하며, 프로그램의 이름, 실행 경로, 마지막 실행 시간, 참조하는 자원의 목록이 기록된다.

WinPrefetchView 도구를 이용하여 프리패치 파일의 정보를 확인한다. 도구의 하단에는 프로그램

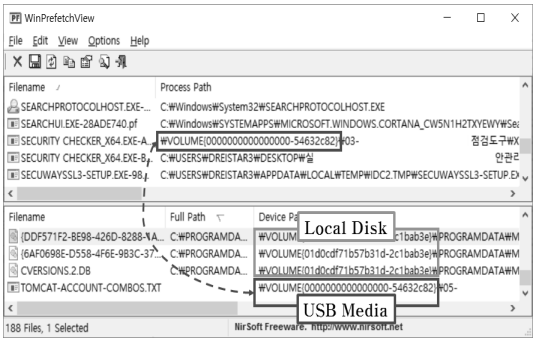

Fig. 7. Analysis the job history using Prefetch file

이 실행되는 중에 필요하거나 참조하는 자원의 정보가 출력된다. 주의해야 할 사항은 도구에서 보이는 Full Path는 프리패치 파일 내부에 포함된 정보가 아니라 도구에서 임의로 출력하는 정보라는 것이다. 로컬 볼륨에서 참조하는 객체의 드라이브 문자와 주소는 확인이 용이하지만, USB 매체의 경우 이를 식별하기 쉽지 않기 때문이다.

Full Path만을 이용하여 USB 매체의 작업이력을 판별하기란 쉽지는 않다. 추가적으로 해당 도구에서 확인할 수 있는 Device Path 정보를 참조한다. Device Path는 대상 객체의 주소를 마운트 드라이브 문자가 아닌 디스크 볼륨번호 혹은 볼륨 GUID 값을 기준으로 출력한다. 프로그램의 주소와 참조하는 자원의 주소에서 로컬 디스크 볼륨번호와 볼륨 GUID를 포함하지 않는다면, USB 매체에 의한 작업이력으로 판단할 수 있다. 프리패치 파일에서는 USB 매체에 대한 상세정보를 확인할 수 없기 때문에 작업의 주체가 되었던 USB 매체를 특정할 수는 없지만, 임의의 USB 매체에서 수행되었던 작업이력만은 참조할 수 있다.

2.6 IconCache.db 파일

IconCache.db 파일은 Windows 시스템에서 파일과 프로그램이 갖는 아이콘 이미지를 출력하기 위해서 참조된다. 기본 아이콘을 사용하는 일반적인 프로그램과는 달리, 대부분의 프로그램은 프로그램 특성을 나타낼 수 있는 별도의 아이콘 이미지를 사용한다. 프로그램의 아이콘 이미지를 출력하기 위해서 번번이 리소스 영역에 접근한다면, 시스템 자원의 낭비가 발생한다. 이를 방지하기 위해서 Windows 시스템은 프로그램의 아이콘 이미지와 관련된 정보를

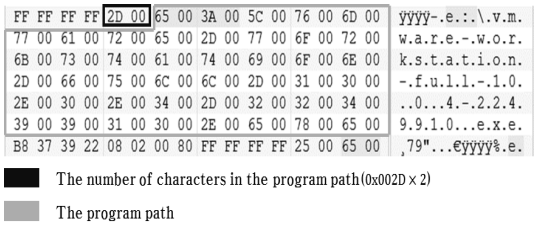

Fig. 8. Analysis the job history using IconCache

IconCache.db 파일에 저장한 뒤 시스템이 부팅할 때마다 메모리에 캐쉬하여 불필요한 자원의 낭비를 줄이고 있다[8].

IconCache.db 파일은 아이콘 이미지와 프로그램을 연관 짓기 위해서 아이콘 이미지의 색인 번호, 대상 프로그램의 위치 정보 등을 갖지만, 아이콘 이미지가 IconCache.db 파일에 기록된 시간 정보는 갖지 않는다. 또한 USB 매체를 특정할 수 있는 정보를 포함하지 않기 때문에 매체에 의한 작업이력을 판별하기에는 부적절할 수도 있다. 하지만 프로그램의 경로에서 확인할 수 있는 드라이브 문자만으로도 외부 매체에서 임의의 작업이 실행되었음을 추측할 근거가 마련된다. 프로그램의 경로가 로컬 드라이브의 마운트 문자를 포함하지 않을 경우 임의의 외부 매체를 이용한 작업이 있었음을 판단할 수 있다.

III. 아티팩트 상관관계를 활용한 작업이력 점검

앞서 설명한 시스템 아티팩트의 단일 분석만으로도 USB 매체에 대한 단편적인 작업이력 점검이 가능하다. 하지만 둘 혹은 셋 이상의 아티팩트에서 획득한 정보를 조합하면, 단일 아티팩트 분석만으로는 확인할 수 없는 작업이력 점검이 가능하다. 예를 들

Table 2. Analysis job history of the USB media by combining the data qualified by system artifacts

Data combination	Description
$A_{Dt} = (A_1 \cap A_2)$	the connection time of USB Media
$A_W = (A_3 \cup A_4 \cup A_5)$	the job history
$A_{Dt} \cap A_W$	the job history of USB media
$(A_{Dt})^c \cap (A_W)^c \cap A_6$ $(A_{Dt})^c \cap A_W \cap A_6$	the attempt to remove system artifacts

어, 악의적으로 작업이력을 숨기고자 아티팩트에 대한 훼손이나 삭제가 발생하였을 경우, 이를 판단하기 위한 근거를 아티팩트 상관관계에서 찾을 수 있다.

다음의 표는 아티팩트에서 획득한 정보의 상관관계 분석으로 점검할 수 있는 작업이력을 정리한다.

3.1 USB 매체 연결시간 점검

USB 매체의 연결시간은 작업이력의 판별뿐만 아니라 매체 사용자를 식별하기 위해서 반드시 확인해야한다. USB 매체의 연결시간을 확인하기 위해서 Windows 시스템의 이벤트 로그를 분석할 수 있으며, 매체의 보다 상세한 정보를 확인하기 위해서 관련된 레지스트리 키를 추적하는 방안을 제안한다.

USB 매체가 시스템에 연결되거나 해제될 경우 사용자 모드 PnP 관리자는 USB 매체의 정보를 이벤트 로그에 기록한다. Windows 시스템에서 지원하는 이벤트 뷰어 프로그램을 이용하여 개별 이벤트를 분석할 수 있다. USB 매체의 연결시간을 확인하기 위해서 이벤트에 기록된 lifetime 값을 참조한다. USB 매체가 시스템에 연결되는 매순간마다 생성되는 값으로 해당 매체와 시스템 사이의 통신과 관련된 이벤트들만이 공유한다. 동일한 USB 매체라 할지라도 시스템에 연결된 시간에 따라서 다른 lifetime 값을 갖는다. lifetime 값을 기준으로 USB 매체의 연결을 담당하는 이벤트와 해제를 담당하는 이벤트의 발생시간을 비교할 수 있다면, USB 매체가 시스템에 연결되고 해제되었던 시간정보를 확인할 수 있다.

Fig. 9. Analysis connection time of the USB meida

3.2 USB 저장매체 작업이력 점검

USB 저장매체의 작업이력을 점검하기 위해서 점검하고자 하는 USB 저장매체를 선별하고, 해당 매체가 시스템에 연결되었던 시간정보와 함께 시스템에 마운트되어 할당받은 드라이브 문자를 확인한다. 다음으로 파일, 프로그램에 대한 작업이력을 갖는 Link 파일 및 Jump 파일, 웹 브라우저의 히스토리 파일, Prefetch 파일을 분석하여, USB 저장매체가 마운트된 시간동안 생성, 수정, 실행 등의 작업이 수행되었음을 증명한다.

먼저 HKLM\SYSTEM\ControlSet00x\Enum\USBSTOR 레지스트리의 하위 키에서 작업이력을 분석할 USB 저장매체를 선별한다. 선별된 하위 레지스트리 키는 시스템이 매체와의 통신을 지원하기 위해서 생성한 이벤트에서 기록되는 instance와 동일한 값을 갖는다. instance 값을 기준으로 이벤트들을 분류하고, lifetime 값을 기준으로 이벤트를 소규모 그룹으로 다시 한 번 분류한다. 이벤트에서 기록되는 lifetime 값은 USB 매체가 삽입되어있는 시간동안 생성된 이벤트만이 공유하는 값으로 동일한 매체라 할지라도 시스템에 삽입된 시간이 다를 경우 별개의 값을 갖는다. lifetime으로 구분한 이벤트 소그룹에서 매체의 연결 이벤트 발생시간과 해제 이벤트 발생시간을 비교하여, USB 저장매체가 시스템에 연결되었던 정확한 시간정보를 확인한다.

레지스트리와 이벤트 분석으로 확인한 USB 저장매체의 시스템 연결시간과 마운트 드라이브 문자를 기준으로 USB 저장매체의 작업이력을 점검한다. 파일 혹은 프로그램에 대한 생성, 삭제, 수정 등의 작업이 실행되었을 경우, 작업이력을 추적할 수 있는 시스템 아티팩트가 생성된다. 대표적인 예로 link 파일, jump 파일, 웹 브라우저 history 파일, prefetch 파일 등이 존재한다. 이들 아티팩트를 분석하면 파일 혹은 프로그램을 대상으로 수행되었던 작업의 시간정보와 함께 대상 프로그램의 절대경로를 확인할 수 있다. 작업이 발생한 시간이 USB 저장매체가 시스템에 연결된 시간이고, 대상 파일 혹은 프로그램의 절대경로가 USB 저장매체가 시스템에 마운트되어 할당받은 드라이브 문자를 포함할 경우 해당 매체에서의 작업이력을 판별할 수 있다.

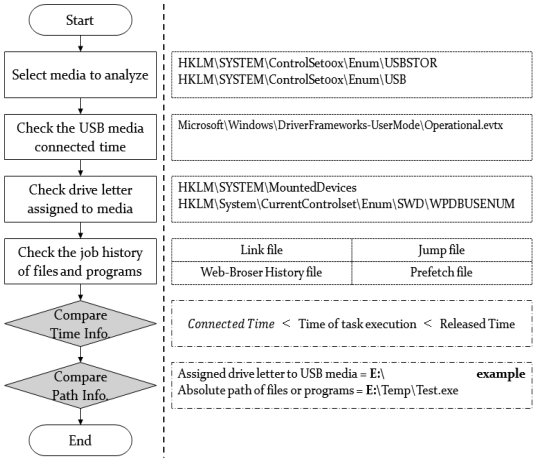

Fig. 10. The flowchart to analysis job history of the USB media

3.3 매체 충전 및 단순 연결이력 점검

USB 포트를 이용한 매체가 증가함에 따라서 단순히 충전만을 목적으로 USB 매체를 시스템에 연결하는 경우가 증가하고 있다. 이러한 기기들 중에서도 특히 스마트폰, 태블릿 PC, 내비게이션 등 자료저장 기능을 보유한 장비를 시스템에 연결하는 경우가 증가함에 따라서 단순 충전을 위한 시스템 연결이었는지, 저장매체로써 이용을 위한 시스템 연결이었는지 구분할 필요성이 요구되고 있다.

시스템에 새로운 USB 매체를 연결하면, PnP 관리자는 매체의 동작을 지원하는 드라이버를 시스템에 설치한다. 커널모드 PnP 관리자는 시스템에 설치된 드라이버를 분석하여, USB 매체의 용도를 파악한다. 만일 드라이버 설치가 저장매체 이용을 위한 것으로 판명된다면, USB 매체와 관련된 정보들은 HKLM\SYSTEM\ControlSet00x\Enum\USBSTOR에 기록된다. 시스템 이벤트 분석으로 저장매체의 기능을 갖는 USB 매체의 연결이 확인되었지만, HKLM\SYSTEM\ControlSet00x\Enum\USBSTOR 레지스트리에서 해당 USB 매체의 정보가 확인되지 않고, USB 매체가 시스템에 연결된 시간동안 파일 및 프로그램에 대한 작업이력이 확인되지 않는다면, 해당 매체는 단순히 충전만이 목적으로 시스템에 연결되었음을 확인한다. 하지만 레지스트리에 대한 변조 및 삭제가 용이하고, USB 매체와 관련된 레지스트리를 전문적으로 훼손하는 도구들을 쉽게 구할 수 있다는 점에서 충전만을 목적으로 시스템에 연결되었는지 정확하게 판별하기가 쉽지 않다.

3.4 증적 데이터 훼손 및 삭제 점검

USBDeview, USB Oblivion 등 시스템에서 USB 매체의 정보를 식별하고, 매체와 관련된 시스템 아티팩트를 훼손하거나 삭제하는 도구를 손쉽게 구할 수 있다. 이와 같은 도구의 기능은 대부분 USB 매체의 드라이버 설치정보를 기록하는 로그 파일과 USB 매체의 정보를 담고 있는 레지스트리 분석을 바탕으로 한다. 시스템에서 USB 매체의 정보를 감추기 위해서 도구는 로그 파일과 레지스트리를 훼손한다. USB 매체의 정보를 확인하고, 증적 데이터 훼손여부를 점검하기 위해서는 다른 아티팩트 분석을 진행해야만 한다.

USB 매체의 드라이버 설치정보를 기록하는 로그 파일과 USB 매체의 정보를 갖는 레지스트리가 훼손 및 삭제된 상황을 가정한다. 먼저, 프리패치 파일은 프로그램의 실행에 필요한 자원의 정보를 기록한다. 세부적으로는 프로세스의 이름, 프로세스의 경로, 생성 시간, 접근 시간, 프로그램 동작에 필요한 라이브러리 정보 등이 포함된다. USB 매체에서 프로그램을 실행하였을 경우 프리패치 파일에는 프로그램의 실행경로가 마운트 드라이브 문자가 아닌 디스크 볼륨번호 혹은 볼륨 GUID 값을 기준으로 기록된다. 또한 프로그램 접근 시간은 가장 마지막에 접근한 시간 정보뿐만 아니라 가장 마지막에 접근한 시간을 포함한 최대 8번의 접근 시간 정보를 순차적으로 저장한다. 이벤트 분석으로 확인한 USB 매체의 연결 시간 중에 프리패치 파일에서 확인한 프로그램의 실행

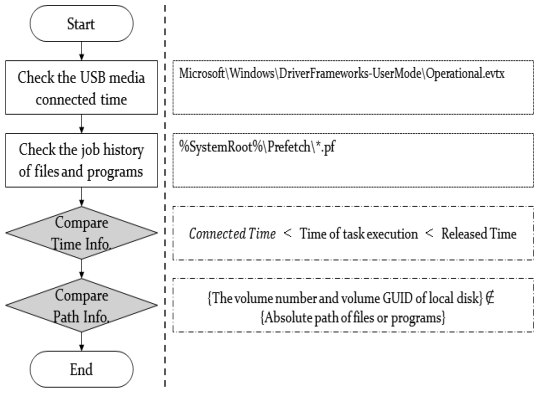

Fig. 11. The flowchart to analysis the attempt to remove system artifact

시간이 포함되고, 프리패치 파일에서 기록한 프로그램의 실행경로가 로컬 디스크 볼륨번호 혹은 볼륨 GUID 값을 포함하지 않는 경우, 증적 데이터에 대한 훼손 및 삭제 시도가 있었음을 판단할 수 있다.

다음으로 IconCache.db 파일은 프로그램을 대표하는 아이콘 이미지 정보를 저장한다. 아이콘 이미지와 프로그램의 연계성을 확보하기 위해서 아이콘 이미지를 가리키는 색인 정보와 프로그램의 경로를 함께 기록한다. IconCache.db 파일에서 기록하는 파일 및 프로그램의 경로에서 로컬 드라이브 문자를 포함하지 않는다면, USB 매체를 이용한 작업이력이 존재하였음을 유추할 수 있다. 또한 파일에서 USBDeview, USB Oblivion 등의 프로그램 정보가 확인된다면, 증적 데이터에 대한 훼손 및 삭제 시도가 있었음을 추론할 수 있다[8].

IV. 결 론

본 논문에서는 USB 매체의 작업이력을 점검하기 위해서 참조할 수 있는 시스템 아티팩트와 분석방법을 소개하였고, 둘 이상의 아티팩트에서 획득한 정보의 상관관계를 토대로 추가적인 작업이력 점검하는 방안을 제안하였다. USB 매체의 작업이력을 점검하기 위한 분석대상은 분석하고자 하는 USB 매체와 USB 매체가 연결되었던 시스템으로 나뉜다. USB 매체가 연결되어 시스템에 새로이 생성되거나 변경된 시스템 아티팩트 분석으로 USB 매체의 작업이력을 점검한다. 단일 아티팩트 분석은 시스템에 연결되었던 USB 매체에 대한 식별, 파일 및 프로그램의 작업이력 점검 등 단편적 사실을 검증하며, 다중 아티팩트 분석은 USB 매체의 시스템 연결시간 점검, USB 매체의 작업이력 점검 등 보다 상세한 작업이력 검증을 가능하게 한다. USB 매체의 훼손으로 직접적인 분석이 힘든 상황에서도 시스템 분석으로 매체의 작업이력 검증이 가능하다. 하지만, 논문에서 소개한 아티팩트는 훼손 및 삭제가 용이하며, 이 때문에 작업이력 점검에 어려움을 갖는 것이 현실이다. 향후에는 훼손된 아티팩트에 대한 복원 연구 및 USB 매체의 작업이력을 검증하기 위한 추가적인 아티팩트 분석 연구를 진행할 계획이다.

References

[1] Dongho Won, "USB memory storage market trend in USA", Retrieved from http://news.kotra.or.kr/user/globalBbs/kotranews/25/globalBbsDataView.do?setIdx=254&dataIdx=136436&pageViewType=&column=title&search=%EB%AF%B8%EA%B5%AD%20usb&searchAreaCd=&searchNationCd=&searchTradeCd=&searchStartDate=&searchEndDate=&searchCategoryIdxs=&searchIndustryCateIdx=&searchItemCode=&searchItemName=&page=1&row=10KOTRA, Oct. 2014.

[2] Jin-Kuk Kim, "USB Device Tracking on Windows", Retrieved from http:// forensic-proof.com/archives/3632, Jun, 2012.

[3] Tanushree Roy, Aruna Jain, "Windows Registry Forensics : An Imperative Step in Tracking Data Theft via USB Devices", International Journal of Computer Science and Information Technologies, vol3, pp.4427-4433, 2012

[4] Jan Axelson, "USB Complete 3rd", acorn-pub, pp.101-103, Jan. 2011.

[5] Jason Hale, "The Windows 7 Event Log and USB Device Tracking", Retrieved from http://dfstream.blogspot.kr/2014/01/the-windows-7-event-log-and-usb-device.html, Jan. 2014.

[6] Microsoft Corporation, "Shell Link (.LNK) Binary File Format", pp.10-15, Jul, 2016.

[7] Narasimha Shashidhar, Dylan Novak, "Digital Forensic Analysis on Prefetch Files", International Journal on Information Security Science, Vol.4, pp.39-49, Jun. 2015.

[8] Chan-Youn Lee, Sangjin Lee, "Structure and application of IconCach e.db files for digital forensics", Digital Investigation, Vol.11, pp.102-110, May. 2014.

〈저자소개〉

사 진

이 성 재 (Seong-Jae Lee) 정회원
2013년 2월: 전남대학교 전자컴퓨터공학부 (공학사)
2015년 2월: 전남대학교 정보보안협동과정 (이학석사)
2015년 3월~현재: 전남대학교 정보보안협동과정 박사과정
2015년 5월~현재: ETRI 부설연구소 기술원
〈관심분야〉 디지털 포렌식, 악성코드 분석, 취약점 분석

노 봉 남 (Bong-Nam Noh) 종신회원
1987년: 전남대학교 수학교육과 (이학사)
1992년: KAIST 전산학과 (이학석사)
1994년: 전북대학교 전산과 (이학박사)
1983년~현재: 전남대학교 전자컴퓨터공학부 교수
2000년~현재: 시스템보안연구센터 소장
〈관심분야〉 디지털 포렌식, 시스템 및 네트워크 보안, 정보사회와 사이버 윤리