Windows 10 Artifact Analysis Tool (H - AAT) 개발

김유철 성중곤 홍수빈

서 울 호 서 전 문 학 교
사이버해킹보안과

내용

그림 목차

1. 서론

Microsoft사의 신규 운영체제 Windows 10이 2015년 7월 29일에 공식 발표되었다. Microsoft는 올해 7월 1일부터 기존 Windows7, 8, 8.1 운영체제 사용자를 대상으로 무료 업그레이드 예약을 받았다. 그리고 7월 29일부터 순차적으로 전세계 190개국의 기존 운영체제 사용자를 대상으로 한 Windows 10 업그레이드가 진행되었다. 시장조사기관에서의 PC 운영체제 점유율 조사결과, 2015년 9월을 기준으로 Windows 10의 시장 점유율은 6.63%에 달하는 것으로 나타났다.

이전의 Windows 운영체제와 다르게 기존 사용자에게는 무료로 배포되므로, 많은 사용자가 점차 Windows 10으로 업그레이드를 진행할 것으로 예상된다. Windows 8에서 8.1의 마이너 업그레이드와는 다르게 신규 운영체제로 메이저 업그레이드 되면서 많은 신기능이 추가됐고, 이에 따라서 기존에는 존재하지 않았던 사용자의 새로운 아티팩트가 생성되었다.

포렌식 조사관의 입장에서 신규 운영체제에 대한 아티팩트 정보의 사전지식은 포렌식 분석에 있어서 발 빠른 대처에 중요하게 작용하는 요인이다. 만약에 신규 운영체제에 추가된 아티팩트 정보를 활용하여 사용자 흔적 수집한다면, 아티팩트에 대해 조사하는 과정이 불필요하게 된다. 결과적으로, 분석에 소요되는 시간을 절감할 수 있으며 정보 누락으로 인한 잘못된 판단을 예방할 수 있다.

따라서, 본 프로젝트에서는 Windows 10의 아티팩트를 조사하고 결과를 확인할 수 있는 도구를 제작하여 포렌식 조사관이 실제 분석 시에 적용할 수 있도록 하는 것에 초점을 두고 진행하였다.

2. 아티팩트의 정의

2.1 아티팩트(Artifact) 정의

아티팩트란 사전적 의미로 "인공물", "유물"을 뜻한다. 영문으로 Artefact 또는 Artifact 라고 표기한다. 일반적으로, 시스템에 생성되는 증거를 생성증거와 보관증거 2가지로 분류하며 생성증거에 해당하는 것이 아티팩트다. 두 증거의 차이점은 데이터에 고의성이 있는지 여부이다. 보관 증거의 경우, 고의가 들어간 데이터이기 때문에 전문 법칙이 적용되어 증거로 인정받기 위해서는 전문 법칙에 예외 규정을 따져봐야 하지만, 생성 증거의 경우 전문 법칙이 적용되지 않아 증거로의 가치가 매우 높다.

2.2 디지털 포렌식 관점의 아티팩트

디지털 포렌식 관점에서 아티팩트의 의미는 사용자가 운영체제와 애플리케이션을 사용했을 때 자동으로 생성되는 흔적을 말한다.

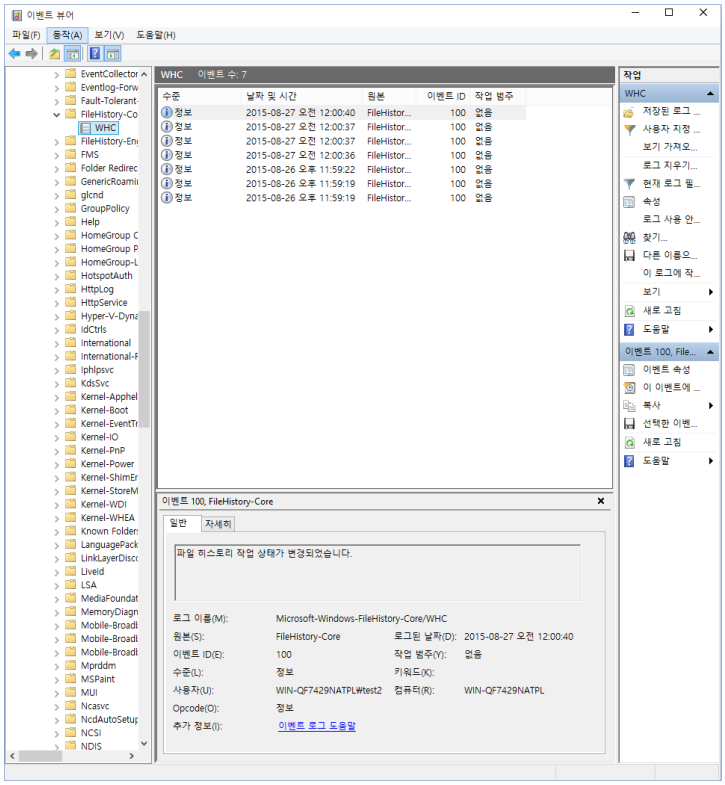

그림 1 아티팩트의 예(이벤트 로그)

위 [그림 1]과 같이 사용자가 운영체제에서 한 행위는 모두 이벤트 로그에 기록된다. 디지털 포렌식 관점으로 Windows 시스템에 접근했을 때 찾을 수 있는 아티팩트는 레지스트리, 프리/슈퍼패치, 이벤트 로그 등이 있다. 본 프로젝트에서는 이와 같은 Windows 10 아티팩트를 항목별로 분류한 후 분석하는 도구 개발을 진행했다.

3. Windows 10 아티팩트 분석 도구(H-AAT) 개발

3.1 H-AAT(Hoseo - Artifact Analysis Tool) 개요

아티팩트 분석시, 포렌식 조사관이 직접 수작업으로 항목별 경로를 찾으며 조사를 진행하는데 있어 많은 시간이 소요된다. 따라서, 어떠한 환경에서든지 한 번의 클릭만으로 PC의 레지스트리 정보와 Windows 10 신규 아티팩트 정보들을 자동으로 분석해주는 도구가 필요하다. 본 프로젝트에서는 Batch 스크립트 파일로 시스템의 아티팩트를 분석하고, 보고서를 HTML파일로 자동으로 도출해 웹 형태로 보고서를 출력시키는 방향으로 개발했다.

.

PC 이름	WIN-HT34JTKJUKI
O S	Microsoft Windows 10 Pro K
OS 버전	10.0.10240 N/A 빌드 10240
C P U	Intel i7-4960X 3.60GHz
R A M	16GB
H D D	4TB
S S D	256GB

그림 2 분석 PC 환경

분석은 [그림 2]에 제시된 PC환경에서 진행되었고, 다음 항목과 같이 Windows10에서 추가, 변경된 기능과 아티팩트를 분류해 해당 경로의 디렉터리 정보와 레지스트리 값을 추출해서 웹페이지 형태로 출력될 수 있도록 Batch 스크립트를 작성했다.

3.2 어플리케이션(Application)

3.2.1 Internet Explorer

사용자가 Windows 10 기본 내장 웹 브라우저인 Internet Explorer를 통해 방문했던 사이트의 캐시(cache)파일 및 쿠키(Cookie)파일이 아래 경로에 존재한다.

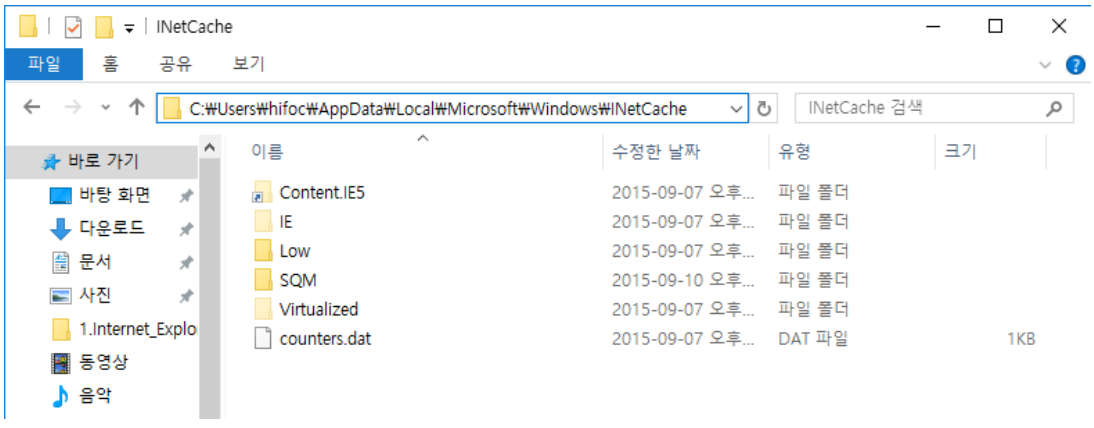

그림 3 INetCache 디렉터리

- 인터넷 캐시(Internet Cache)가 저장된 디렉터리
  %LocalAppData%₩Microsoft₩Windows₩INetCache

```
55  dir /a "C:\Users\%USERNAME%\AppData\Local\Microsoft\Windows\INetCache" >Extraction\Analyze\inetcache.txt
```

그림 4 INetCache 추출

[그림 4]의 dir 명령어를 사용해 해당 디렉터리의 정보를 inetcache.txt 파일에 작성했다.

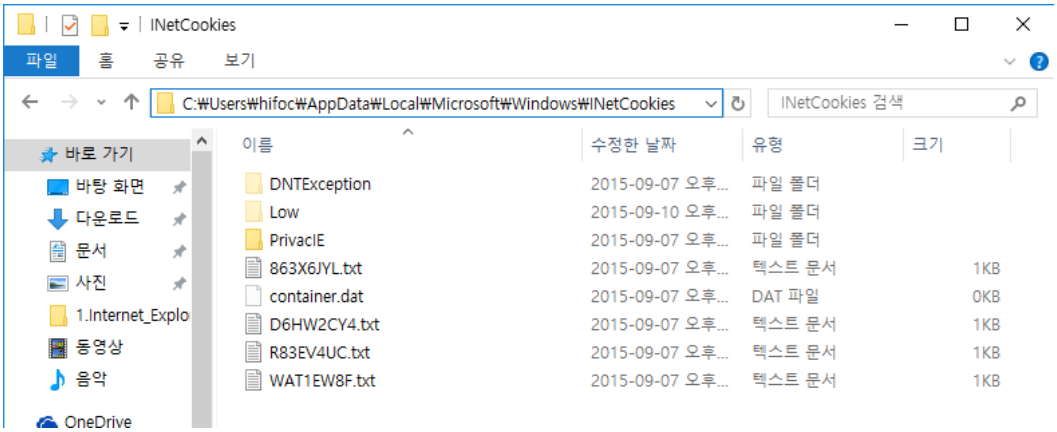

그림 5 INetCookies 디렉터리

- 인터넷 쿠키(Internet Cookies)가 저장된 디렉터리
  %LocalAppData%₩Microsoft₩Windows₩INetCookies

```
57  dir /a "C:\Users\%USERNAME%\AppData\Local\Microsoft\Windows\INetCookies" >Extraction\Analyze\inetcookie.txt
```

그림 6 INetCookies 추출

[그림6] 의 명령어를 사용해 해당 디렉터리의 정보를 inetcookie.txt 파일에 작성했다.

3.2.2 Microsoft Edge

Windows 10에서 새로 추가된 웹 브라우저 어플리케이션이다. Preview버전에서 Project Spartan이라는 이름으로 공개되었지만 정식 배포버전부터 Edge라고 변경되었다.
명칭은 변경되었지만 Preview버전 당시 Project Spartan 브라우저와 아티팩트 저장 경로는 동일하였다. Edge 브라우저의 환경설정, 웹 캐시(Web Cache), 캐시(Cache), 쿠키(Cookies), 마지막 활성 브라우징 세션(Last Active Browsing Session)은 아래 경로에 존재한다.

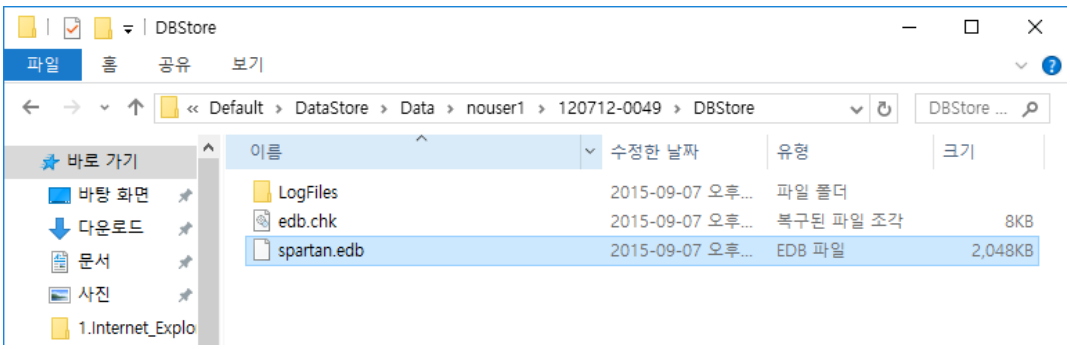

그림 7 Edge 브라우저 환경설정

- Edge 브라우저 환경설정이 저장된 edb 파일 경로
  %LocalAppData%₩Packages₩Microsoft.MicrosoftEdge_8wekyb3d8bbwe₩AC₩MicrosoftEdge₩User₩Default₩DataStore₩Data₩nouser1₩120712-049₩DBStore₩spartan.edb

[그림 7]의 spartan.edb파일은 Edge 브라우저에서 사용자가 조작한 환경설정을 담고 있는 edb 파일이다.

```
64  dir /a "C:\Users\%USERNAME%\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\MicrosoftEdge\User\Default\DataStore\Data\nouser1\120712-0049\DBStore\spartan.edb" >Extraction\Analyze\edgeSetting.txt
```

그림 8 Edge 브라우저 환경설정 추출

[그림8]의 명령어를 사용해 해당 디렉터리 정보를 edgeSetting.txt파일에 작성했다.

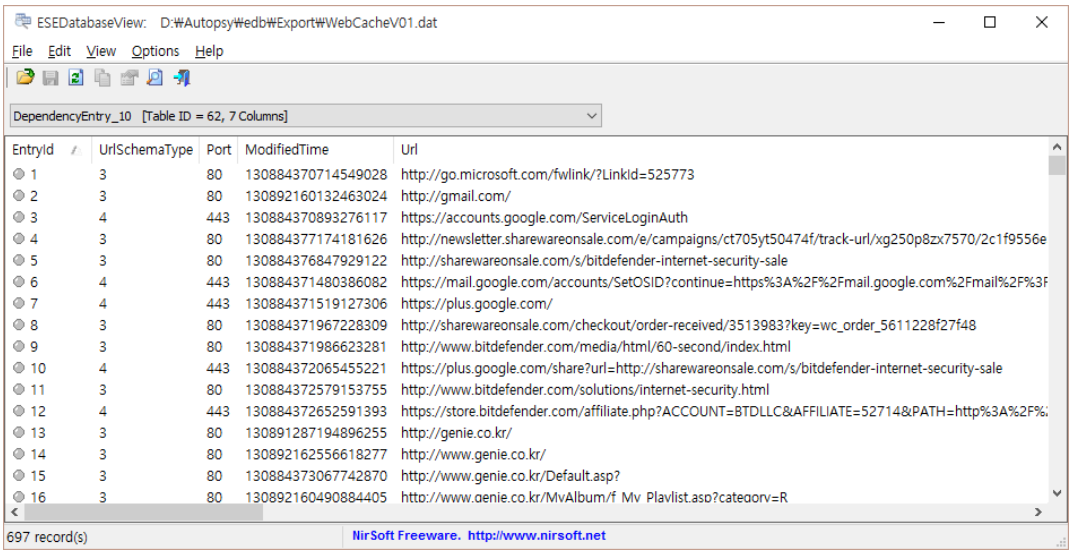

그림 9 Edge 브라우저 웹 캐시(Web Cache)

- Edge 브라우저 웹 캐시(Web Cache) 저장 경로
  %LocalAppData%₩Microsoft₩Windows₩WebCache₩WebCacheV01.dat

[그림 9]는 nirsoft(www.nirsoft.net)의 ESEDatabaseView 프로그램을 통해 Web Cache를 확인한 결과다. 사용자가 Edge 브라우저에서 방문한 웹사이트의 주소가 남아있다

```
76  dir /a %LocalAppData%\Microsoft\Windows\WebCache >Extraction\Analyze\edgeWebCache.txt
```

그림 10 Edge 브라우저 웹 캐시(Web Cache)추출

[그림10]의 명령어를 사용해 해당 디렉터리의 정보를 edgeWebCache.txt파일에 작성하였다.

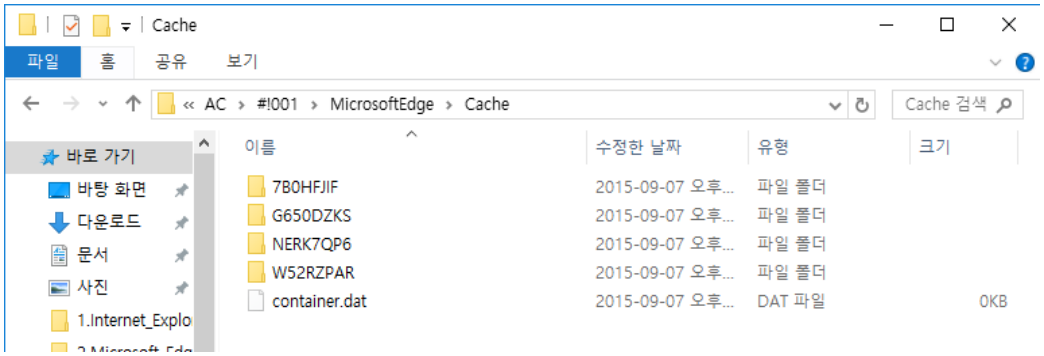

그림 11 Edge 브라우저 캐시(Cache)

- Edge 브라우저 캐시(Cache) 저장 경로
%LocalAppData%₩Packages₩Microsoft.MicrosoftEdge_8wekyb3d8bbwe₩AC₩#!001₩MicrosoftEdge₩Cache

```
66 dir /a "C:\Users\%USERNAME%\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\#!001\MicrosoftEdge\Cache" >Extraction\Analyze\edgeCache.txt
```

그림 12 Edge 브라우저 캐시(Cache)추출

[그림 12]의 명령어를 사용해 해당 디렉터리의 정보를 edgeCache.txt폴더에 작성했다.

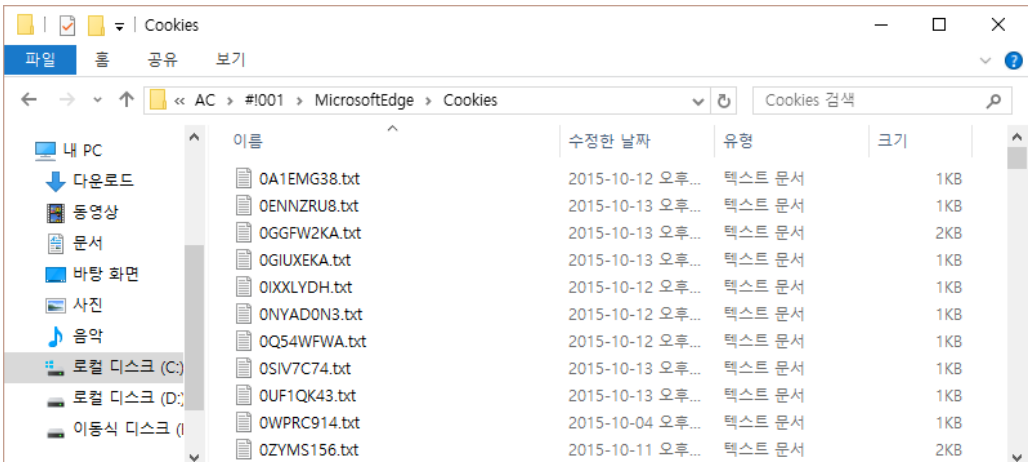

그림 13 Edge 브라우저 쿠키(Cookies)

- Edge 브라우저 쿠키(Cookies) 저장 경로
%LocalAppData%₩Packages₩Microsoft.MicrosoftEdge_8wekyb3d8bbwe₩AC₩#!001₩MicrosoftEdge₩Cookies

[그림 7]의 디렉터리 내부에 존재하는 txt 파일에 사용자가 Edge 브라우저를 사용하면서 남은 사이트별 쿠키(Cookie)값이 기록되어있다.

```
74 dir /a "C:\Users\%USERNAME%\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\#!001\MicrosoftEdge\Cookies" >Extraction\Analyze\edgeCookie.txt
```

그림 14 Edge 브라우저 쿠키(Cookies) 추출

[그림 14]의 명령어를 사용해 해당 디렉터리 정보를 edgeCookie.txt파일에 작성했다.

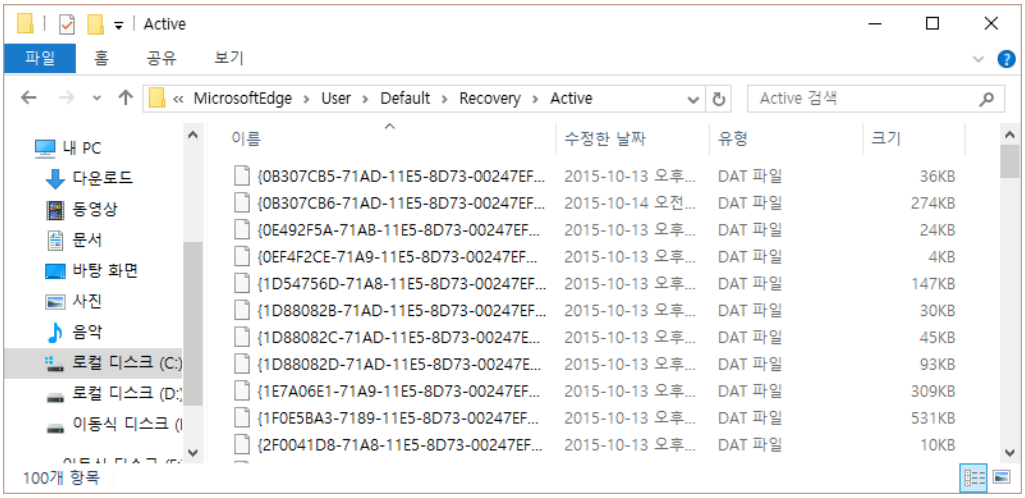

그림 15 브라우저 마지막 활성 브라우징 세션(Last Active Browsing Session)

- Edge 브라우저 마지막 활성 브라우징 세션(Last Active Browsing Session) 경로
  %LocalAppData%₩Packages₩Microsoft.MicrosoftEdge_8wekyb3d8bbwe₩AC₩MicrosoftEdge₩User₩Default₩Recovery₩Active

```
68 dir /a "C:\Users\*USERNAME*\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\MicrosoftEdge\User\Default\Recovery\Active">Extraction\Analyze\edgeSession.txt
```

그림 16 브라우저 마지막 활성 브라우징 세션(Last Active Browsing Session)추출

[그림 16]의 명령어를 사용해 해당 디렉터리의 정보를 edgeSession.txt에 작성했다.

3.2.3 Windows Store

Windows Store관련 아티팩트는 이벤트로그(Event Log), 어플리케이션 검색 및 설치 흔적, 어플리케이션 삭제 흔적이 존재한다.

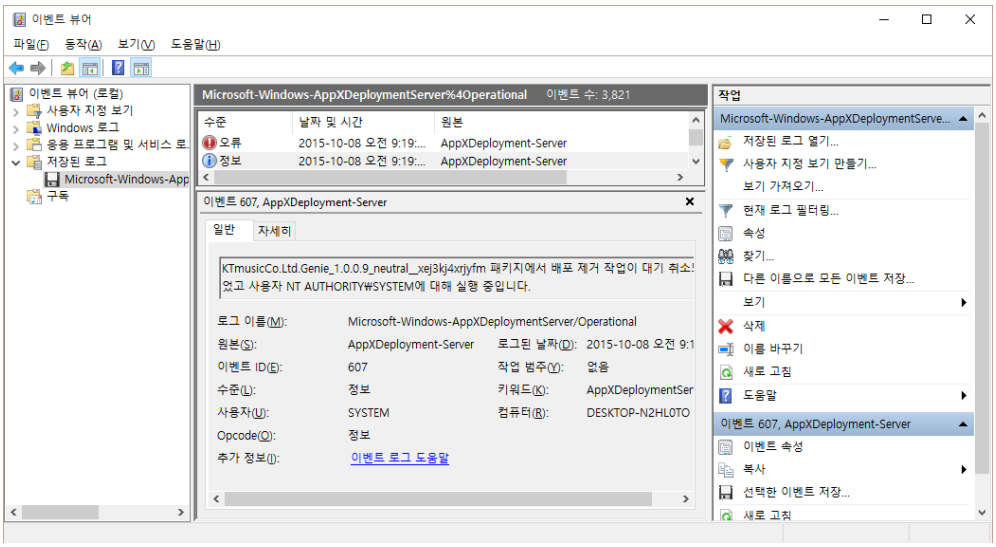

그림 17 Windows Store Deployment 이벤트 로그

- Windows Store Deployment이벤트 로그 경로
  C:₩Windows₩System32₩winevt₩LogsMicrosoft-Windows-AppXDeployment%4Operational.evtx

```
115  dir /a C:\Windows\System32\winevt\Logs >Extraction\Analyze\WSeventLog.txt
```

그림 18 Windows Store 이벤트 로그 추출

[그림 18]의 명령어로 Windows Store 이벤트 로그 파일들을 WSeventLogs.txt파일에 작성했다.

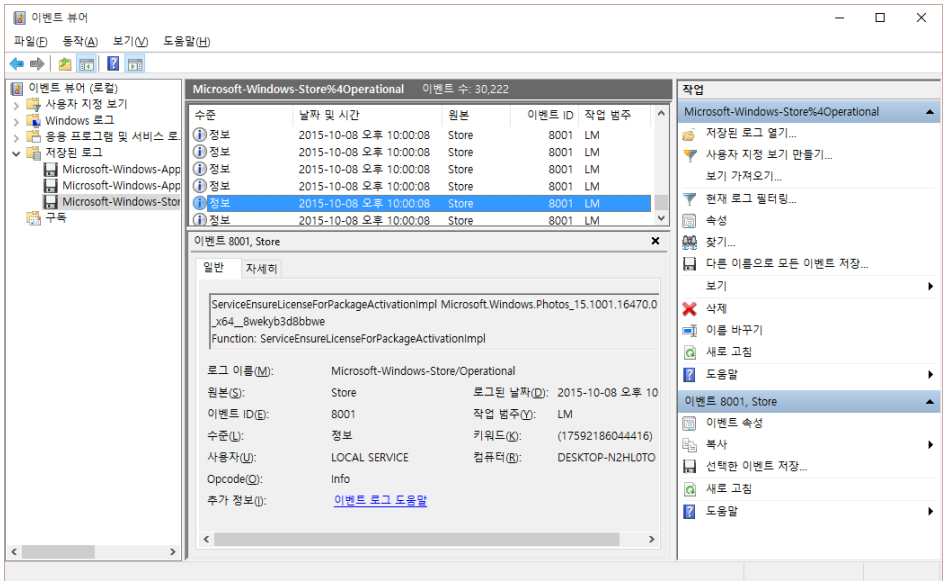

그림 19 Windows Store 설치, 검색 이벤트 로그

- Windows Store 설치, 검색 이벤트 로그 경로
  C:₩Windows₩System32₩winevt₩Logs₩Microsoft-Windows-Store%4Operational.evtx

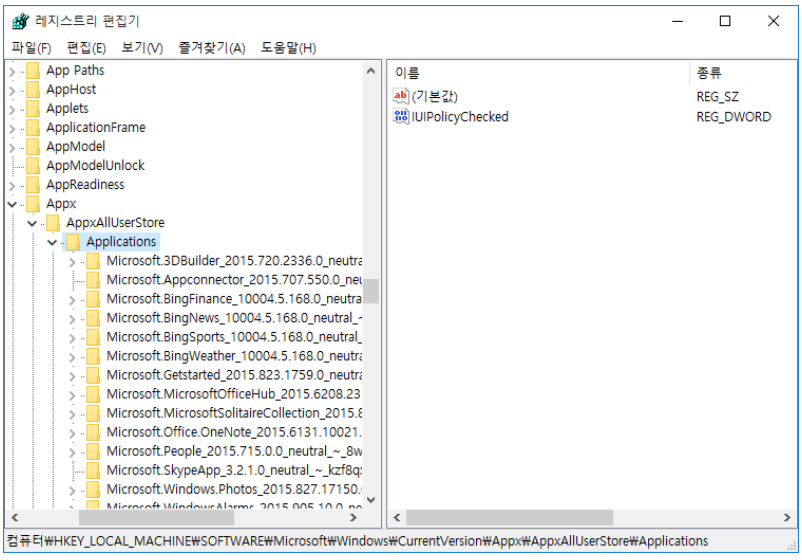

그림 20 설치된 Windows Store 어플리케이션 레지스트리

- 설치된 Windows Store 어플리케이션 레지스트리 경로
HKEY_LOCAL_MACHINE₩Software₩Microsoft₩Windows₩CurrentVersion₩Appx₩AppxAllUserStore₩Applications₩

```
117 reg query "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Appx\AppxAllUserStore\Applications">Extraction\Analyze\installedApp.txt
```

그림 21 Windows Store 설치 이벤트 로그 추출

[그림 21]의 명령어를 사용해 설치된 Windows App 목록을 installedApp.txt에 작성했다.

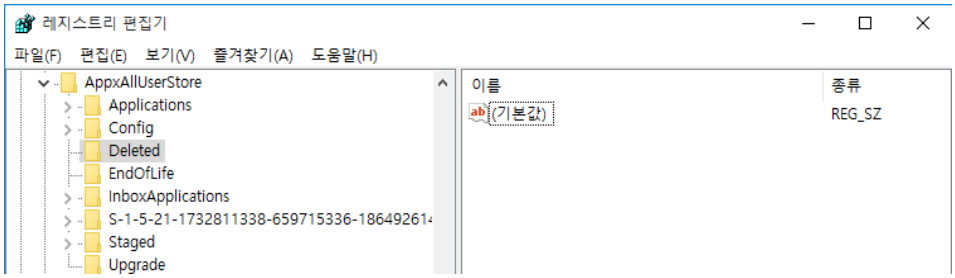

그림 22 삭제된 Windows Store 어플리케이션 레지스트리

- 삭제된 Windows Store 어플리케이션 레지스트리 경로
HKEY_LOCAL_MACHINE₩Software₩Microsoft₩Windows₩CurrentVersion₩Appx₩AppxAllUserStore₩Deleted

```
119 reg query "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Appx\AppxAllUserStore\Deleted">Extraction\Analyze\deletedApp.txt
```

그림 23 Windows Store 삭제 이벤트 로그 추출

[그림 23]의 명령어를 사용해 삭제된 Windows App목록을 deletedApp.txt에 작성했다.

3.2.4 Faceook App

Facebook App 사용자가 운영체제에 남기는 아티팩트의 종류로는 친구 정보, 친구 요청 정보, 메시지, 알림, 스토리가 존재한다. 이 정보들은 sqlite파일로 저장된다. sqlite 파일을 열어 저장된 정보를 확인하기 위해서는 'sqlitebrowser'와 같은 도구를 사용하면 된다.

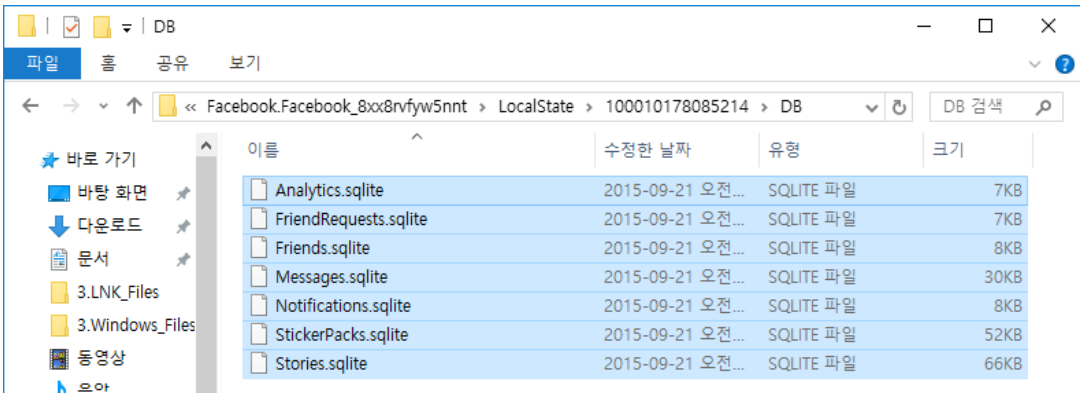

그림 24 Facebook App 아티팩트

3.2.5 Twitter App

Twitter App은 사용자가 앱을 통해 방문했던 트위터 페이지의 캐시 파일이 아래 경로에 저장된다.

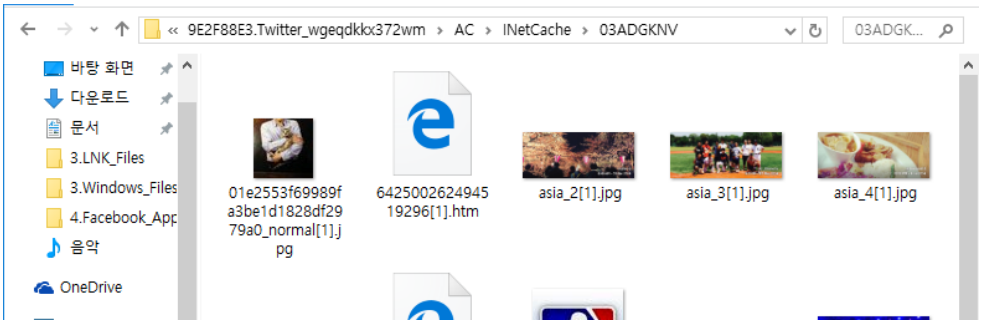

그림 25 Twitter App 페이지 캐시파일

- Twitter App 페이지 캐시파일 저장 경로
%LocalAppData%₩Packages₩9E2F88E3.Twitter_wgeqdkkx372wm₩AC₩INetCache

3.2.6 Maps App

Windows 10 기본 어플리케이션인 Maps App은 사용자가 어플리케이션에서 검색했던 정보 및 검색되었던 장소의 위도, 경도를 확장자가 ttl인 파일을 저장한다.

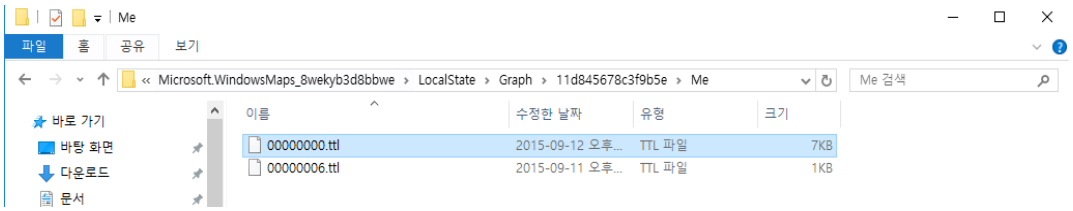

그림 26 Maps App 아티팩트

- Maps App 아티팩트 저장 경로
%LocalAppData%₩Packages₩Microsoft.WindowsMaps_8wekyb3d8bbwe₩LocalState₩Graph₩11d845678c3f9b5e₩Me

```
103 dir /a "C:\Users\%USERNAME%\AppData\Local\Packages\Microsoft.WindowsMaps_8wekyb3d8bbwe\LocalState\Graph\11d845678c3f9b5e">Extraction\Analyze\maps.txt
```

그림 27 Maps App 정보 추출

[그림 27]의 명령어를 사용해 Maps App의 아티팩트를 maps.txt에 작성했다.

3.3 레지스트리(Registry)

3.3.1 Web Browser

Windows 10에 내장된 웹 브라우저인 Internet Explorer는 사용자가 주소창에 입력했던 기록을 비롯해 마지막 크롤링(crawling), 업그레이드 시간, 즐겨찾기 경로, 브라우저 버전

정보를 레지스트리에 모두 저장한다. 레지스트리 저장 경로는 아래와 같다.

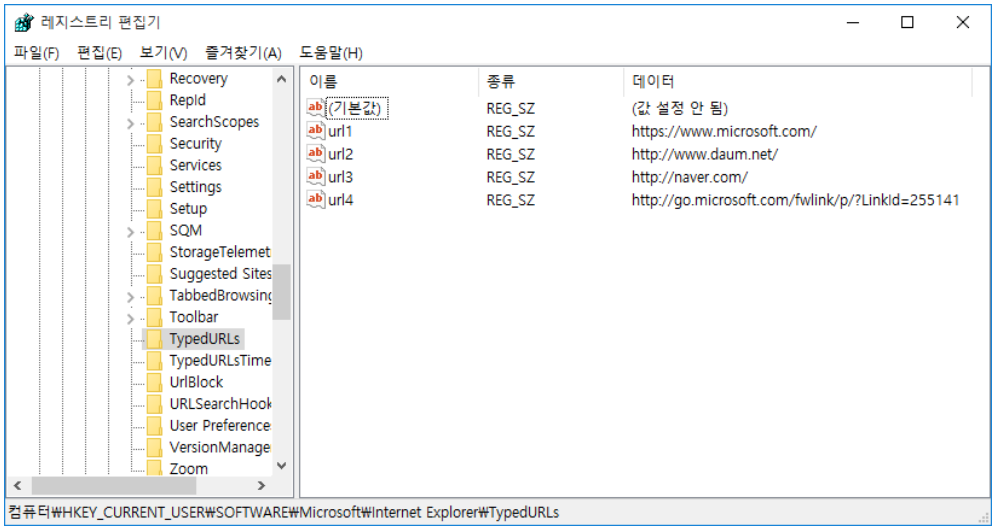

그림 28 Internet Explorer 주소창 입력기록

- Internet Explorer 주소창 입력기록 레지스트리 경로
HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Internet Explorer₩TypedURLs

```
74  reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\TypedURLs"> Extraction\Analyze\typedURL.txt
```

그림 29 Internet Explorer 주소창 입력기록 추출

[그림 29]의 명령 프롬프트 dir 명령어를 사용해서 레지스트리에 저장된 IE 주소창에 입력했던 기록정보를 typeURL.txt 파일에 작성했다.

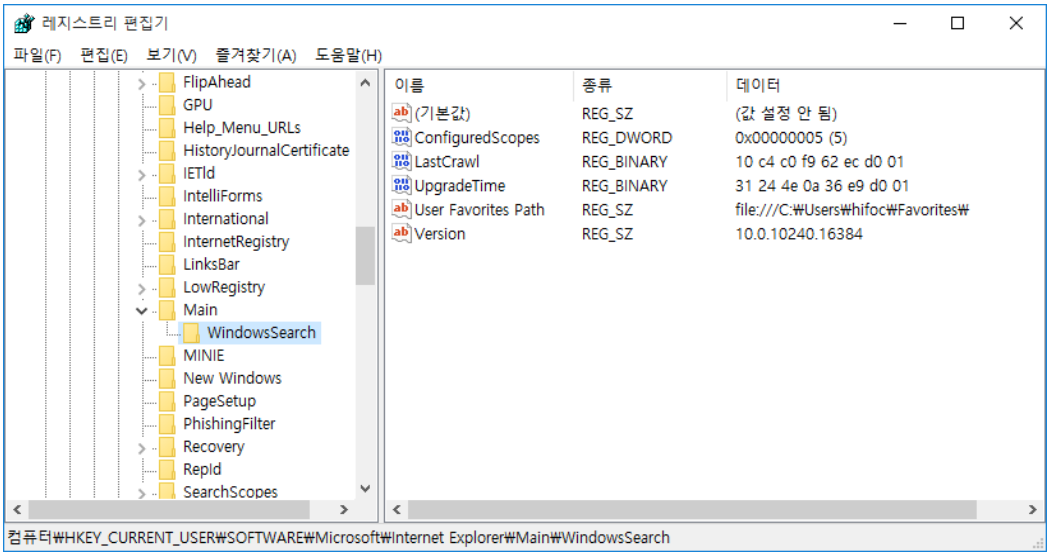

그림 30 Internet Explorer 정보 레지스트리

- Internet Explorer 정보 레지스트리 경로
HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Internet Explorer₩Main₩WindowsSearch

```
76 reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\Main\WindowsSearch">Extraction\Analyze\ieVersion.txt
```

그림 31 Internet Explorer 정보 레지스트리 추출

[그림 31]의 명령어를 사용해 IE 정보 레지스트리를 추출해 ieVersion.txt에 작성했다.

3.3.2 USB Activity

USB활동 흔적과 관련하여 레지스트리에서 찾을 수 있는 아티팩트 정보는 PC에 Mount 된 볼륨 목록, 연결되었던 USB장치 정보, 현재 연결된 USB장치 정보가 있다.
USB Activity 관련 레지스트리 경로는 각각 아래와 같다.

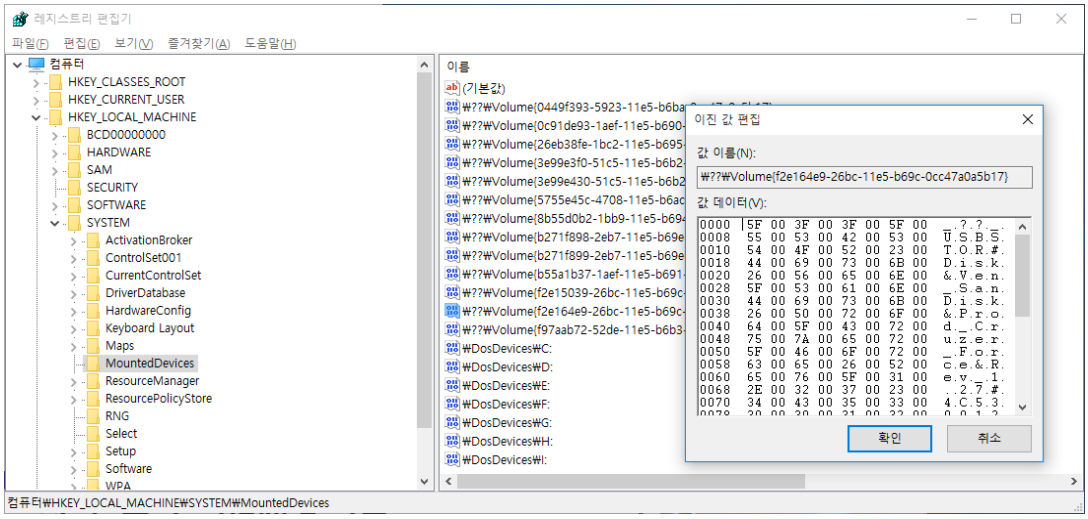

그림 32 Mount된 볼륨 목록

- Mount된 볼륨 목록 레지스트리 경로
HKEY_LOCAL_MACHINE₩SYSTEM₩MountedDevices

```
78 reg query "HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices">Extraction\Analyze\mountedDev.txt
```

그림 33 Mount된 볼륨 목록 추출

[그림 33]의 명령어를 사용해 레지스트리에 저장된 Mount된 볼륨의 목록을 추출하였으며, mountedDev.txt에 저장했다.

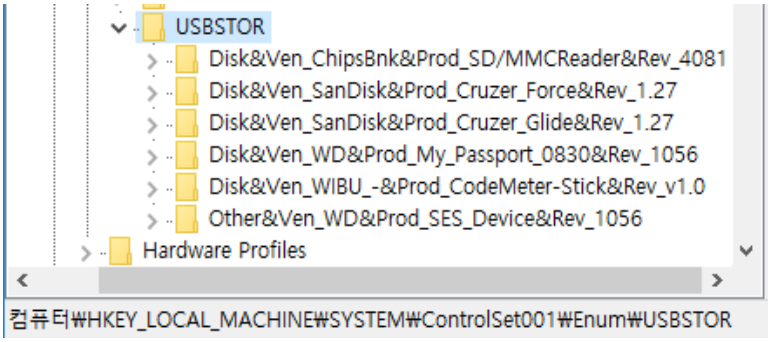

그림 34 연결되었던 USB 장치 정보

- 연결되었던 USB 장치 정보 레지스트리 경로
HKEY_LOCAL_MACHINE₩SYSTEM₩ControlSet001₩Enum₩USBSTOR

```
80  reg query "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\USBSTOR">Extraction\Analyze\usbStor.txt
```

그림 35 연결되었던 USB 장치 목록 추출

[그림 35]의 명령어를 사용해 레지스트리에 저장된 연결되었던 USB 장치 목록 기록을 usbStor.txt파일에 작성했다.

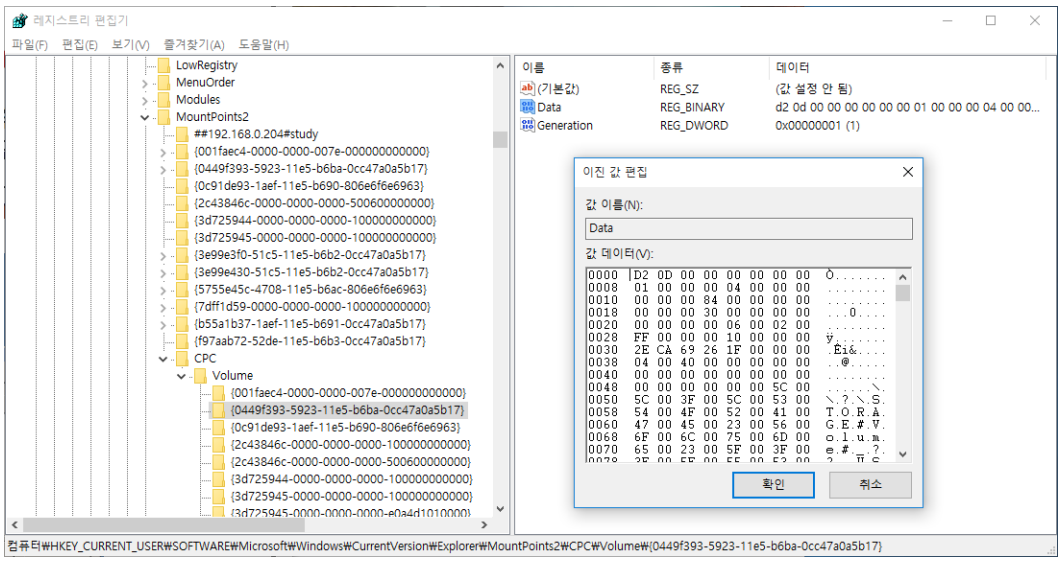

그림 36 현재 연결된 USB 장치 정보

- 현재 연결된 USB 장치 정보 레지스트리 경로
HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Windows₩CurrentVersion₩Explorer₩MountPoints2₩CPC₩Volume₩

```
82  reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\CPC\Volume">Extraction\Analyze\currConDiv.txt
```

그림 37 현재 연결된 USB 장치 정보 추출

[그림 37]의 명령어를 사용해 레지스트리에 저장된 현재 연결된 USB 장치 정보를 currConDiv.txt파일에 작성했다.

3.3.3 MRU(Most Recently Used)

MRU(Most Recently Used)란, 가장 최근에 사용된 파일 및 프로그램에 대한 정보를 레지스트리에 기록되어 있는 것을 의미한다. MRU의 종류로는 최근에 열거나 저장된 파일 목록이 기록돼있는 OpenSaveMRU, LastVisitedMRU와 파일 탐색기를 통해 최근에 연 파일 목록이 기록돼있는 RecentDocs, 실행에서 입력 및 실행된 명령어의 목록이 기록되어 있는 RunMRU가 존재한다.

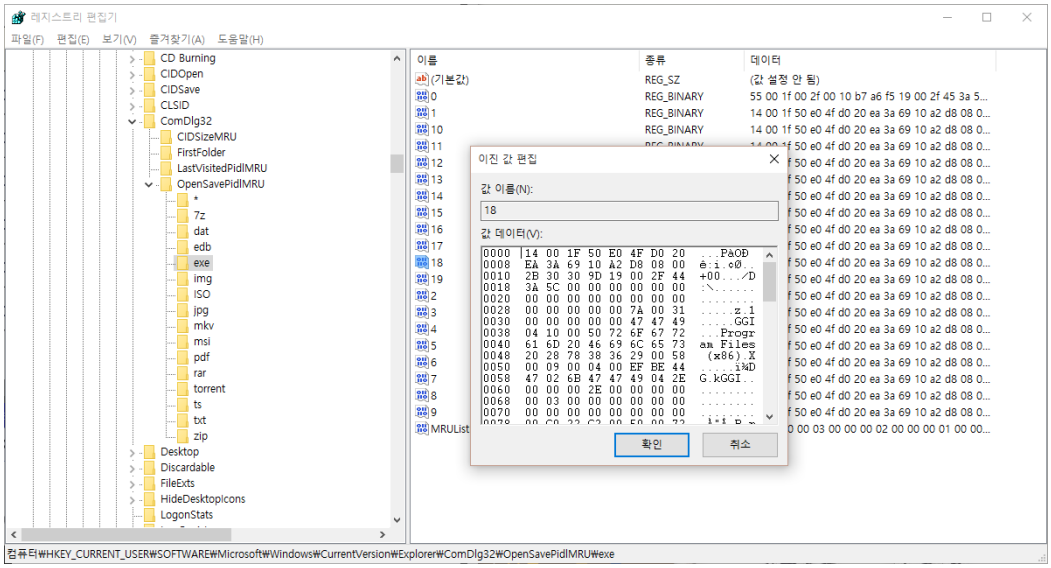

그림 38 OpenSaveMRU 레지스트리

- OpenSaveMRU 레지스트리 경로
HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Windows₩CurrentVersion₩Explorer₩ComDlg32₩OpenSavePidlMRU

```
84  reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidlMRU">Extraction\Analyze\openSaveMRU.txt
```

그림 39 OpenSaveMRU 레지스트리 추출

[그림 39]의 명령어를 사용해 레지스트리에 저장된 OpenSaveMRU 값을 openSaveMRU.txt 파일에 작성했다.

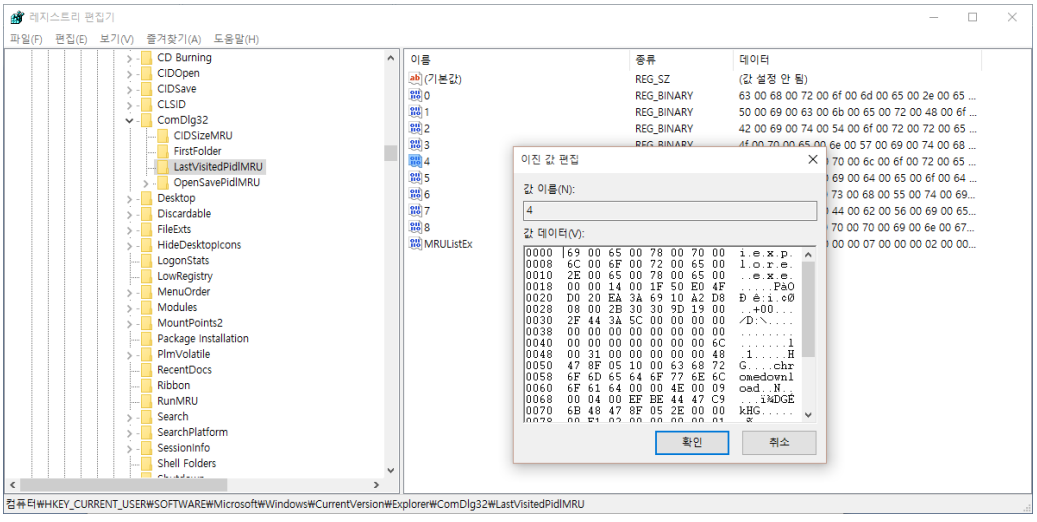

그림 40 LastVisitedMRU레지스트리

- OpenSaveMRU 레지스트리 경로
HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Windows₩CurrentVersion₩Explorer₩ComDlg32₩LastVisitedPidlMRU

OpenSaveMRU와 LastVisitedMRU 모두 파일을 접근하고 저장할 때 해당 레지스트리 경로에 순차적으로 기록된다.

```
86 reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRU">Extraction\Analyze\lastVisitedMRU.txt
```

그림 41 LastVisitedMRU 레지스트리 추출

[그림 41]의 명령어를 사용해 레지스트리에 저장된 LastVisitedMRU 값을 lastVisitedMRU.txt 파일에 작성했다.

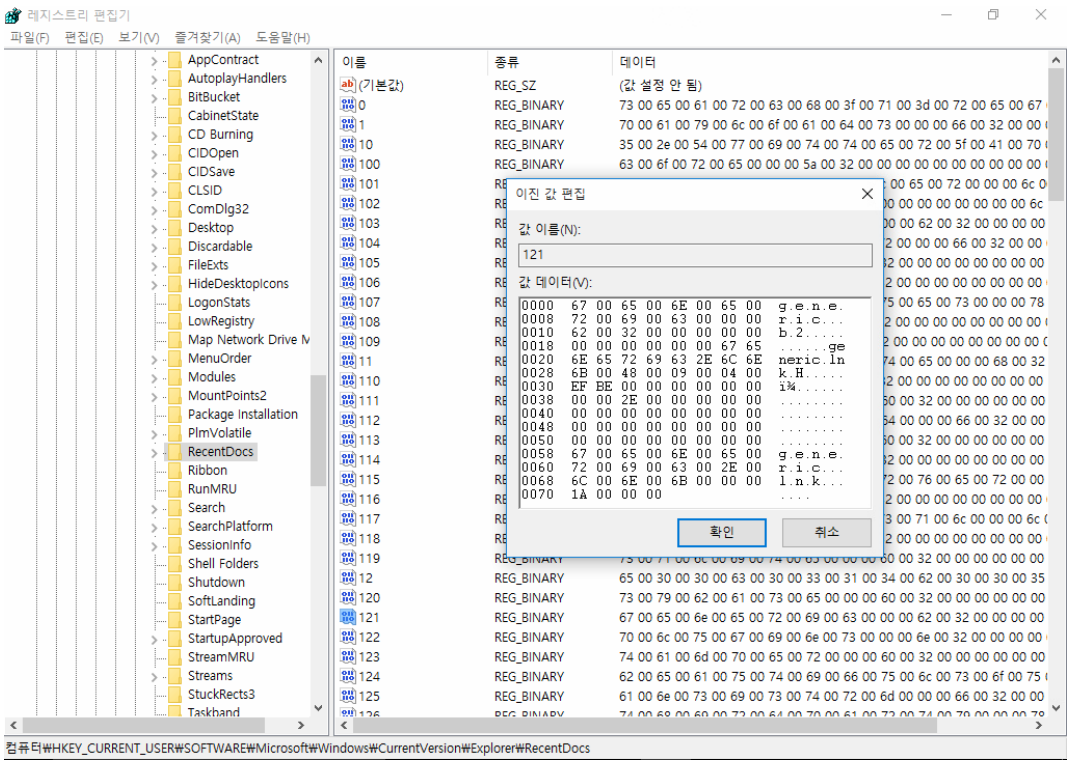

그림 42 RecentDocs 레지스트리

- RecentDocs 레지스트리 경로
  HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Windwos₩CurrentVersion₩Explorer₩RecentDocs

```
88 reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs">Extraction\Analyze\recentDocs.txt
```

그림 43 RecentDocs 레지스트리 추출

[그림 43]의 명령어를 사용해 레지스트리에 저장된 RecentDoc값을 recentDocs.txt파일에 작성하였다.

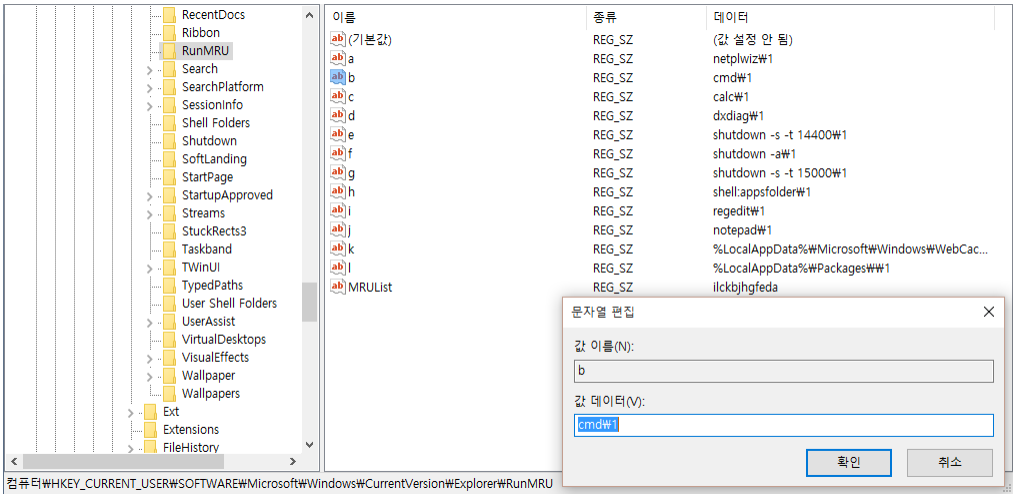

그림 44 RunMRU 레지스트리

- RunMRU 레지스트리 경로
HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Windows₩CurrentVersion₩Explorer₩RunMRU

[그림 44]에서 볼 수 있듯이, 실행창에서 실행되었던 명령어 기록이 모두 저장되어있다

```
90  reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RunMRU">Extraction\Analyze\runMRU.txt
```

그림 45 RunMRU 레지스트리 추출

[그림 45]의 명령어를 사용해 레지스트리에 저장된 RunMRU를 runMRU.txt파일로 추출했다.

3.4 메모리(Memory)

3.4.1 가상 메모리(Virtual Memory)

가상 메모리(Virtual Memory) 파일은 %SystemDrive%에 존재하며, pagefile.sys 파일과 wapfile.sys 파일이 있다. 이 파일들은 메모리상의 활성 데이터를 포함하고 있고, 메모리 관리 설정은 [그림 46]과 같이 레지스트리에 저장되어 있다.

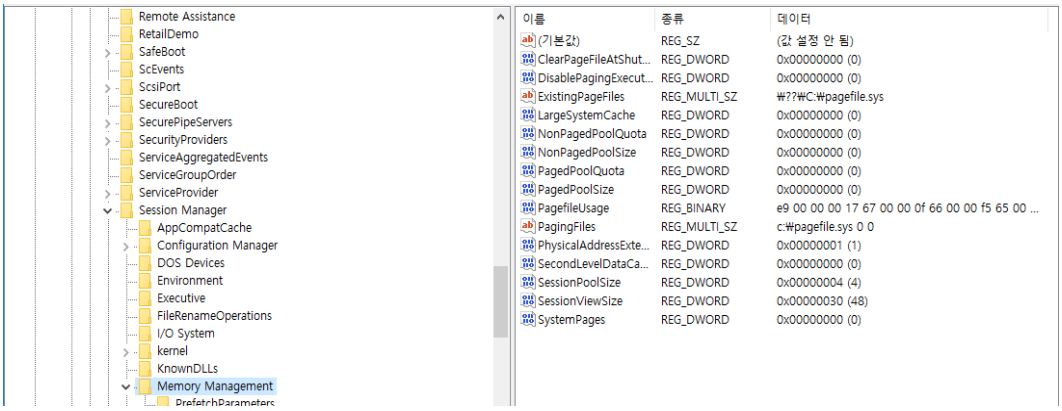

그림 46 Memory Management 레지스트리

- Memory Management 레지스트리 경로
HKEY_LOCAL_MACHINE₩SYSTEM₩CurrentControlSet₩Control₩Session Manager ₩Memory Management

3.4.2 프리패치(Prefetch)

프리패치(Prefetch) 파일은 응용 프로그램을 실행할 때 속도를 향상시키기 위해 사용되는 기술이다. 해당 파일은 %SystemRoot%₩Prefetch에 확장자명 pf로 위치한다.

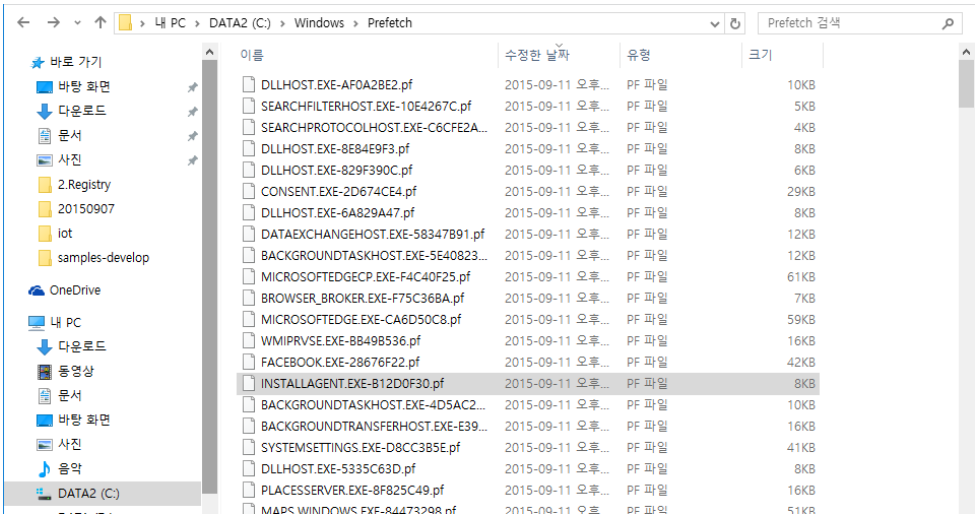

그림 47 프리패치(Prefetch) 파일

```
31  dir /a c:\windows\prefetch >prefetch.txt
```

그림 48 프리패치(Prefetch) 파일 추출

Prefetch 파일이 저장되어 있는 디렉터리의 파일 목록을 prefetch.txt 파일에 작성했다.

3.4.3 슈퍼패치(Superfetch)

슈퍼패치(Superfetch)는 프리패치(Prefetch)의 기능을 보완하기 위해 만들어진 기술이며, Superfetch 파일은 Prefetch 상태 정보를 관리한다. 즉, 슈퍼패치를 통해 Prefetch 파일 정보를 확인할 수 있다. Superfetch는 악성코드 실행 흔적을 파악할 수 있는 중요한 단서가 될 수 있다. Superfetch 파일은 [그림 49]과 같다.

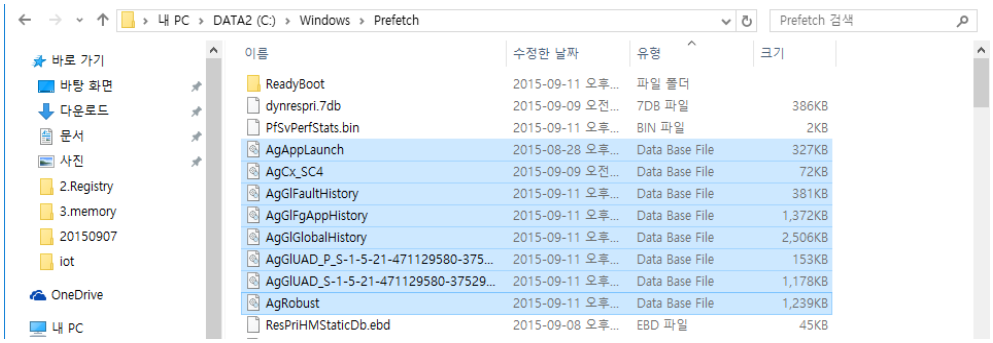

그림 49 슈퍼패치(Superfetch) 파일

```
33  dir /a c:\windows\prefetch >superfetch.txt
```

그림 50 슈퍼패치(Superfetch) 파일 추출

Superfetch 파일이 저장되어 있는 디렉터리의 파일 목록을 superfetch.txt 파일에 작성했다

3.5 윈도우 파일(Windows Files)

3.5.1 이벤트 로그(Event Log)

Windows 시스템에서 발생되는 모든 이벤트가 기록된다. 기존 Windows와 포맷 차이는 존재하지 않았다. 이벤트로그 저장경로는 C:₩Windows₩System32₩winevt₩Logs 이다.

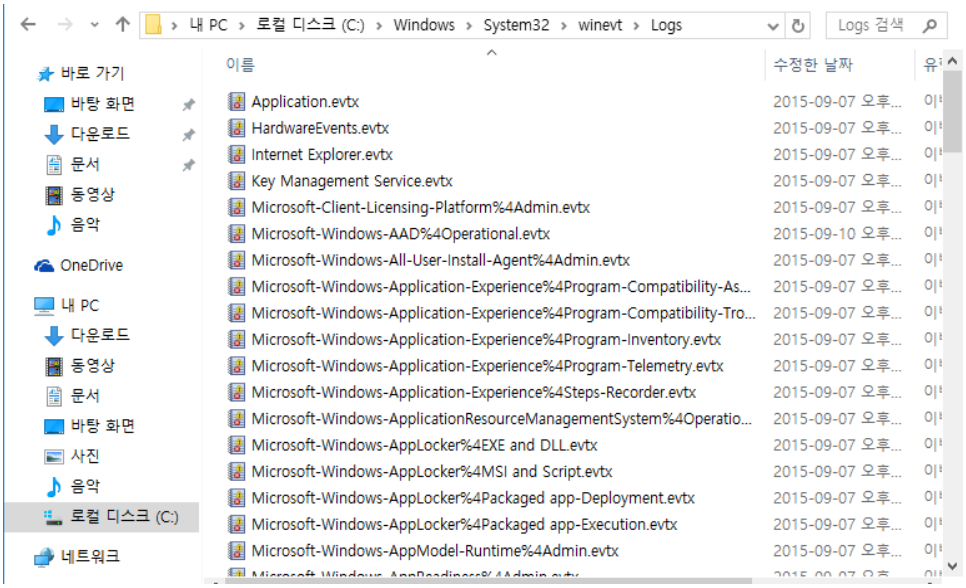

그림 51 이벤트 로그(Event Log)

```
109  dir /a C:\Windows\System32\winevt\Logs>Extraction\Analyze\eventLogs.txt
```

그림 52 이벤트 로그(Event Log) 추출

이벤트로그 저장 경로의 파일 목록을 출력하기 위하여 eventLogs.txt 파일로 작성하였다.

3.5.2 쉘백(Shellbag)

쉘백(Shellbag)은 링크 히스토리나 파일/폴더의 정보 등을 저장한다. 쉘백(Shellbag) 레지스트리 경로는 HKEY_CURRENT_USER₩SOFTWARE₩Microsoft₩Windows₩Shell₩Bags다.

그림 53 쉘백(Shellbag) 레지스트리

```
113  reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\Shell\Bags\1\Desktop">Extraction\Analyze\shellBag.txt
```

그림 54 쉘백(Shellbag) 레지스트리 추출

쉘백(Sehllbag)의 모든 레지스트리 정보를 shellBag.txt라는 이름의 파일로 추출하였다.

3.5.3 링크 파일(LNK Files)

링크 파일(LNK Files)은 Windows 시작메뉴 프로그램의 링크 파일에 해당한다. 파일 저장 경로는 C:₩ProgramData₩Microsoft₩Windows₩Start Menu₩Programs 이다.

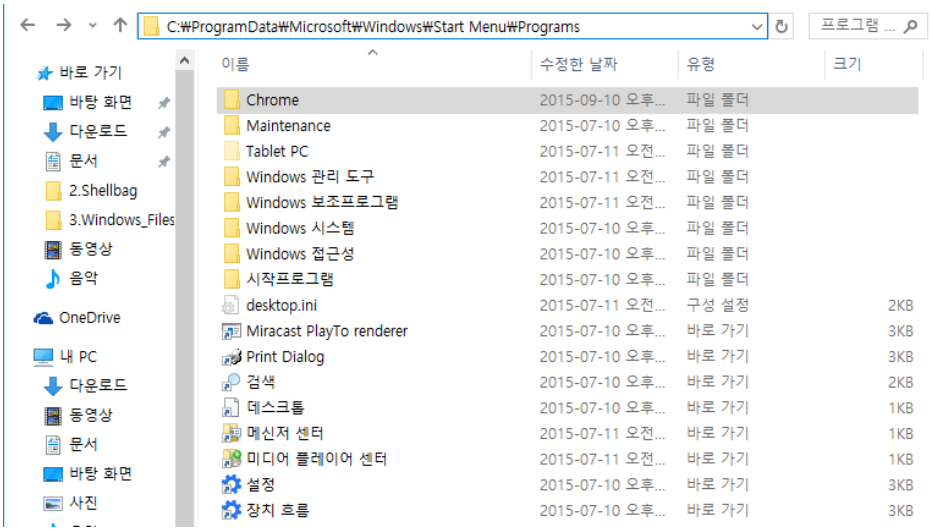

그림 55 링크 파일(LNK Files)

```
115  dir /a "C:\ProgramData\Microsoft\Windows\Start Menu\Programs">Extraction\Analyze\lnkFiles.txt
```

그림 56 링크 파일(LNK Files) 추출

링크 파일(LNK Files)이 저장돼있는 디렉터리 내부 목록을 lnkFiles.txt로 추출하였다.

3.5.4 썸네일캐시(Thumbcache)

썸네일캐시(Thumbcache)란, 사용자가 파일 탐색기를 통해 사진을 쉽게 보기 위해 사진을 작게 만들어 Windows에서 썸네일(Thumbnail) 형태로 자동 생성하는 파일이다.
썸네일캐시는 %LocalAppData%₩Microsoft₩Windows₩Explorer에 저장된다.

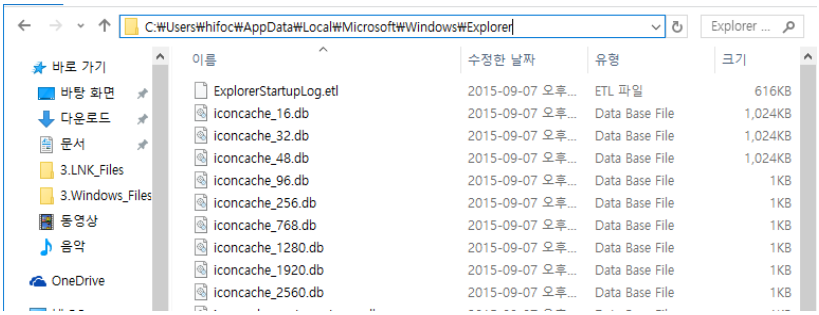

그림 57 썸네일캐시(Thumbcache)

```
117  dir /a "C:\Users\%USERNAME%\AppData\Local\Microsoft\Windows\Explorer">Extraction\Analyze\thumbCaches.txt
```

그림 58 썸네일캐시(Thumbcache) 추출

썸네일캐시(Thumbcache)가 생성되는 디렉터리의 파일목록을 추출하여 thumbCaches.txt 파일로 저장하였다.

3.5.5 휴지통(Recycle Bin)

휴지통의 경로는 C:₩Recycle.Bin₩<SID> 이다.

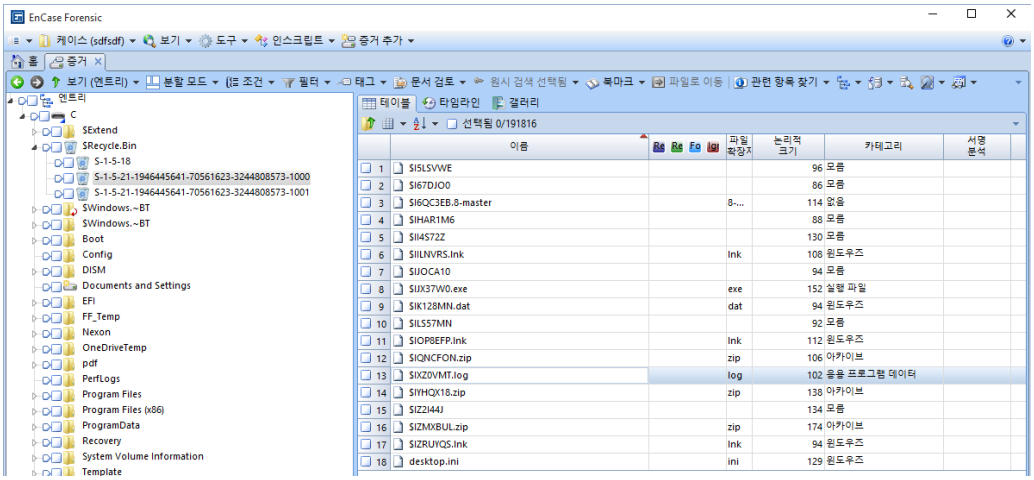

그림 59 휴지통(Recycle Bin)

```
119  dir /a C:\$Recycle.Bin\%SID% >Extraction\Analyze\reCycle.txt
```

그림 60 휴지통(Recycle Bin) 추출

휴지통에 담겨있는 모든 파일 목록을 reCycle.txt 파일로 추출하였다.

3.5.6 볼륨 섀도우 카피(Volume Shadow Copies)

볼륨 섀도우 카피(Volume Shadow Copies)는 짧은 시간에 대기 없이 복사본을 유지하는 기능으로 시스템의 복원지점 정보를 포함하고 있다.

명령 프롬프트 명령어는 vssadmin list shadows 이다.

```
C:₩WINDOWS₩system32>vssadmin list shadows
vssadmin 1.1 - 볼륨 섀도 복사본 서비스 관리 명령줄 도구
(C) Copyright 2001-2013 Microsoft Corp.

섀도 복사본 세트 ID의 콘텐츠: {dbe07a14-2bea-4e72-bb75-bf7bd11332e0}
   다음 작성 시간에 1 섀도 복사본 포함: 2015-09-11 오전 3:34:05
      섀도 복사본 ID: {076d9762-3621-4653-b972-277bf8b11843}
         원본 볼륨: (C:)₩₩?₩Volume{3d725945-0000-0000-0000-100000000000}₩
         섀도 복사본 볼륨: ₩₩?₩GLOBALROOT₩Device₩HarddiskVolumeShadowCopy13
         원본 컴퓨터: WIN-HT34JTKJUKI
         서비스 컴퓨터: WIN-HT34JTKJUKI
         공급자: 'Microsoft Software Shadow Copy provider 1.0'
         형식: ClientAccessibleWriters
         특성: Persistent, Client-accessible, No auto release, Differential, 자동 복구됨
```

그림 61 볼륨 섀도우 카피(Volume Shadow Copies)

3.5.7 윈도우 인덱싱 서비스(Windows Indexing Service)

윈도우 인덱싱 서비스(Windows Indexing Service)는 윈도우 검색(Windows Search)에 사용하기 위한 색인 정보가 저장되어 있다.해당 내용은 Windows.edb 파일에 저장되며, ESEDatabaseView와 같은 도구를 이용하여 열람할수 있다.

저장경로는 %ProgramData%₩Microsoft₩Search₩Data₩Applications₩Windows이다.

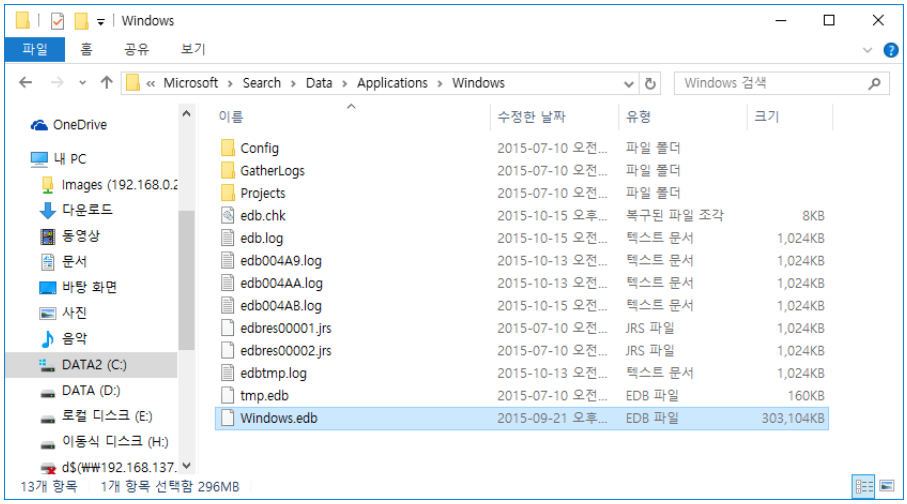

그림 62 윈도우 인덱싱 서비스(Windows Indexing Service)

```
121  dir /a "C:\ProgramData\Microsoft\Search\Data\Applications\Windows">Extraction\Analyze\WIS.txt
```

그림 63 윈도우 인덱싱 서비스(Windows Indexing Service) 추출

윈도우 인덱싱 서비스(Windows Indexing Service) 정보가 저장돼있는 Windows.edb 파일 존재 여부 확인을 위하여 디렉터리 파일 목록을 WIS.txt 파일에 추출하였다.

3.5.8 코타나(Cortana)

코타나(Cortana)는 Windows 10에서 새롭게 추가된 기능으로서, 음성 명령으로 조작 가능한 음성인식 개인비서 서비스다. 현재 한국은 코타나 서비스 지원 대상 국가에서 제외되어 기능이 비활성화 되어있어 Windows10 영문판으로 아티팩트 분석을 진행하였다. 사용자가 코타나를 통해 검색한 결과, 검색한 장소의 위도 및 경도 등 여러 아티팩트가 [그림 64, 66]과 같이 아래의 경로에 각각 저장된다.

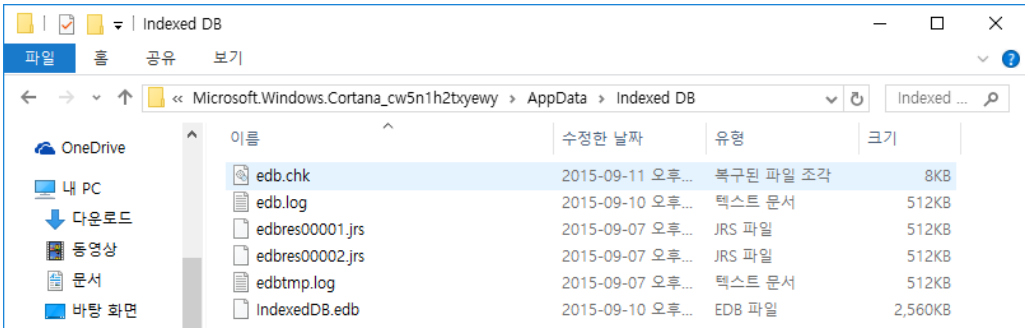

그림 64 코타나(Cortana) Indexed DB

- Cortana Indexed DB저장 경로
  %LocalAppData%₩Packages₩Microsoft.Windows.Cortana_cw5n1h2txyewy₩AppData₩Indexed DB

```
125 dir /a "C:\Users\%USERNAME%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\AppData\Indexed DB">Extraction\Analyze\indexDB.txt
```

그림 65 코타나(Cortana) Indexed DB 추출

Cortana의 IndexedDB 파일 존재여부 확인을 위해 해당 디렉터리 파일 목록을 indexDB.txt 파일에 추출하여 저장하였다.

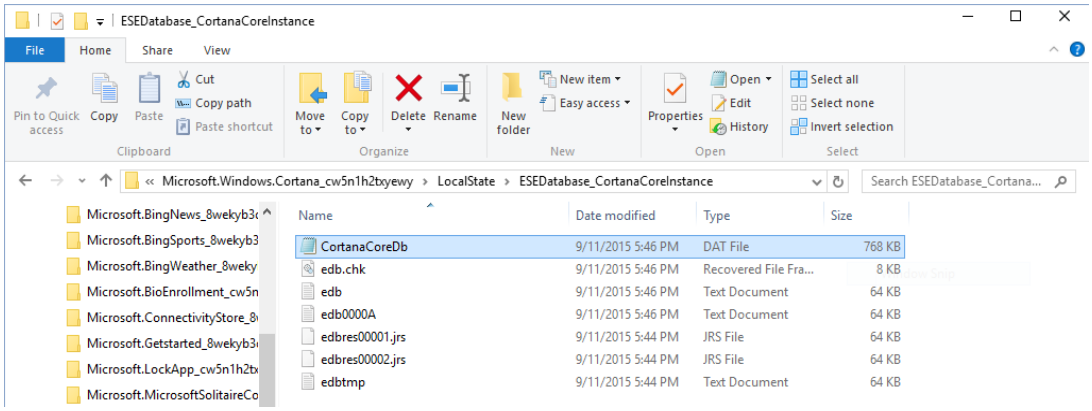

그림 66 코타나(Cortana) Core DB

- Cortana CoreDB저장 경로
  %LocalAppData%₩Packages₩Microsoft.Windows.Cortana_cw5n1h2txyewy₩LocalState₩ESEDatabase_CortanaCoreInstance₩CortanaCoreDb

```
123 dir /a "C:\Users\%USERNAME%\AppData\Local\Packages\Microsoft.Windows.Cortana_cw5n1h2txyewy\LocalState\ESEDatabase_CortanaCoreInstance\CortanaCoreDb">Extraction\Analyze\coreDB.txt
```

그림 67 코타나(Cortana) Core DB 추출

Cortana의 Core DB 파일 또한 시스템에 존재하는지 여부를 확인하기 위해 디렉터리 내부 파일 목록을 coreDB.txt 파일에 추출하였다.

3.5.9 알림 센터(Notification Center)

알림 센터(Notification Center)는 Windows에 설치된 어플리케이션 및 시스템 설정으로부터 발생되는 알림을 보관하는 기능을 한다. 알림 항목이 포함된 데이터베이스가 appdb.dat 라는 파일로 [그림 68]과 같이 저장되어 있다.

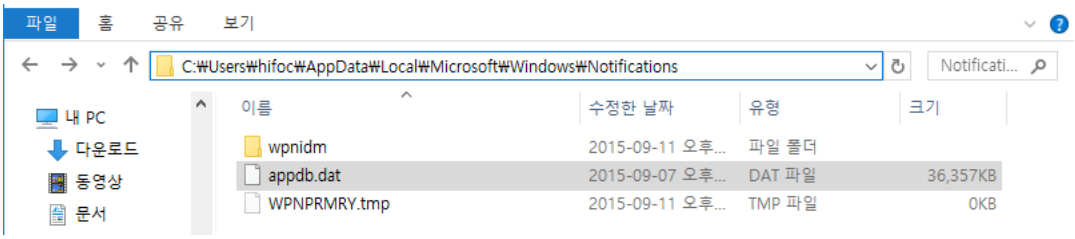

그림 68 알림 센터(Notification Center) appdb.dat파일

- 알림 센터(Notification Center) 아티팩트 저장 경로
  %LocalAppData%₩Microsoft₩Windows₩Notifications

```
127 dir /a "C:\Users\%USERNAME%\AppData\Local\Microsoft\Windows\Notifications">Extraction\Analyze\NotiCen.txt
```

그림 69 알림 센터(Notification Center) 아티팩트 추출

알림 센터(Notification Center)의 정보가 기록되는 appdb.dat파일 확인을 위해 디렉터리 내부 목록을 NotiCen.txt 파일로 저장하였다.

3.5.10 사진 암호(Picture Password)

사진 암호(Picture Password)는 사진을 통해 사전에 사용자가 입력한 제스처(Gesture)로 Windows 암호를 해제하는 기능이다. 사진 암호를 설정하면, 설정된 암호 파일의 경로가 레지스트리에 남고, 시스템에 암호로 사용되는 사진 파일이 저장된다.

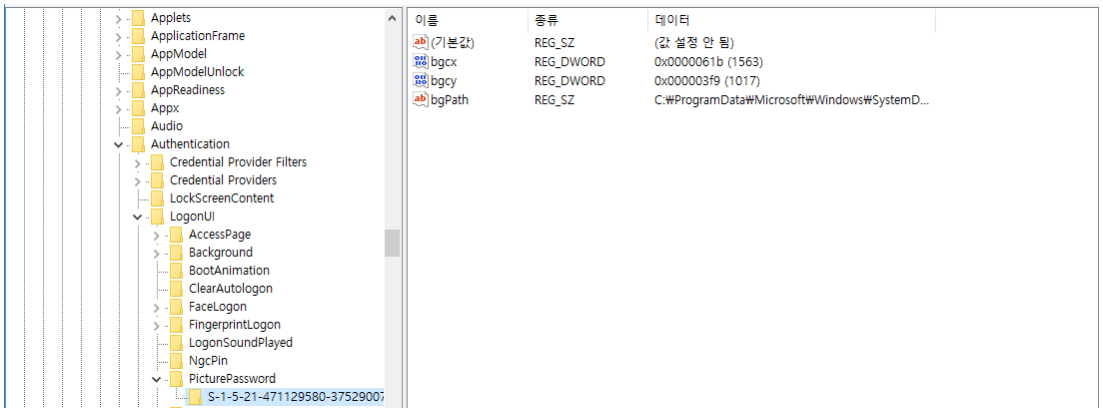

그림 70 사진 암호(Picture Password) 레지스트리

- 사진 암호(Picture Password) 레지스트리 경로
  HKEY_LOCAL_MACHINE₩Software₩Microsoft₩Windows₩CurrentVersion₩Authentication₩LogonUI₩PicturePassword₩<user_GUID>

```
129 reg query "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Authentication\LogonUI\PicturePassword\%SID%">Extraction\Analyze\picpassReg.txt
```

그림 71 사진 암호(Picture Password) 레지스트리 추출

사진 암호 설정을 통해 시스템에 남은 레지스트리 정보를 picpassReg.txt 파일에 추출했다.

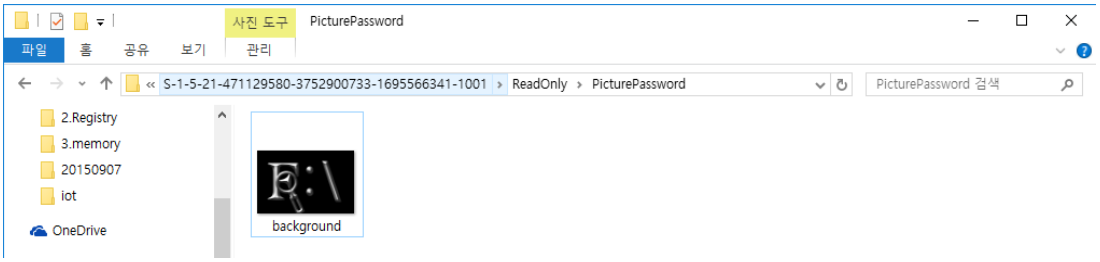

그림 72 사진 암호(Picture Password) 원본 사진파일

- 사진 암호(Picture Password)원본 사진파일 저장 경로
  C:₩ProgramData₩Microsoft₩Windows₩SystemData₩user_GUID₩ReadOnly₩Picture Password₩background.png

```
131 dir /a "C:\ProgramData\Microsoft\Windows\SystemData\%SID%\ReadOnly\PicturePassword\">Extraction\Analyze\picpassImg.txt
```

그림 73 사진 암호(Picture Password) 원본 사진파일 추출

사진 암호를 설정해서 시스템에 저장된 원본 사진파일의 존재 유무 정보를 picpassImg.txt 파일로 추출했다.

4. 프로젝트 결론

아래 그림은 모두 분석한 아티팩트를 항목별로 분류하여 각각 하나의 txt 파일로 문자열을 추출하도록 명령하는 Batch 파일을 작성한 후, 이를 병합시켜 HTML 파일로 결과가 도출되는 스크립트 코드이다.

```
REM ========================================
REM             Registry
REM ========================================

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\TypedURLs"> Extraction\Analyze\typedURL.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\Main\WindowsSearch">Extraction\Analyze\ieVersion.txt

reg query "HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices">Extraction\Analyze\mountedDev.txt

reg query "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\USBSTOR">Extraction\Analyze\usbStor.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\CPC\Volume">Extraction\Analyze\currConDiv.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidlMRU">Extraction\Analyze\openSaveMRU.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRU">Extraction\Analyze\lastVisitedMRU.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs">Extraction\Analyze\recentDocs.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RunMRU">Extraction\Analyze\runMRU.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\CIDSizeMRU">Extraction\Analyze\cidSizeMRU.txt

reg query "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Currentversion\Run">Extraction\Analyze\run.txt

reg query "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Currentversion\RunOnce">Extraction\Analyze\runOnce.txt


REM ========================================
REM          Maps Application
REM ========================================

dir /a "C:\Users\%USERNAME%\AppData\Local\Packages\Microsoft.WindowsMaps_8wekyb3d8bbwe\LocalState\Graph\11d845678c3f9b5e">Extraction\Analyze\maps.txt

REM ========================================
REM          Windows Files
REM ========================================

dir /a C:\Windows\System32\winevt\Logs>Extraction\Analyze\eventLogs.txt

dir /a C:\Users\%USERNAME%\AppData\Local\Microsoft\Windows>Extraction\Analyze\userClass.txt

reg query "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\Shell\Bags\1\Desktop">Extraction\Analyze\shellBag.txt
```

그림 74 각 정보 텍스트 파일 추출

```
REM ====================================
REM          HTML        Maker
REM ====================================
chcp 1252
cd Extraction
echo ^<!DOCTYPE html^> >index.html
echo ^<html lang="en"^> >>index.html
echo.>>index.html
echo ^<head^> >>index.html
echo.>>index.html
echo ^<title^>Hoseo Artifact  Viewer^</title^>
echo     ^<meta charset="utf-8"^> >>index.html
echo     ^<meta http-equiv="X-UA-Compatible" content="IE=edge"^> >>index.html
echo     ^<meta name="viewport" content="width=device-width, initial-scale=1"^> >>index.html
echo     ^<meta name="description" content=""^> >>index.html
echo     ^<meta name="author" content=""^> >>index.html
echo.>>index.html
echo     ^<title^>SB Admin - Bootstrap Admin Template^</title^> >>index.html
echo.>>index.html
echo     ^<link href="css/bootstrap.min.css" rel="stylesheet"^> >>index.html
echo. >>index.html
echo     ^<link href="css/sb-admin.css" rel="stylesheet"^> >>index.html
echo. >>index.html
echo     ^<link href="css/plugins/morris.css" rel="stylesheet"^> >>index.html
echo. >>index.html
echo     ^<link href="font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css"^> >>index.html
echo. >>index.html
echo     ^<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries --^>>>index.html
echo    ^<!-- WARNING: Respond.js doesn't work if you view the page via file:// --^>>>index.html
echo     ^<!--[if lt IE 9]^>>>index.html
echo         ^<script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"^>^</script^> >>index.html>>index.html
echo         ^<script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"^>^</script^> >>index.html>>index.html
echo     ^<![endif]--^>>>index.html
echo.>>index.html
echo ^</head^> >>index.html
echo.>>index.html
echo ^<body^> >>index.html
echo.>>index.html
```

그림 75 index.html 작성

```
REM ==============================
REM              Source
REM ==============================
echo ^<!DOCTYPE html^> >source.html
echo ^<html lang="en"^>>>source.html
echo.>>source.html
echo ^<head^>>>source.html
echo.>>source.html
echo ^<title^>Hoseo Information Forensic Artifact Viewer^</title^> >>source.html
echo    ^<meta charset="utf-8"^>>>source.html
echo    ^<meta http-equiv="X-UA-Compatible" content="IE=edge"^>>>source.html
echo    ^<meta name="viewport" content="width=device-width, initial-scale=1"^>>>source.html
echo    ^<meta name="description" content=""^>>>source.html
echo    ^<meta name="author" content=""^>>>source.html
echo.>>source.html
echo    ^<title^>SB Admin - Bootstrap Admin Template^</title^>>>source.html
echo    ^<link href="css/bootstrap.min.css" rel="stylesheet"^>>>source.html
echo.>>source.html
echo    ^<link href="css/sb-admin.css" rel="stylesheet"^>>>source.html
echo.>>source.html
echo    ^<link href="css/plugins/morris.css" rel="stylesheet"^>>>source.html
echo.>>source.html
echo    ^<link href="font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css"^>>>source.html
echo.>>source.html
echo    ^<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries --^>
echo    ^<!-- WARNING: Respond.js doesn't work if you view the page via file:// --^>
echo    ^<!--[if lt IE 9]^>
echo        ^<script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"^>^</script^>>>source.html
echo        ^<script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"^>^</script^>>>source.html
echo    ^<![endif]--^>
echo.>>source.html
echo.>>source.html
echo     ^</style^>>>source.html
echo ^</head^>>>source.html
echo.>>source.html
echo ^<body style="overflow=hidden"^>>>source.html
echo.>>source.html
echo    ^<div id="wrapper"^>>>source.html
echo.>>source.html
```

그림 76 source.html 작성

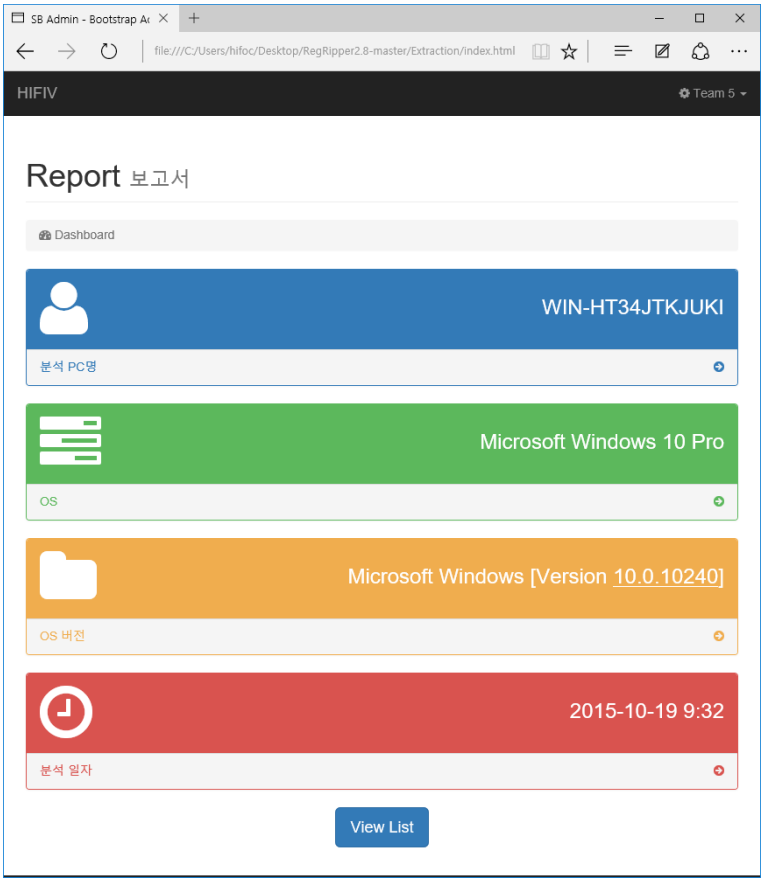

그림 77 보고서 메인 페이지

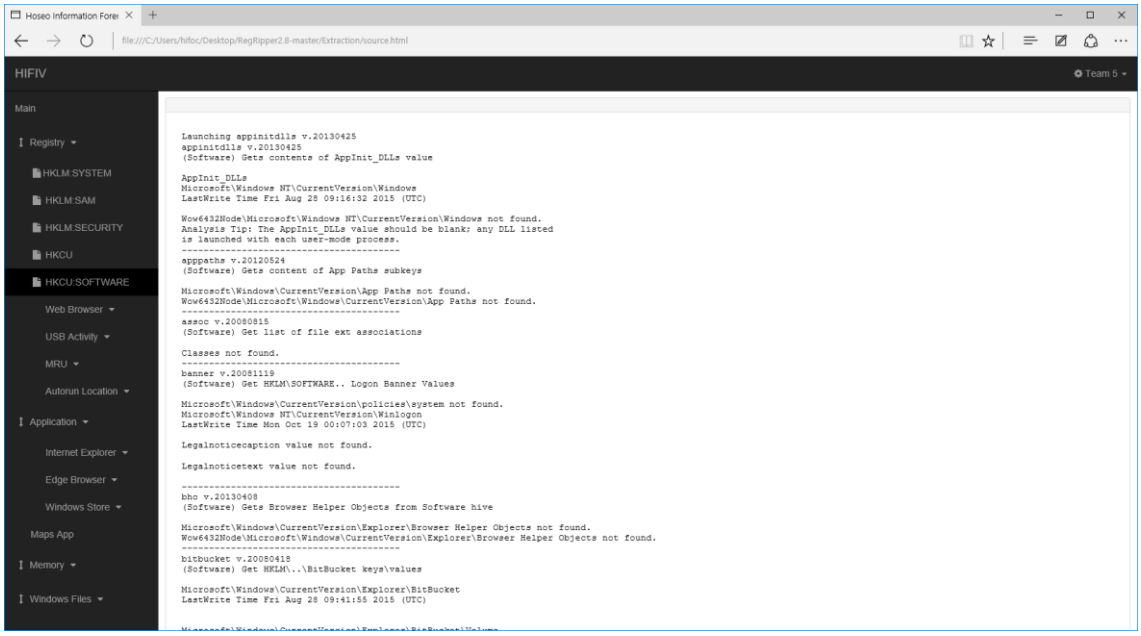

그림 78 보고서 내용 페이지

Batch 파일 실행 후, 생성된 보고서 파일이다. 웹 형태의 HTML 파일로 보고서가 출력된다. 부트스트랩을 통해 UI 를 작성했으며, 모든 아티팩트를 항목화시켜 포렌식 조사관이 필요한 정보만 확인할 수 있도록 배치하였다.

이 프로젝트는 Windows 10 아티팩트 분석도구(H-AAT)를 사용해서 포렌식 조사관이 현재 보급된 지 얼마 되지 않은 Windows 10 을 실제 분석 시에 운용하여, 실시간 대응에 도움이 되는 것에 목표를 두고 진행되었다.

때문에, 이 도구는 제작자의 블로그(http://protorn.tistory.com/)를 통해 오픈소스로 배포돼 Windows 10 아티팩트 정보 수집을 필요로 하는 많은 조사관에게 널리 공유되고, Windows 업데이트를 통해 추가로 발생하는 신규 아티팩트 정보 또한 Tool 에 지속적으로 업데이트를 적용하는 것을 최종 목표로 두고 프로젝트를 마쳤다.

5. 참고문헌

[1] 한국인터넷진흥원(KISA) - 신규 운영체제 환경에서의 Forensic 기법 연구

[2] 경찰대학 사이버범죄 연구회 - A Forensic Analysis of the Windows Registry

[3] 안랩 보안 매거진 월간 安 2012 년 8 월호 – 윈도우 8 포렌식

[4] Leahy Center for Digital Investigation – Windows 10 Forensics

[5] 한국 마이크로소프트 - http://www.microsoft.com/ko-kr/

[5] 포렌식 프루프 - http://forensic-proof.com/

[6] Forensic Focus - http://articles.forensicfocus.com/

[7] CHAMPLAIN COLLEGE - http://computerforensicsblog.champlain.edu/

팀원 소개

이　름 : 김 유 철
전　공 : 포렌식
E-Mail : ainpc@naver.com

졸업 프로젝트 역할 : Regripper 커스터마이징, 자료 조사

이　름 : 성 중 곤
전　공 : 포렌식
E-Mail : pooh910125@naver.com

졸업 프로젝트 역할 : 총괄, 아티팩트 분석도구 제작

이　름 : 홍 수 빈
전　공 : 포렌식
E-Mail : protornict@gmail.com

졸업 프로젝트 역할 : 환경 구성, 아티팩트 분석 및 문서화