Deep in the artifacts

MaJ3stY

saiwnsgud@gmail.com

http://maj3sty.tistory.com

Rather be dead than cool.

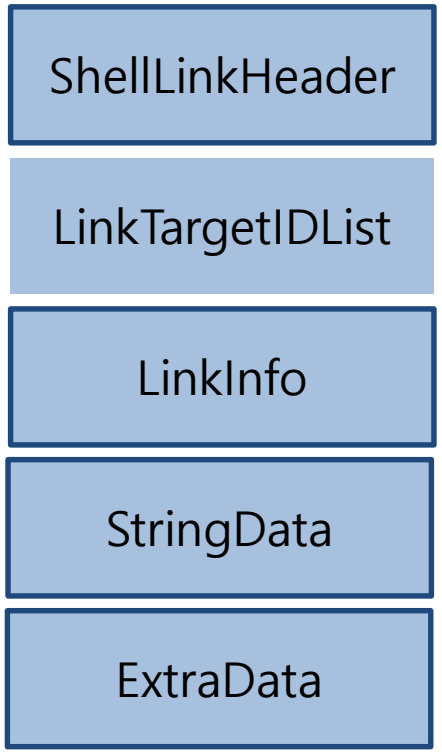
Deep in the artifacts

1. Lnk & Jumplist
2. Volume Shadow Copy
3. Shellbag
4. Prefetch

Lnk & Jumplist

Lnk & Jumplist

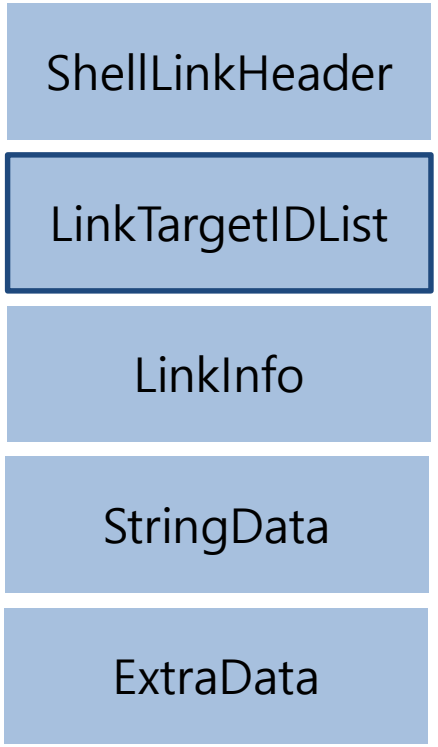
- **기존 포렌식 관점**
  - Lnk 파일 생성/수정/접근 시간
  - 원본 파일 생성/수정/접근 시간
  - 볼륨 시리얼 번호
  - 볼륨명
  - 원본 파일 경로
  - DestList 마지막 수정 시간 (Jumplist)



Lnk & Jumplist

- **새로운 포렌식 관점**
  - Target**New**DroidVolumeID
  - Target**Birth**DroidVolumeID
  - Target**New**DroidFileID
  - Target**Birth**DroidFileID

- ID의 형태는 UUID(**U**niversally **U**nique **ID**entifier)



Lnk & Jumplist

- **UUID (U**niversally **U**nique **ID**entifier**)**
  - 네트워크 상에서 서로 모르는 개체를 식별하기 위한 고유 식별자
  - 128bit로 구성, 표현은 32개의 hex value로 이뤄짐
    - ✓ 8-4-4-4-12 구조
    - ✓ Ex) 550e8400-e29b-41d4-a716-446655440000
  - 버전은 총 5가지
    - ✓ **Version 1 : MAC Address + Timestamp(millisecond)**
    - ✓ Version 2 : Timestamp 4bytes
    - ✓ Version 3 : MD5(URL)
    - ✓ Version 4 : Random
    - ✓ Version 5 : SHA1(URL)

Lnk & Jumplist

- **Target VolumeID**
  - VolumeID는 볼륨이 만들어질 때 생성
  - Volume 별로 고유한 값을 가지고 있음
  - VolumeID는 $Volume의 $MFT Entry($OBJECT_ID)에 기록 되어 있음

- **Target FileID**
  - FileID는 파일이 최초 **실행**될 때 생성 ➔ 원본 파일의 최초 실행시간
  - File 별로 고유한 값을 가지고 있음
  - FileID는 파일 별 $MFT Entry($OBJECT_ID)에 기록 되어 있음

- VolumeID와 FileID는 UUID(Version 1)을 사용 함

Lnk & Jumplist

- **#1 파일 이동**
  - C:₩ → K:₩ 볼륨으로 파일을 잘라내었을 경우

항목	원본 파일	잘라내기된 파일
파일명	TestCross.egg	
TargetNewDroidFileID	2ac7e360-f963-11e6-aeca-005056c00001	2ac7e360-f963-11e6-aeca-005056c00001
TargetBirthDroidFileID	2ac7e360-f963-11e6-aeca-005056c00001	2ac7e360-f963-11e6-aeca-005056c00001
TargetNewDroidVolumeID	0109db74-26fe-4888-83d4-355f14c87f02	**b60e70b6-568b-41e8-9acb-72d04750f88d**
TargetBirthDroidVolumeID	0109db74-26fe-4888-83d4-355f14c87f02	**0109db74-26fe-4888-83d4-355f14c87f02**

  - 잘라내기 되어진 파일의 **TargetNewDroidVolumeID**와 **TargetBirthDroidVolumeID**가 다름

Lnk & Jumplist

- **#2 파일 복사**
  - C:₩ → K:₩ 볼륨으로 파일을 복사했을 경우

항목	원본 파일	복사된 파일
파일명	TestCross.egg	TestCross2.egg
TargetNewDroidFileID	2ac7e360-f963-11e6-aeca-005056c00001	3ac32460-f444-11d6-abca-038386c34221
TargetBirthDroidFileID	2ac7e360-f963-11e6-aeca-005056c00001	2ac7e360-f963-11e6-aeca-005056c00001
TargetNewDroidVolumeID	0109db74-26fe-4888-83d4-355f14c87f02	b60e70b6-568b-41e8-9acb-72d04750f88d
TargetBirthDroidVolumeID	0109db74-26fe-4888-83d4-355f14c87f02	0109db74-26fe-4888-83d4-355f14c87f02

  - 잘라내기 되어진 파일의 **TargetNewDroidVolumeID**와 **TargetBirthDroidVolumeID**가 다름

Lnk & Jumplist

- **#3 파일의 이전 경로 추적**
  - C:₩ 볼륨의 파일을 다른 볼륨으로 잘라내기 했을 경우

볼륨명	TargetNewDroidVolumeID	TargetBirthDroidVolumeID
C:₩	0109db74-26fe-4888-83d4-355f14c87f02	0109db74-26fe-4888-83d4-355f14c87f02
E:₩	255ea846-a079-4b19-b814-0483bafdd2f1	255ea846-a079-4b19-b814-0483bafdd2f1
K:₩	**b60e70b6-568b-41e8-9acb-72d04750f88d**	**b60e70b6-568b-41e8-9acb-72d04750f88d**
H:₩file.txt	255ea846-a079-4b19-b814-0483bafdd2f1	**b60e70b6-568b-41e8-9acb-72d04750f88d**

  - 잘라낸 파일의 **TargetBirthDroidVolumeID**는 다른 파일들의 **TargetBirthDroidVolumeID** 또는 **TargetNewDroidVolumeID**와 동일 함
  - 특정 파일의 ID를 다른 파일의 ID와 비교하면 이전에 파일이 어떤 볼륨에 있었는지 확인 가능

Lnk & Jumplist

- **#4 파일 생성 순서**
  - ID는 모두 UUID(Version 1)를 사용하고 있어, 시간 순으로 UUID 값이 증가 함

파일명	TargetNewDroidFileID	TargetBirthDroidFileID
C:₩TestFile.txt	2ac7e378-f963-11e6-aeca-005056c00001	2ac7e378-f963-11e6-aeca-005056c00001
E:₩TestCross.egg	2ac7e391-f963-11e6-aeca-005056c00001	2ac7e391-f963-11e6-aeca-005056c00001
K:₩TTTTEEEESSSTTT.docx	2ac7e3ab-f963-11e6-aeca-005056c00001	2ac7e3ab-f963-11e6-aeca-005056c00001

  - Millisecond 이므로 동일한 시간 대에 생성된 파일도 정렬 가능

Lnk & Jumplist

- **결론**
  - #1 방법은 **분산 추적 시스템(Distributed Link Tracking System)**에서 활용
  - #1 방법을 통해 CrossVolumeMoveFlag 활성
    - ✓ 확인 : fsutil objected query <File Path>

항목	원본 파일	잘라내기된 파일
파일명	TestCross.egg	
Object ID	a395aa22-0102-e711-9203-005056c00001	a395aa22-0102-e711-9203-005056c00001
BirthVolume ID	74db0901-fe26-8848-83d4-355f14c87f02	75db0901-fe26-8848-83d4-355f14c87f02
BirthObjectid ID	a395aa22-0102-e711-9203-005056c00001	a395aa22-0102-e711-9203-005056c00001
Domain ID	00000000-0000-0000-0000-000000000000	00000000-0000-0000-0000-000000000000

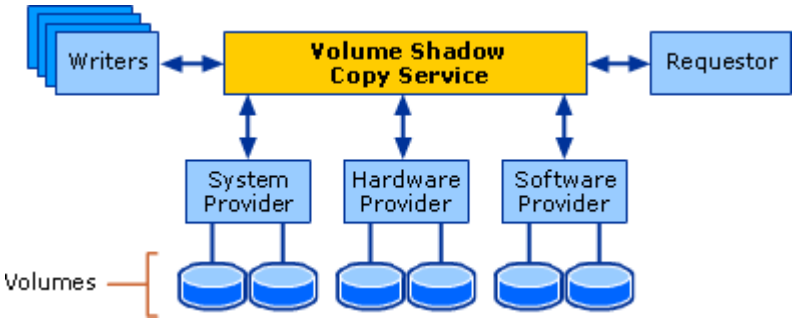
  - 복사된 파일은 서로 다른 파일로 간주

Volume Shadow Copy

Volume Shadow Copy

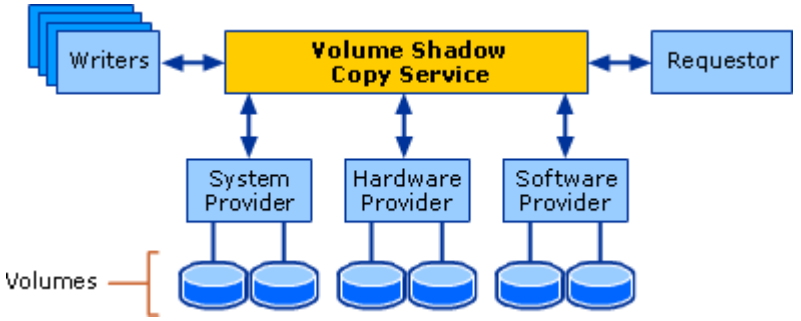
- **기존 포렌식 관점**
  - 이전 파일 복원
  - 스냅샷 시점의 운영체제 파일 분석

- **새로운 포렌식 관점**
  - 파일시스템 메타데이터 파일 분석
  - 비할당 영역에서의 파일 복구



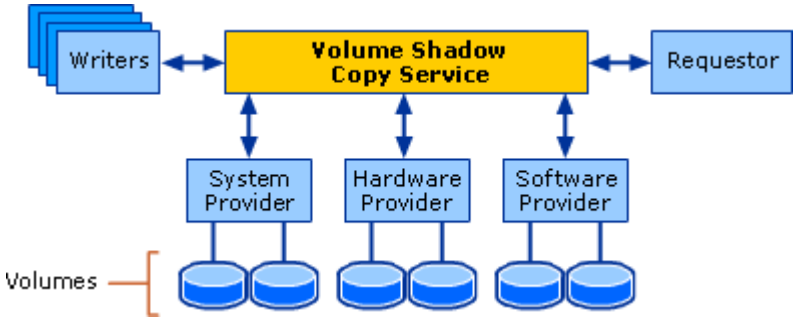
https://technet.microsoft.com/en-us/library/cc785914(v=ws.10).aspx

Volume Shadow Copy



- **용어 정리**
  - **Requestors**
    - ✓ Shadow Copy 생성을 VSC Service에게 요청하는 요소
    - ✓ 백업 대상 정보를 수집
  - **Writers**
    - ✓ 디스크 데이터의 불일치성을 줄이기 위해 존재
    - ✓ Volume Shadow Copies(VSC)가 생성될 동안 기타 쓰기 작업이 수행되지 않도록 I/O를 관리
    - ✓ 이미 진행 중이었던 쓰기 작업을 마무리하는 기능을 가짐(디스크 버퍼 Flush) : **데이터 일관성**
    - ✓ 복원 방법을 정의
      - 복원 제외 파일 정의, 아이콘 포함 등

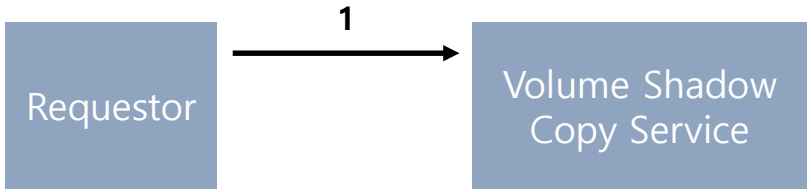
Volume Shadow Copy



- **용어 정리**
  - **Provider**
    - ✓ VSC를 생성하는 요소
    - ✓ VSC 시작 시점과 종료 시점 사이에 VSC 유지 및 수정을 위해 Volume Shadow Copy Service에 지속적으로 신호를 전달 함
    - ✓ Hardware-based Provider
    - ✓ Software-based Provider
    - ✓ System Provider

Volume Shadow Copy

- **VSC 동작 원리**



- 1. Volume Shadow Copy Service에게 섀도 복사본 작성을 준비하도록 요청
  - ✓ Writer의 메타 데이터를 수집하고 열거하여 서비스에게 알림

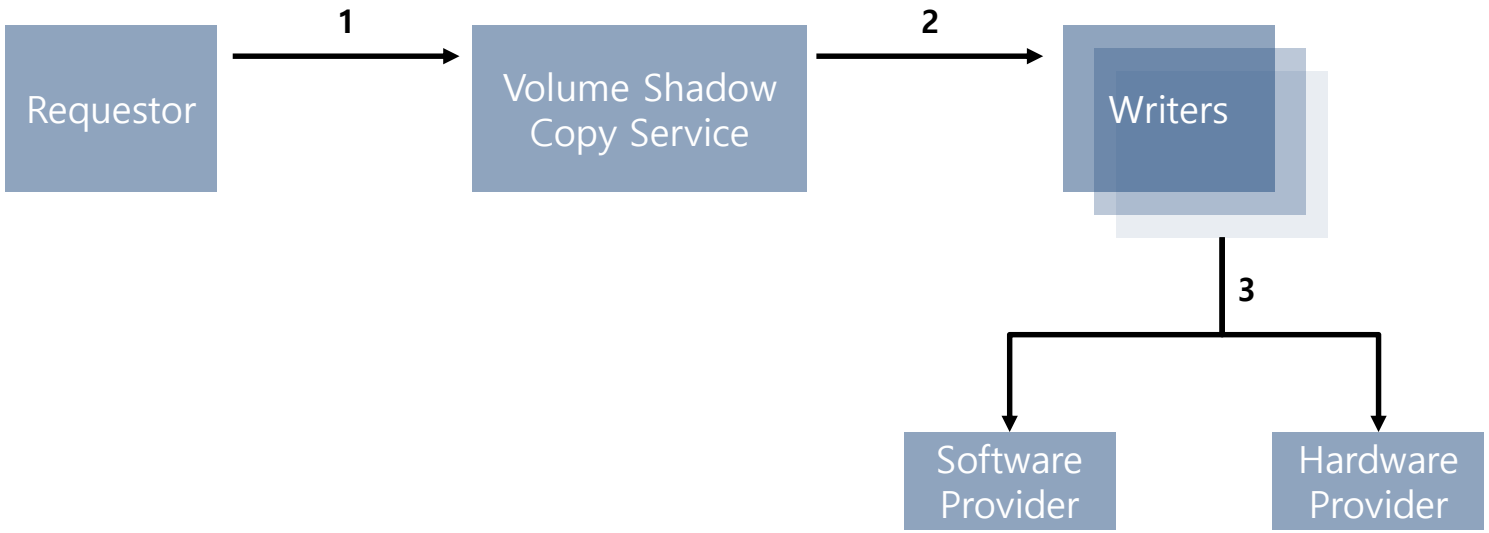
Volume Shadow Copy

- **VSC 동작 원리**

- 2. Volume Shadow Copy Service는 Writer를 준비 함
  - ✓ 백업 구성 요소가 정의된 XML을 Writers에게 전달

Volume Shadow Copy

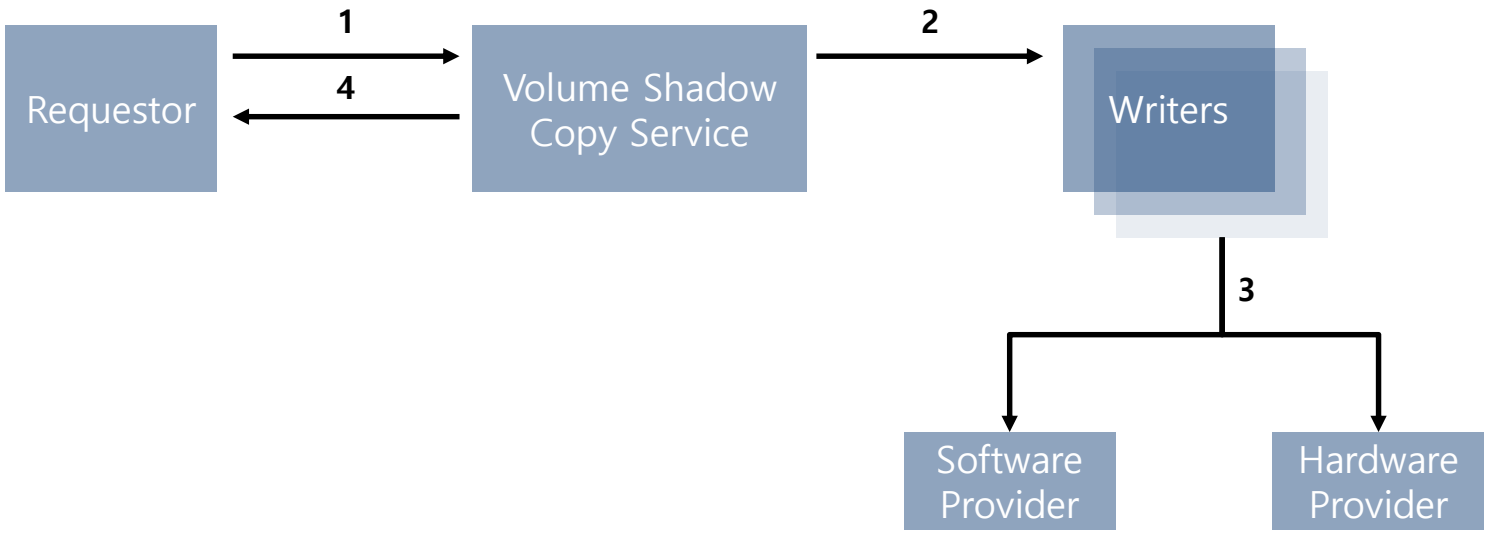
- **VSC 동작 원리**



- 3. Writer는 Provider를 통해 VSC 생성을 준비 함
  - ✓ 열려있는 모든 트랜잭션, 트랜잭션 로그 롤 및 캐시 플러시 등을 수행
  - ✓ 모든 준비 작업이 완료되면 Volume Shadow Copy Service에 생성 준비를 알림

Volume Shadow Copy

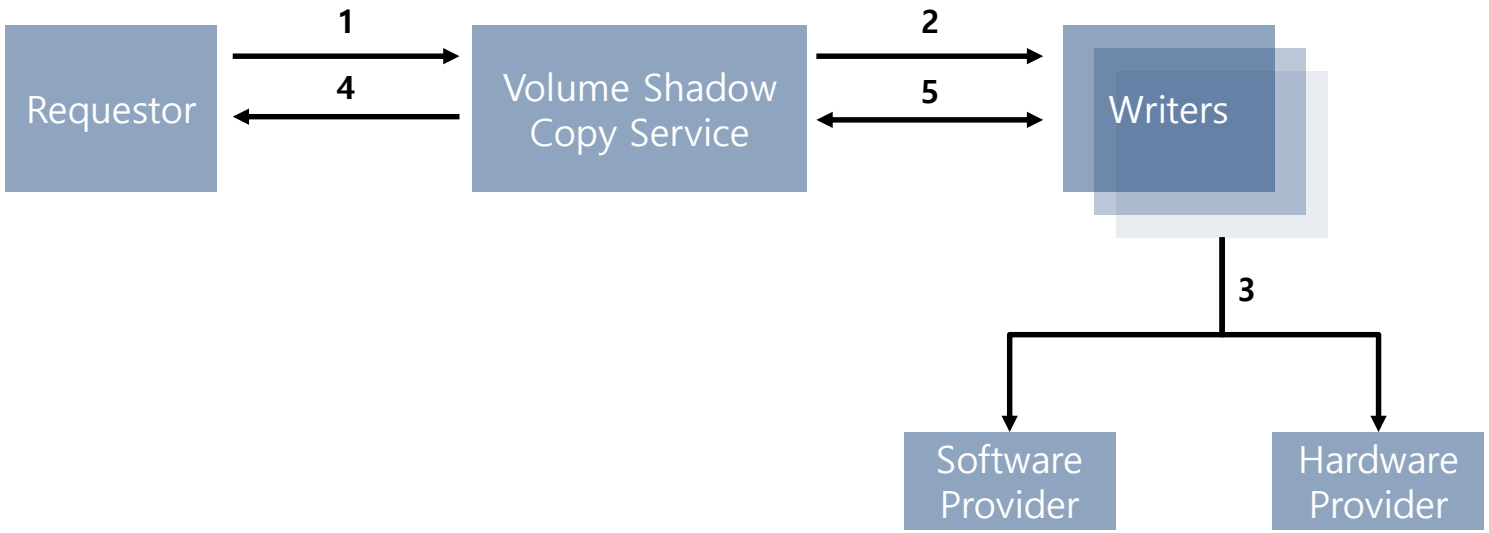
- **VSC 동작 원리**



- 4. Volume Shadow Copy Service는 Requestor에게 VSC 생성 시작을 알림

Volume Shadow Copy

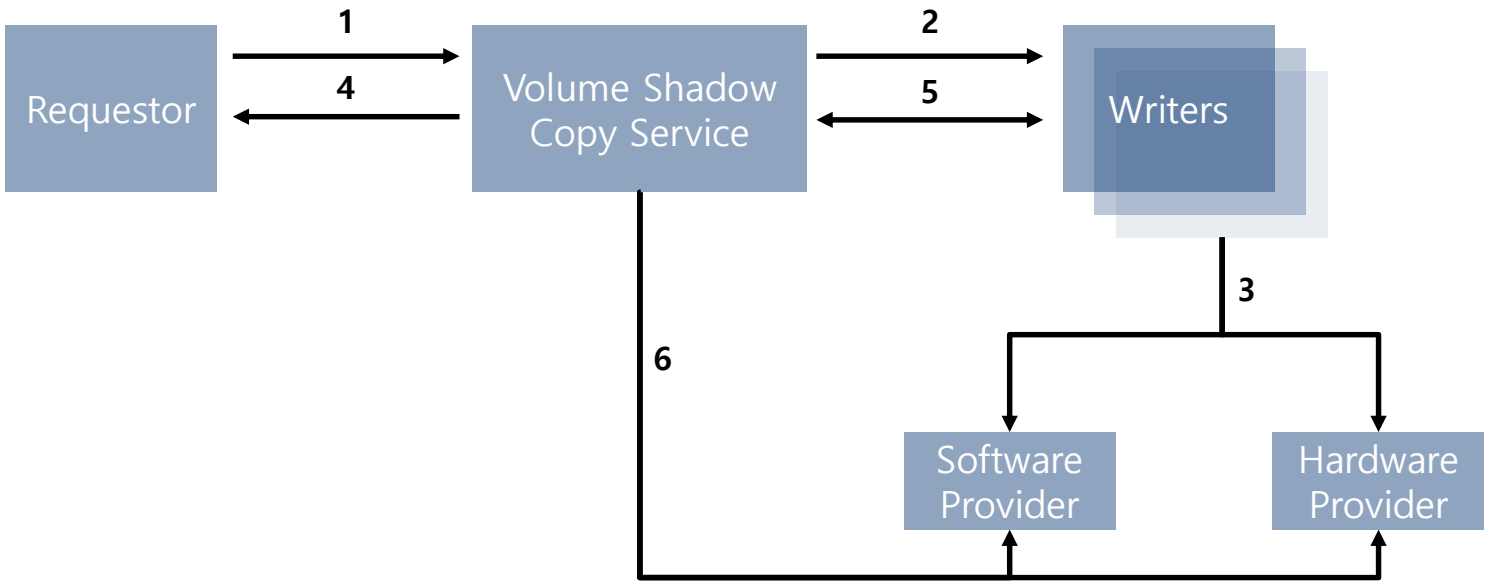
- **VSC 동작 원리**



- 5. Volume Shadow Copy Service는 Writers에게 I/O 쓰기 요청을 일시 정지하도록 명령
  - ✓ 쓰기의 요청의 I/O는 불가능, 읽기 요청의 I/O는 가능
  - ✓ **이때, 파일시스템 버퍼를 flush(메타데이터 포함) 하고 파일시스템을 동결시킴**

Volume Shadow Copy

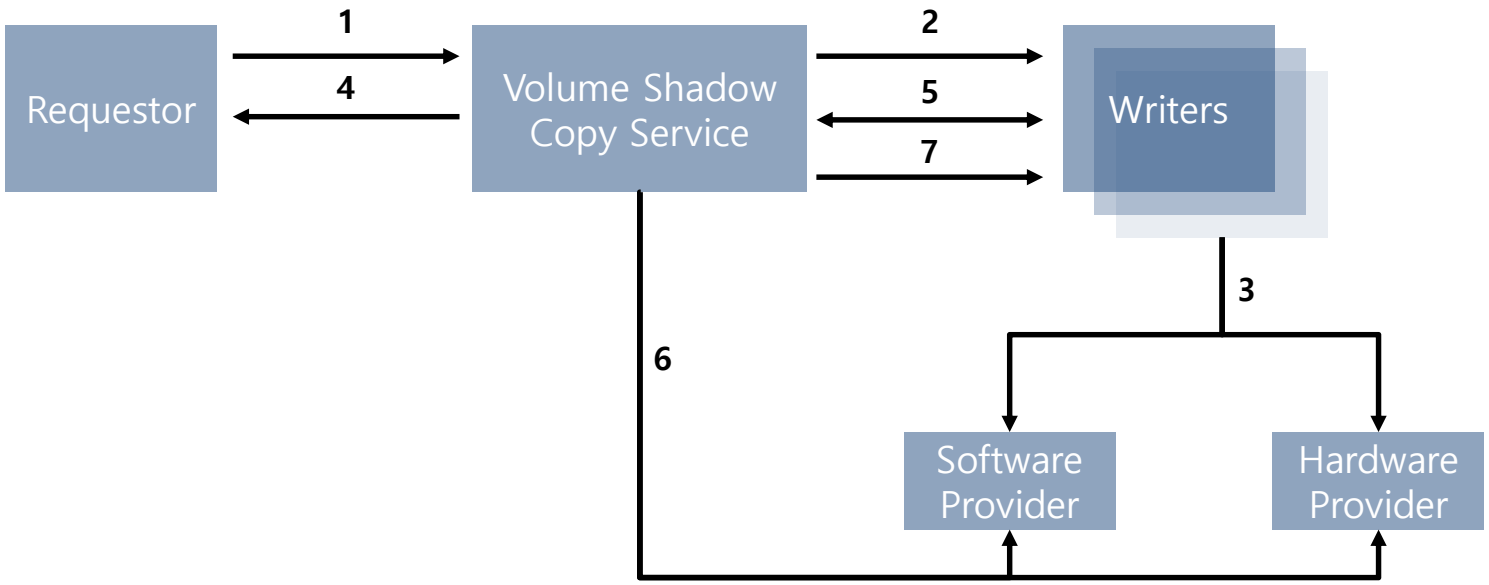
- **VSC 동작 원리**



- 6. Volume Shadow Copy Service는 Provider에게 VSC 생성을 명령
  - ✓ **VSC 생성 명령에 대한 응답이 최대 10초를 넘어서면 VSC 생성 실패로 간주**

Volume Shadow Copy

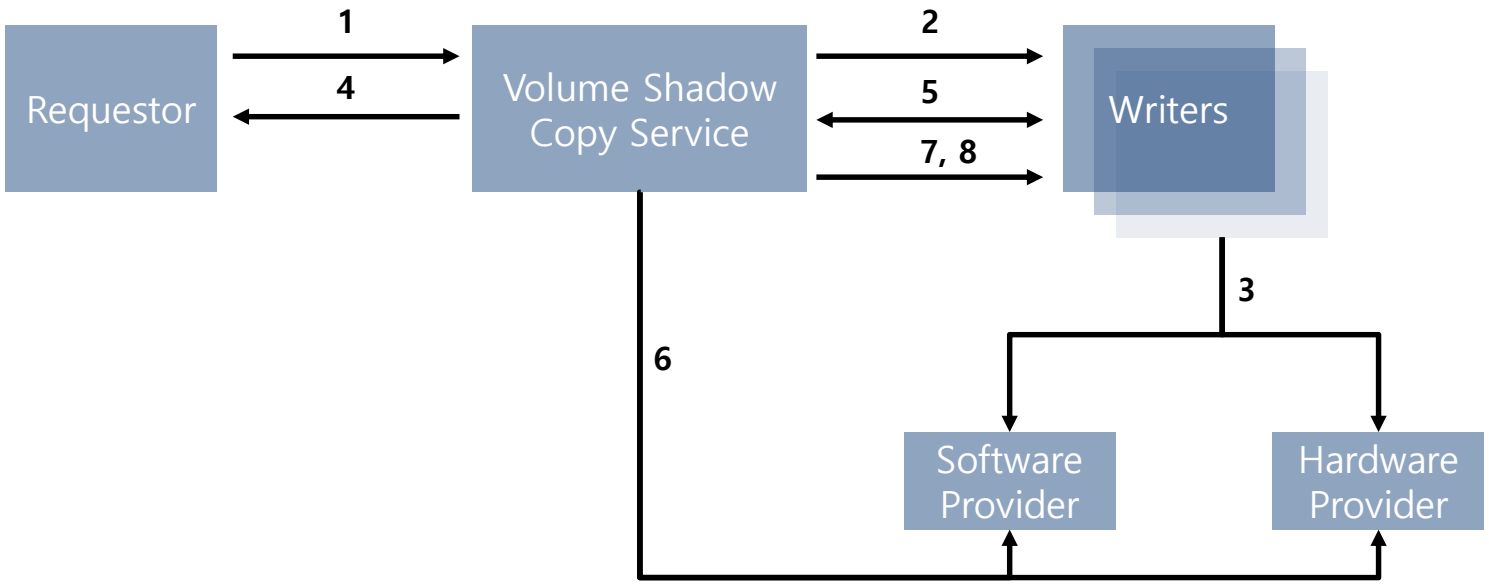
- **VSC 동작 원리**



- 7. Volume Shadow Copy Service는 파일시스템 일시 정지 상태를 해제 함
  - ✓ 대기 중인 쓰기 요청 I/O를 순차적으로 진행 및 완료 함

Volume Shadow Copy

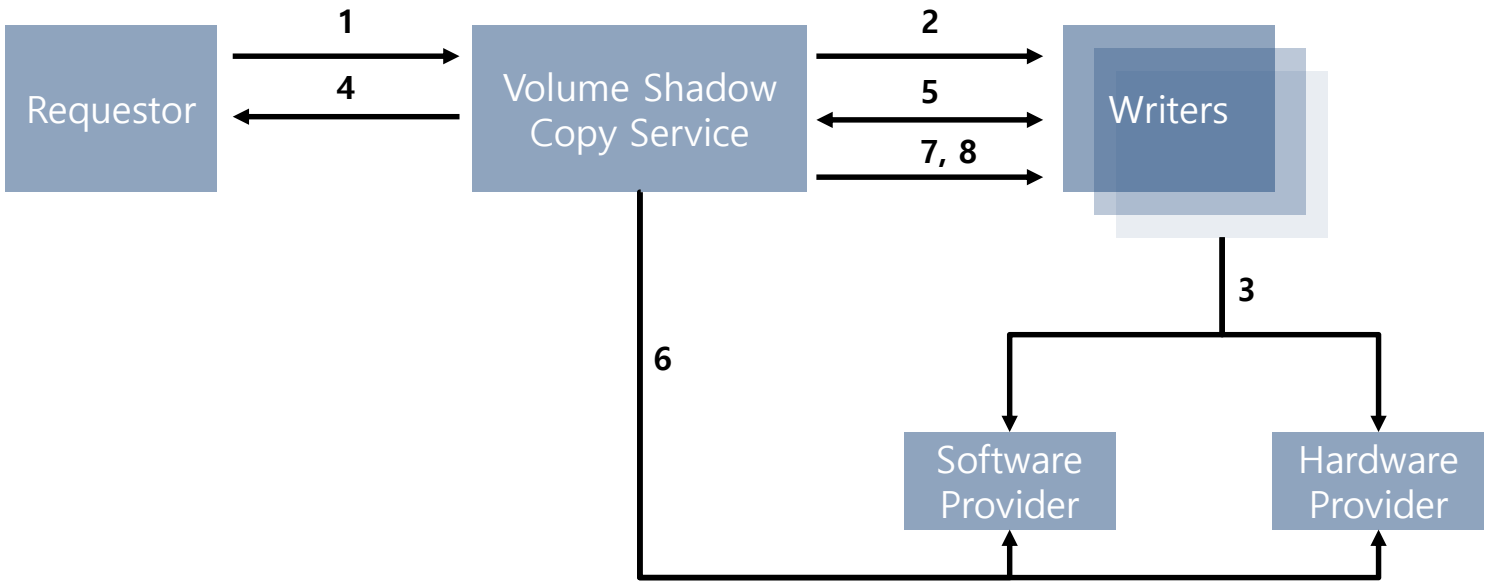
- **VSC 동작 원리**



- 8. Volume Shadow Copy Service는 Writers에게 쓰기 요청 I/O 처리 확인을 요청
  - ✓ Writers에 쓰기 요청 I/O가 잘 수행되었는지 확인하는 쿼리를 전송 함

Volume Shadow Copy

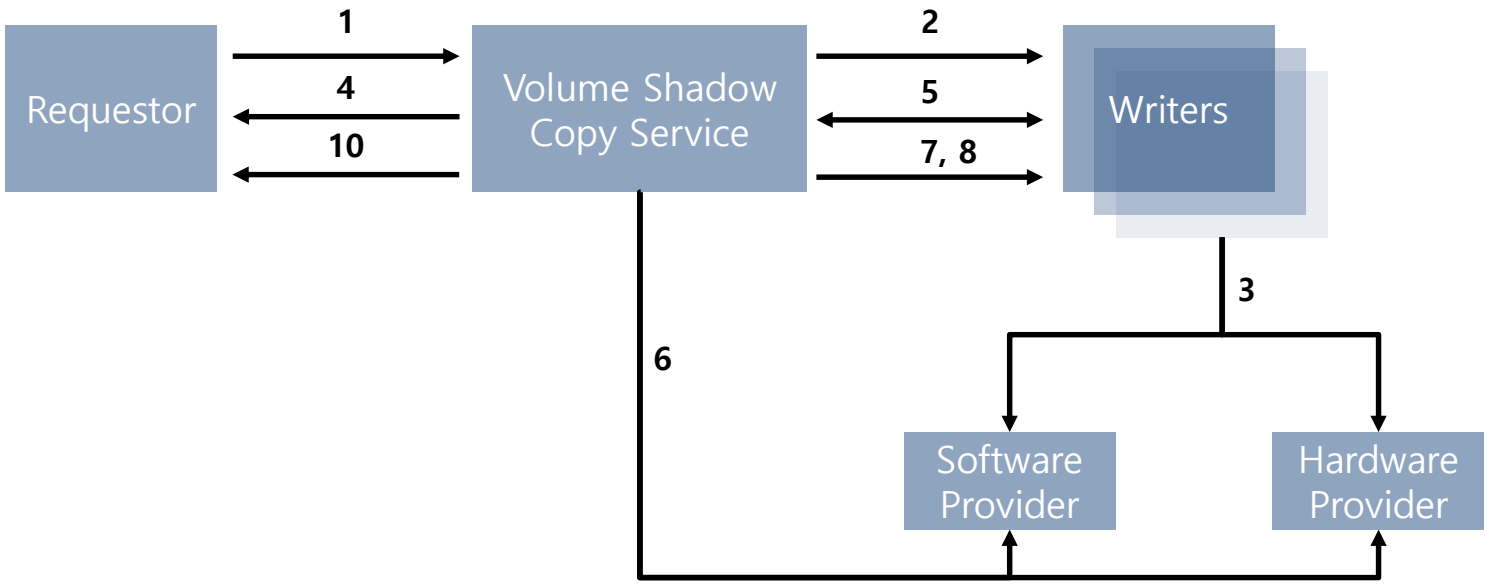
- **VSC 동작 원리**



- 9. 만약, 쓰기가 성공적으로 수행 되지 않았다면 VSC 삭제 요청
  - ✓ VSC가 생성된 이후에 발생한 쓰기 I/O가 성공적으로 수행되지 않았다면 **VSC의 데이터 일관성이 없음**으로 판단하여 VSC를 삭제

Volume Shadow Copy

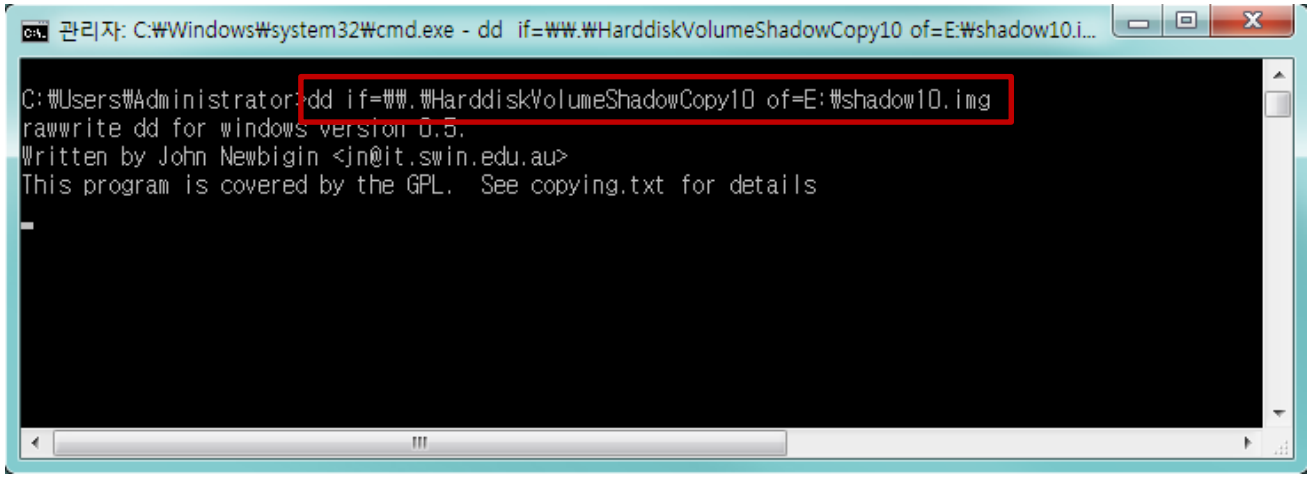
- **VSC 동작 원리**



- 10. VSC 생성이 성공하면 생성 작업을 마치고 섀도 복사본의 위치 정보를 Requestor에게 전달
- **11. 다음 VSC가 생성될 때까지 Volume Shadow Copy Service는 시스템의 변경사항을 모니터링(16KB 단위의 블록)하여 변경 이벤트가 발생하면 현재 VSC에 변경된 블록을 압축하여 저장**

Volume Shadow Copy

- **VSC 이미징**
  - VSC에 접근하기 위해서 다음과 같은 방법이 많이 사용 됨
    - ✓ mklink, VSCToolSet, Shadow Explorer
  - 기존 방법으로는 일반(시스템) 파일만 접근 가능
  - **파일시스템 메타데이터 혹은 비할당 영역에 접근하기 위해서는 raw 데이터가 필요**
    - ✓ 현재로서는 dd(disk dump)를 이용해 접근 가능



AccessData FTK Imager 3.4.2.6

File View Mode Help

Evidence Tree

- shadow10.img
  - 로컬 디스크 [NTFS]
    - [orphan]
    - [root]
      - !Gq]Hgln
      - $BadClus
      - $Extend
      - $Recycle.Bin
      - $Secure
      - $WINDOWS.~BT
      - $Windows.~WS
      - Boot
      - data
      - Documents and Settings
      - ESD
      - Intel
      - Kings
      - MSOCache
      - NVIDIA
      - PerfLogs
      - Program Files
      - Program Files (x86)
      - ProgramData
      - Python27

File List

Name	Size	Type	Date Modified
새 폴더		$I30 INDX Entry	
Work.lnk	1	Regular File	2017-02-20 ...
swapfile.sys.bak	262,144	Regular File	2017-02-21 ...
swapfile.sys	262,148	Regular File	2017-02-21 ...
pagefile.sys.bak	1,966,080	Regular File	2017-02-21 ...
pagefile.sys	1,966,084	Regular File	2017-02-21 ...
Case.lnk.FileSlack	4	File Slack	
Case.lnk	1	Regular File	2016-11-28 ...
000062.xml		$I30 INDX Entry	
000048.xml		$I30 INDX Entry	
000024.xml		$I30 INDX Entry	
000004.xml		$I30 INDX Entry	
$I30	20	NTFS Index All...	2017-02-21 ...

Properties

Name	pagefile.sys
File Class	Regular File
File Size	2,013,270,016
Physical Size	2,013,270,016
Start Cluster	84,872,052
Date Accessed	2017-02-21 오전 3:01:38
Date Created	2017-02-21 오전 3:00:43
Date Modified	2017-02-21 오전 3:01:38
Encrypted	False
Compressed	False
Actual File	True
Start Sector	678,976,416
DOS Attributes	
Hidden	True

Hex Value Interpreter | Custom Content Sources | Properties

```
052c23f0 FF FF FF FF FF FF FF FF-02 00 00 00 38 92 EE 01 ÿÿÿÿÿÿÿÿ····8·î·
052c2400 28 02 00 00 FF FF FF FF-00 00 00 00 00 00 00 00 (···ÿÿÿÿ········
052c2410 18 00 00 00 0C 00 00 00-00 00 00 00 0D 00 00 00 ················
052c2420 76 21 34 2E 30 2E 31 34-33 30 35 2E 30 00 00 00 v!4.0.14305.0···
052c2430 D8 FF FF FF 76 6B 0C 00-0C 00 00 00 28 92 EE 01 Øÿÿÿvk·····(·î·
052c2440 03 00 00 00 01 00 00 00-4D 43 50 5F 30 35 30 31 ····MCP_0501
052c2450 64 35 66 66 00 00 00 00-F0 FF FF FF 00 00 00 00 d5ff····ðÿÿÿ····
052c2460 65 26 36 02 49 0B 00 00-F0 FF FF FF 00 92 EE 01 e&6·I···ðÿÿÿ··î·
052c2470 48 92 EE 01 00 00 00 00-E8 FF FF FF 76 6B 00 00 H·î·····èÿÿÿvk··
052c2480 04 00 00 80 4A 00 00 00-04 00 00 00 00 00 E5 01 ····J·····å·
052c2490 F0 FF FF FF 6C 68 01 00-A0 91 EE 01 86 A0 17 B6 ðÿÿÿlh·· ·î·· ·¶
052c24a0 70 FF FF FF 6E 6B 20 00-0D CC DC 56 DF D1 01 pÿÿÿnk ···ÌÜVßÑ·
052c24b0 00 00 00 00 10 06 2F 01-01 00 00 00 00 00 00 00 ·····/·····
052c24c0 30 94 EE 01 FF FF FF FF-01 00 00 00 A8 93 EE 01 0·î·ÿÿÿÿ····¨·î·
052c24d0 80 05 00 00 FF FF FF FF-1A 00 00 00 00 00 00 00 ····ÿÿÿÿ········
052c24e0 2A 00 00 00 72 00 00 00-00 00 00 00 3E 00 00 00 *···r·····>···
052c24f0 78 38 36 5F 6E 65 74 66-78 34 2D 73 79 73 74 65 x86_netfx4-syste
052c2500 6D 2E 61 64 64 69 6E 5F-62 30 33 66 35 66 37 66 m.addin_b03f5f7f
052c2510 31 31 64 35 30 61 33 61-5F 6E 6F 6E 65 5F 39 63 11d50a3a_none_9c
052c2520 64 61 34 37 33 31 32 63-31 64 31 36 64 62 00 00 da47312c1d16db··
052c2530 D0 FF FF FF 76 6B 15 00-72 00 00 00 30 93 EE 01 Ðÿÿÿvk··r···0·î·
052c2540 03 00 00 00 01 00 00 00-66 32 35 36 21 73 79 73 ····f256!sys
052c2550 74 65 6D 2E 61 64 64 69-6E 2E 64 6C 6C 00 00 00 tem.addin.dll···
052c2560 88 FF FF FF 7A 00 42 00-4B 00 32 00 38 00 68 00 ·ÿÿÿz·B·K·2·8·h·
052c2570 50 00 67 00 44 00 45 00-57 00 34 00 79 00 4C 00 P·g·D·E·W·4·y·L·
052c2580 6C 00 2F 00 73 00 36 00-59 00 70 00 6D 00 66 00 l·/·s·6·Y·p·m·f·
```

Cursor pos = 0; clus = 84872052; log sec = 678976416

shadow10.img/로컬 디스크 [NTFS]/[root]/Users/Administrator/Desktop/pagefile.sys

Volume Shadow Copy

- **결론**
  - **Volume Shadow Copies에서 파일시스템 메타데이터 분석 가능**
    - ✓ VSC 스냅샷 시점의 파일시스템 로그 등을 확인할 수 있음
  - **Volume Shadow Copies에서 비할당 영역 복구 가능**
    - ✓ 현재 시점에서 덮어 씌어진 비할당 영역 데이터를 VSC 스냅샷을 통해 복구할 수 있음
  - **추후에는 VSC 파일 복구 연구**
    - ✓ VSC 메타데이터

ShellBag

ShellBag

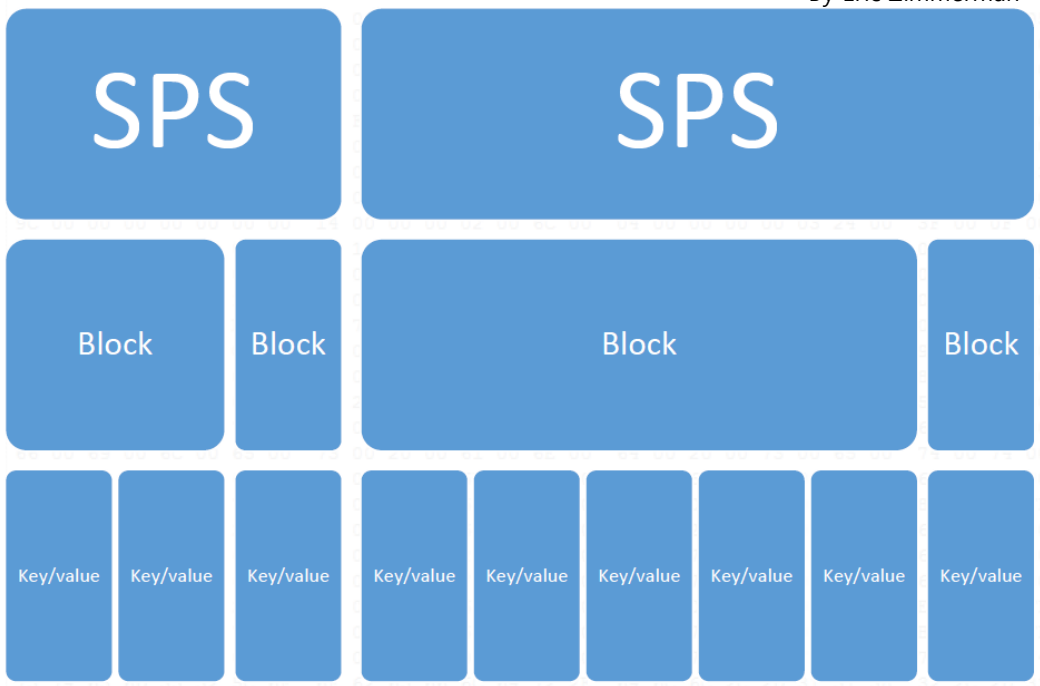
- **기존 포렌식 관점**
  - 레지스트리 키의 마지막 쓰기 시간
  - 생성/수정/접근 시간
  - 폴더 경로

- **새로운 포렌식 관점**
  - 파일시스템 유형
  - $MFT Entry number, seq number
  - ShellBag에 저장되는 다양한 유형

ShellBag

- **Serialized Property Storage (SPS)**
  - Microsoft Property Store Binary File Format
  - 여러 속성 정보를 담고 있는 바이너리 형태의 데이터 블록
  - 기본적으로 데이터는 Key/Value 형태로 저장

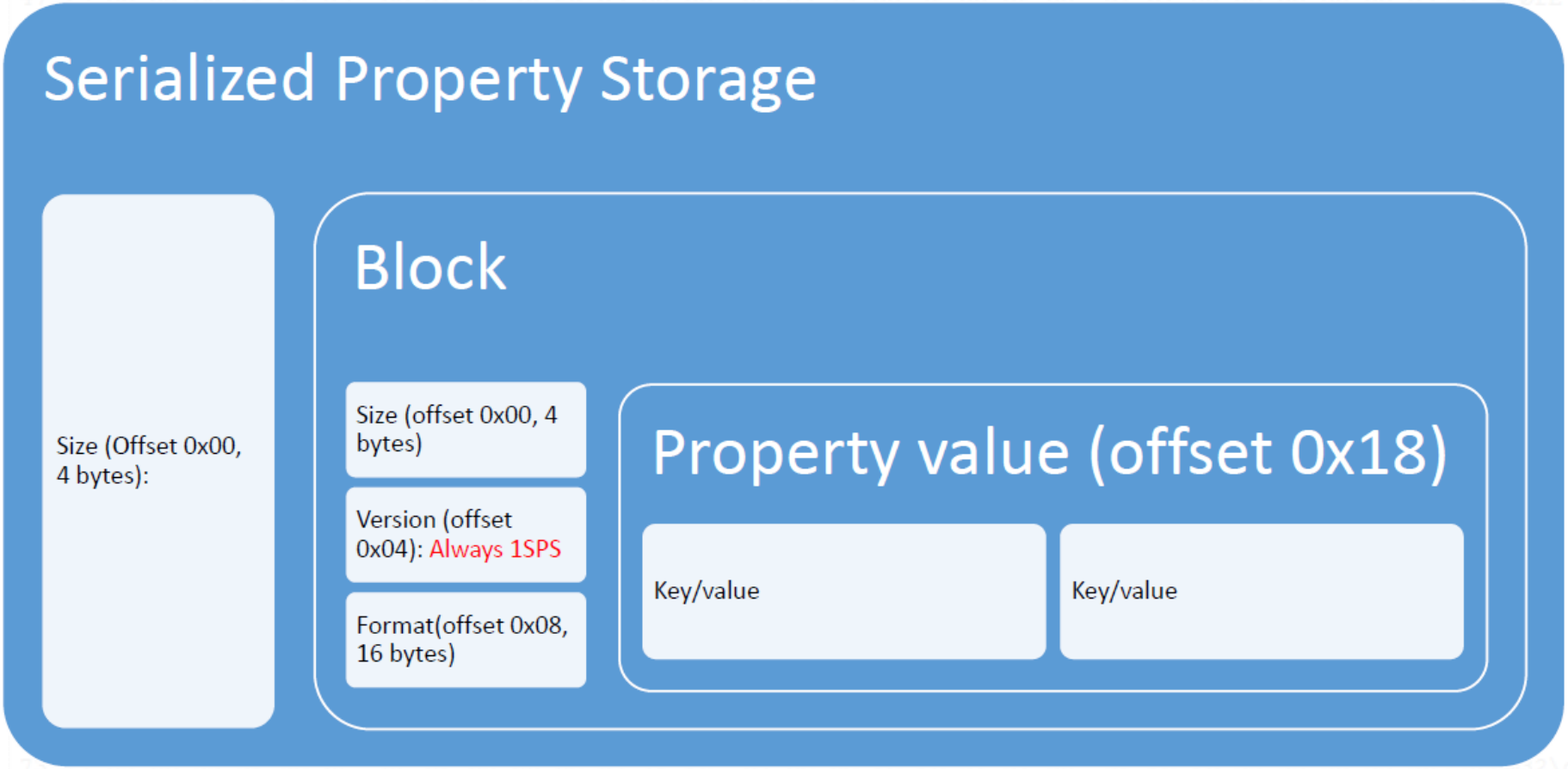
By Eric Zimmerman



ShellBag

- **Serialized Property Storage (SPS)**

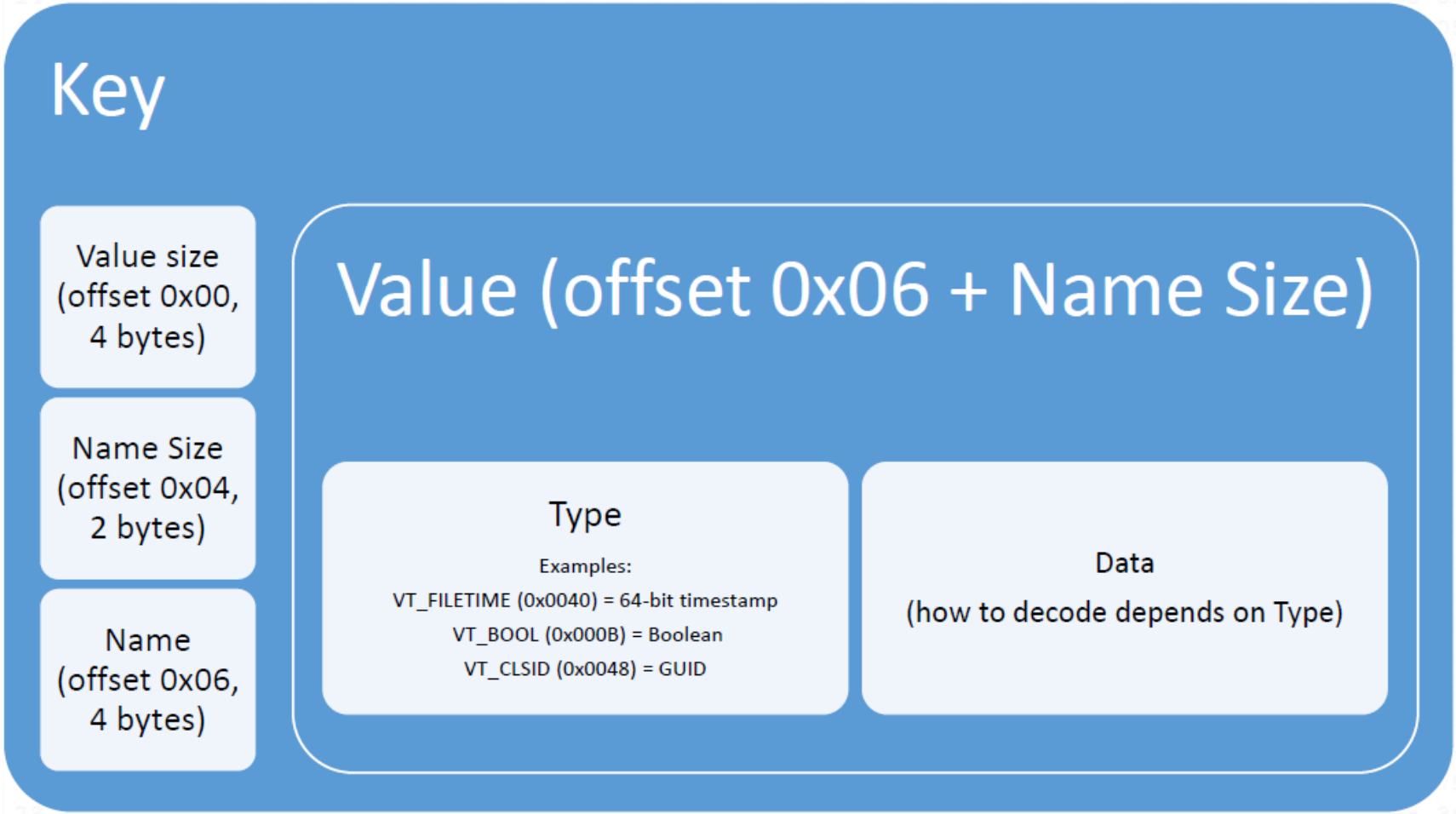
By Eric Zimmerman



ShellBag

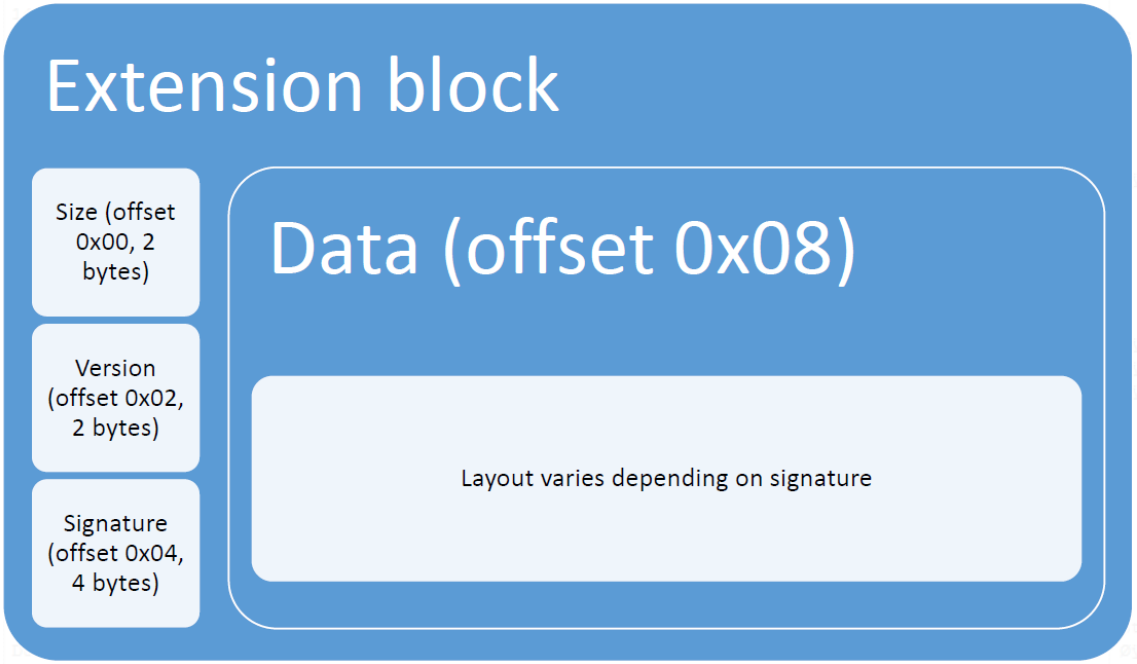
- **Serialized Property Storage (SPS) ➔ Key/Value**

By Eric Zimmerman



ShellBag

- **Extension Blocks**
  - SPS 외부에 존재하는 Block
  - ShellBag 항목은 Extension Block을 여러 개 가질 수 있음
  - SPS 구조 포함 가능



ShellBag

- **Extension Blocks (BEEF0004)**
  - **Timestamps**
    - ✓ Created
    - ✓ Modified
    - ✓ Last Accessed

    *** FAT 파일시스템의 경우 Local Datetime이 저장됨**
  - **Version #**
    - ✓ 아이템이 저장된 파일시스템의 번호

ShellBag

- **Extension Blocks**
  - **File Reference Info**
    - ✓ Record for the directory
    - ✓ NTFS : $MFT Entry #, Sequence #
    - ✓ FAT : Directory Entry #
    - ✓ exFAT : Null
  - **Names**
    - ✓ Long name (Unicode)
    - ✓ Localized name (Unicode or ASCII)

C&AT - Collector & Analyzer Toolbox

File Process Output Help

Collector | Windows Analyzer | Linux Analyzer | MacOSX Analyzer | Timeline

Lnk | Jumplist | Registry | INTERNET | EVENT LOG | Prefetch | FS LOG | $MFT | Thumnbnail

SOURCE

Registry Folder Path: C:\Users\Administrator\Desktop\새 폴더 ....

Windows Info
- Windows Info

User Activity Info
- User Account
- Run Command
- Search Keyword
- IE - Opened page
- Remote Desktop
- Network Drive
- RecentDocs
- ShellBag
- Connected Storage

System Config
- Services
- Drivers
- Auto Runs
- Forensic Readiness

Application Info
- Installed Application
- Application Log
- AppCompat Cache
- 'HWP' Recent Docs
- 'MS Office' Recent Docs
- Cloud Service

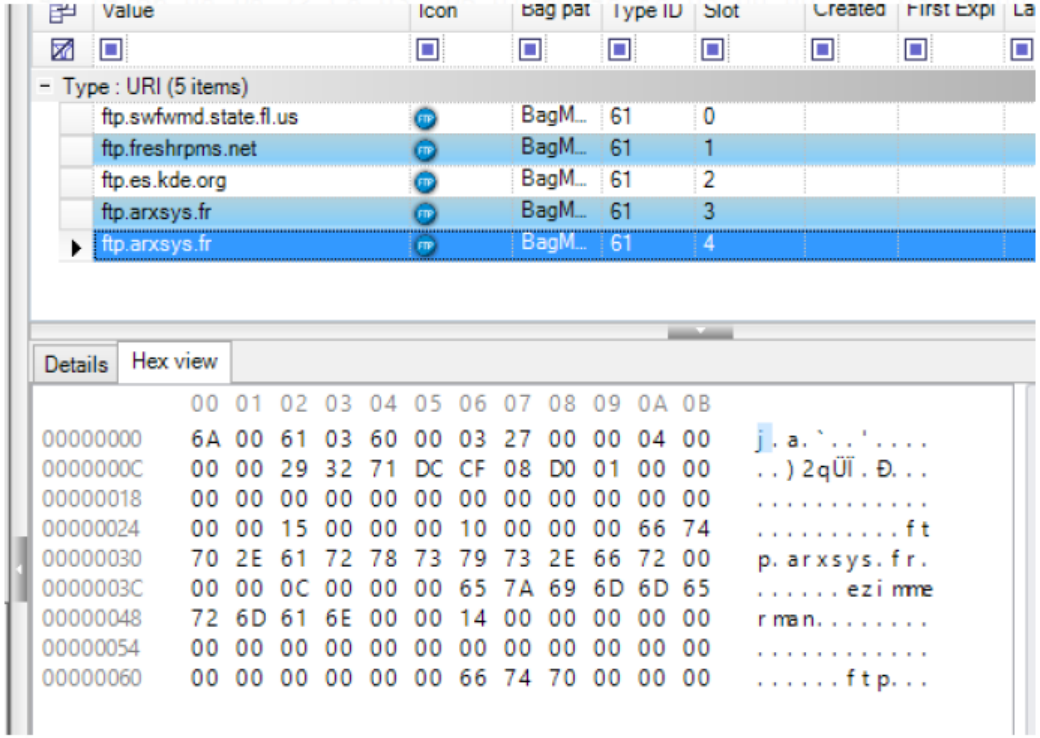
Network Info
- TCP / IP
- Wireless AP

	EntryType	TargetType	TargetFileSystem	TargetFileSize	TargetFullPath	TargetFileReference	
1	-	Archive;	NTFS	0	₩Case.lnk	167223-3462	C:₩
2	-	Archive;	NTFS	0	₩Work.lnk	329325-127	C:₩
3	-	Archive;	NTFS	0	₩Analyzer.sqlite	564072-357	C:₩
4	-	Archive;	NTFS	0	₩Analyzer.log	564083-362	C:₩
5	FOLDER	-	UNKNOWN	0	₩{Network}	-	C:₩
6	UNKNOWN	-	UNKNOWN	0	₩{Network}₩??	-	C:₩
7	-	-	UNKNOWN	0	₩{Network}₩??₩₩₩192.168.1.2₩bays	-	C:₩
8	-	Directory;	NTFS	0	₩{Network}₩??₩₩₩192.168.1.2₩bays₩SAS_S1	0-0	C:₩
9	-	Directory;	NTFS	0	₩{Network}₩??₩₩₩192.168.1.2₩bays₩SAS_S1₩partn-1_exfat	0-0	C:₩
10	-	Directory;	NTFS	0	₩{Network}₩??₩₩₩192.168.1.2₩bays₩blistr	0-0	C:₩
11	-	Directory;	NTFS	0	₩{Network}₩??₩₩₩192.168.1.2₩bays₩SAS_S2	0-0	C:₩
12	-	-	UNKNOWN	0	₩{Network}₩??₩₩₩192.168.1.2₩auditlog	-	C:₩
13	-	Directory;	NTFS	0	₩{Network}₩??₩₩₩192.168.1.2₩auditlog₩pdf	0-0	C:₩
14	-	Directory;	NTFS	0	₩{Network}₩??₩₩₩192.168.1.2₩auditlog₩html	0-0	C:₩
15	UNKNOWN	-	UNKNOWN	0	₩{Network}₩??	-	C:₩
16	-	-	UNKNOWN	0	₩{Network}₩??₩₩₩192.168.1.200₩nfsroot	-	C:₩
17	-	Directory;	NTFS	0	₩{Network}₩??₩₩₩192.168.1.200₩nfsroot₩x3220	0-0	C:₩
18	UNKNOWN	-	UNKNOWN	0	₩{Network}₩??	-	C:₩
19	-	-	UNKNOWN	0	₩{Network}₩??₩₩₩Plainbit-NAS₩web	-	C:₩
20	-	-	UNKNOWN	0	₩{Network}₩??₩₩₩Plainbit-NAS₩web-backup	-	C:₩
21	-	Archive;	NTFS	0	₩Chrome.lnk	559467-14	C:₩
22	FOLDER	-	UNKNOWN	0	₩{Control Panel}	-	C:₩
23	UNKNOWN	-	UNKNOWN	0	₩{Control Panel}₩??	-	C:₩

Loaded registry folder path.

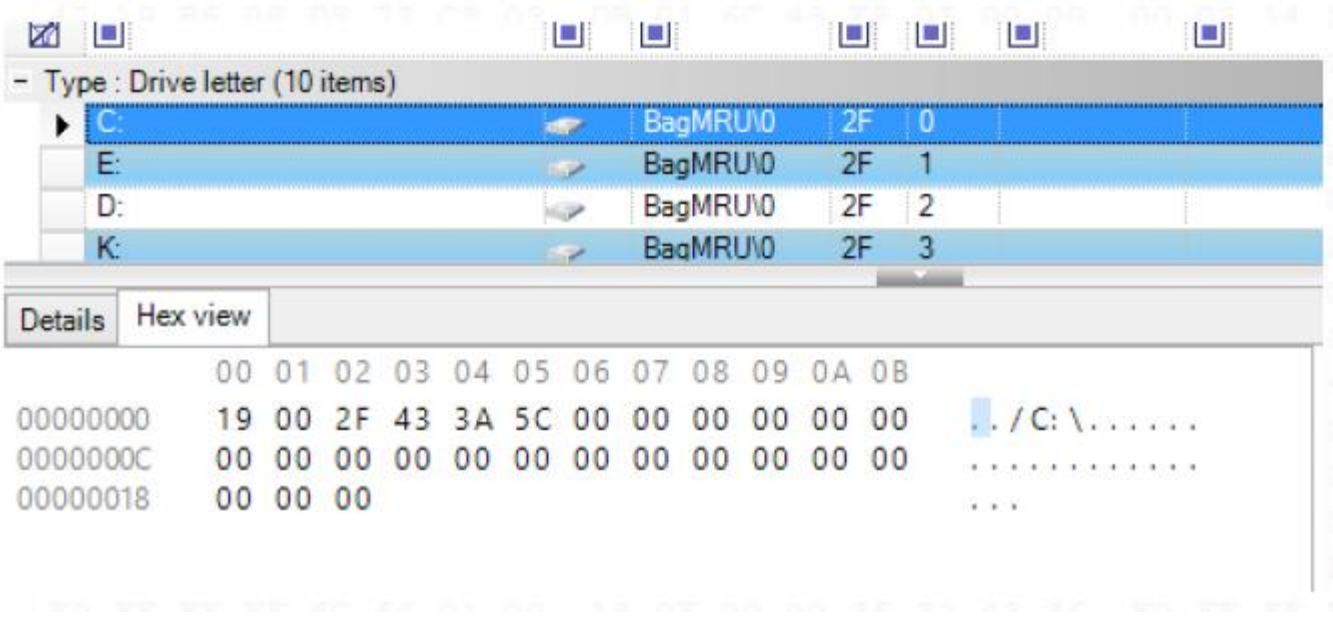
ShellBag

- **Variable item**
  - **Network Resources**
    - ✓ FTP Server directory structures
    - ✓ HTTP and FTP URIs



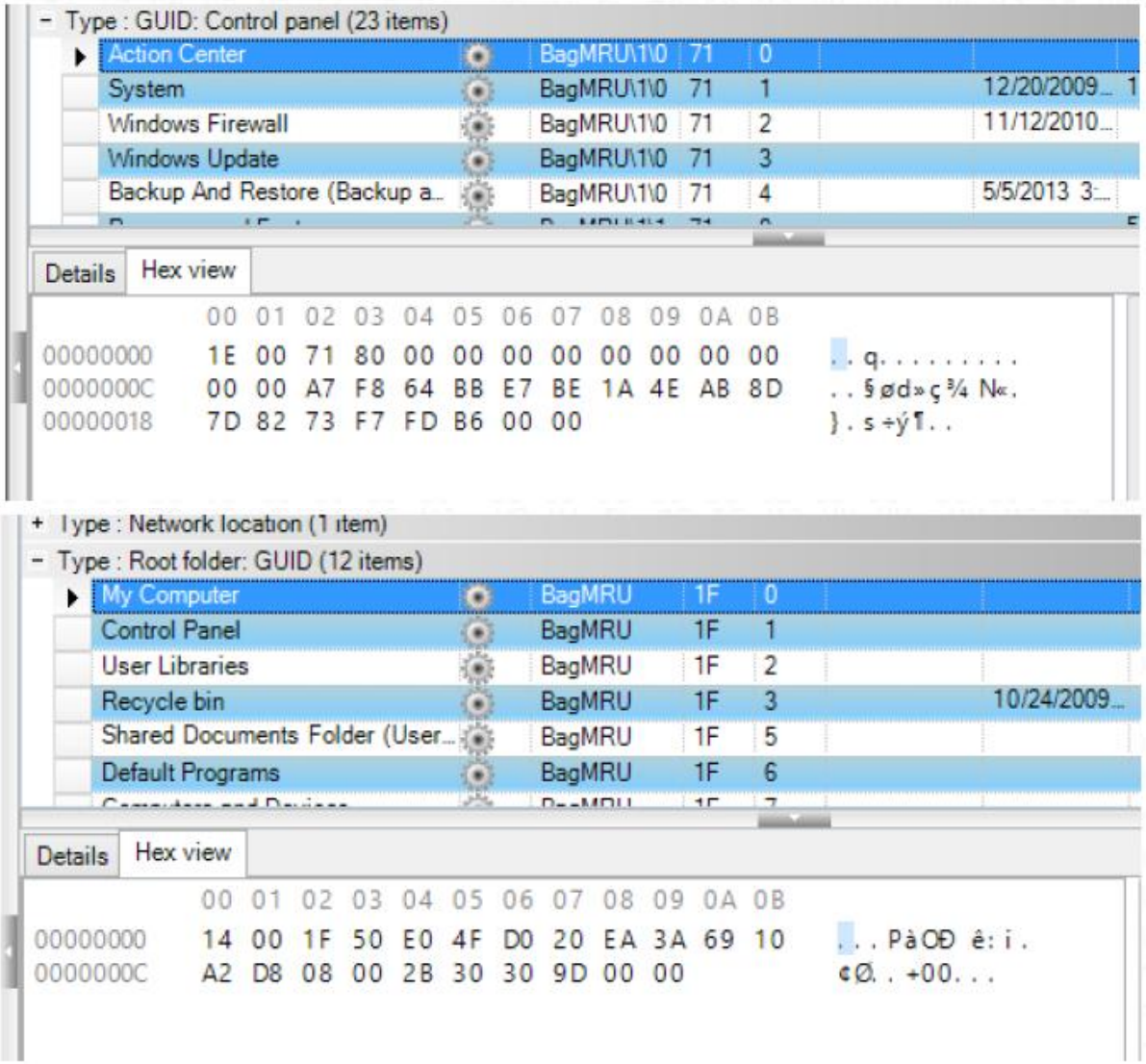
ShellBag

- **Variable item**
  - **Directories**
    - ✓ Local
    - ✓ Remote
  - **MTP**
    - ✓ Volume



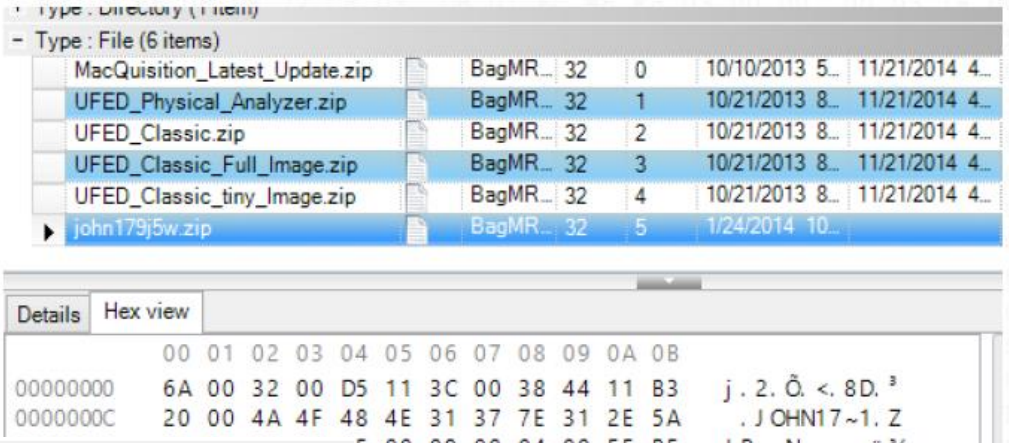
ShellBag

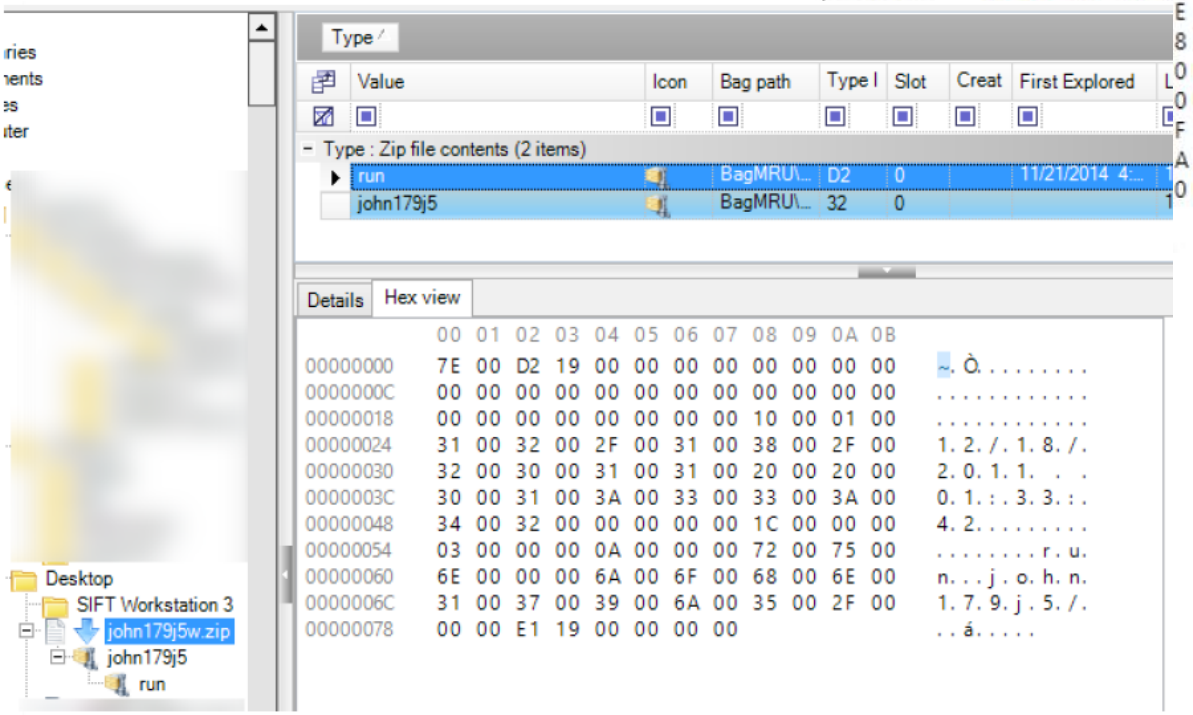
- **Variable item**
  - **GUIDs**
    - ✓ Control panel
    - ✓ Root folder



ShellBag

- **Variable item**
  - **ZIP file contents**
    - ✓ 파일 수정 시각이 기록될 수 있음





ShellBag

- **기록 조건**
  - Desktop 폴더에서 열리거나 Windows Explorer로 폴더가 열릴 경우
  - **압축 파일을 Windows Explorer로 열었을 경우**
    - ✓ 3rd party 프로그램으로 열었을 경우는 남지 않는 것으로 확인 됨
  - **Windows Explorer의 검색 기능을 사용 했을 경우**
    - ✓ 시작 메뉴에 있는 검색창도 유효 함

ShellBag

- **결론**
  - **ShellBag에는 다양한 기록이 존재**
    - ✓ 폴더, 압축 파일 등..
  - **$MFT Entry #, $MFT Seq # 이용해 파일 확인 가능**
    - ✓ 이름이 변경된 파일이어도 동일 파일 여부 확인 가능
  - 파일이 저장되어 있던 볼륨의 파일시스템 파악 가능

*** Extension Block은 Lnk, Jumplist에도 존재 함**

Prefetch

Prefetch

- **기존 포렌식 관점**
  - 마지막 실행 시간
  - 실행 횟수
  - 파일 참조 목록
  - 파일명

- **새로운 포렌식 관점**
  - $MFT File Ref #
  - Volume Info
    - ✓ Volume Name
    - ✓ Volume Serial Number
    - ✓ Volume Created Datetime

Prefetch

- **$MFT File Ref #**
  - $MFT File Ref 값을 이용해 프로그램명이 변경된 프로그램을 찾을 수 있음
    - ✓ 안티포렌식 대응 가능

- **Volume Info**
  - **Volume Name** : 볼륨에 설정된 이름
  - **Volume Serial Number** : 볼륨 시리얼 번호
  - **Volume Created Datetime** : 볼륨 생성 날짜
  - 포터블 프로그램이 실행 됐을 때 어떤 외장저장장치에서 실행 되었는지 파악 가능
  - 운영체제가 설치된 볼륨의 실제 생성 시간과 시리얼 번호를 파악할 수 있음

Question and Answer