Shimcache with AppCompatCacheParser

Type of artifact: Evidence of execution

Basic usage

```
AppCompatCacheParser.exe -f <path to>\SYSTEM --csv c:\temp
```

Output file is a tab separated file that can be imported into Timeline Explorer or Excel.

Key data

Full path for executable and execution flag

Advanced usage

```
AppCompatCacheParser.exe --csv c:\temp
```

Omitting -f switch pulls AppCompatCache data from the Registry hive loaded into memory.

The -d switch can be used to inspect all available details for an entry

Common functionality

Most tools have common options for exporting data, displaying higher precision timestamps, using custom date formats, etc.

When --mp is used, higher precision timestamps are displayed and will also be reflected in any exported data.

Data can be exported to several formats such as csv, json, HTML, etc. at the same time.

```
PECmd.exe -d <directory> --csv c:\temp --html c:\temp\html
```

Amcache.hve with AmcacheParser

Type of artifact: Evidence of execution

Basic usage

```
AmcacheParser.exe -f <path to>\Amcache.hve –csv c:\temp
```

Output file is a tab separated file that can be imported into Timeline Explorer or Excel.

Key data

FullPath: The full path to the executed file
SHA-1: The SHA-1 hash of the file
FileIDLastWriteTimestamp: First executed timestamp
MFTEntryNumber: NTFS entry number from FILE record
MFTSequenceNumber: NTFS sequence number from FILE record

Advanced usage

Use the -b and -w switches to blacklist or whitelist SHA-1 hashes to further reduce the amount of data to review

Use -i to generate a list of associated program/file entries

Download location

Individual tools are available at https://ericzimmerman.github.io/.

Chocolatey packages for each are also available.

To get all tools at once, use chocolatey to install the EricZimmermanTools package

SANS DFIR

DIGITAL FORENSICS & INCIDENT RESPONSE

Eric Zimmerman tools Cheat Sheet v1.0

DFIR.SANS.ORG

Incident Responders are on the front lines of intrusion investigations. This guide aims to support DFIR analysts in their quest to uncover the truth.

How To Use This Sheet

This cheat sheet covers the basics of using several command line programs by Eric Zimmerman.

This sheet is split into these sections:

- Lnk files with LECmd
- Prefetch files with PECmd
- Jumplists with JLECmd
- String searching with bstrings
- Shimcache with AppCompatCacheParser
- Amcache.hve with AmcacheParser

IT'S TIME TO GO HUNTING!

Lnk files with LECmd

Type of artifact: Document creation and opening

Basic usage
LECmd.exe -f <file>
LECmd.exe -d <directory>

Key data
Target timestamps, Volume information, Absolute file path, Target ID information

Advanced usage

Use the --all switch to process all files in a directory vs. only those ending in '.lnk'.

Prefetch files with PECmd

Type of artifact: Evidence of execution

Basic usage
PECmd.exe -f <file>
PECmd.exe -d <directory>

Default output is to standard out. Data can be exported to several formats such as csv, json, HTML, etc.

PECmd.exe -d <directory> --csv c:\temp

Key data
Execution timestamps, total number of executions, and files/directories referenced

Advanced usage
To display higher precision timestamps, use the --mp switch. When --mp is used, the higher precision timestamps will be reflected in any exported data as well.

Jump lists with JLECmd

Type of artifact: Document creation and opening

Basic usage
JLECmd.exe -f <file>
JLECmd.exe -d <directory>

Key data
Same as LECmd key data plus Application ID and DestList entry information (for automaticDestinations jump lists)

Advanced usage
The --ld and --fd switch can be used to display more information about each embedded lnk file.

Use the --appIDs switch to supply a list of application IDs that will be added to the internal list of over 375 appIDs.

In some cases, an automaticDestinations jump list can contain additional lnk files tracked in its Directory that are not accounted for in DestList entries. When this happens, a warning will be given and the --withDir switch can be used to process all available lnk files regardless of them being present in the DestList.

JLECmd also allows for exporting out all available lnk files from a jump list to a directory via the --dumpTo switch. Once lnk files have been dumped from a jump list, they can be investigated using any parser that understands lnk files (LECmd for example).

String searching with bstrings

Type of artifact: Any

Basic usage
bstrings -f <file>

To search for specific strings, use --ls

bstrings -f <file> --ls "forensics"

Use the -x and -m switches to set maximum and minimum string lengths

Use --off to show the offset for each search hit

Advanced usage
In addition to Unicode strings, bstrings looks for strings encoded using Western (1252) code page. Use the --cp switch to search in any other code page supported by .net. A full listing of available code pages is available at https://goo.gl/ig6DxW

bstrings also supports regular expression searches via the --lr switch. bstrings also contains over a dozen built in regular expression patterns for things like credit card numbers, social security numbers, IP addresses, email addresses, and more. To see a list of all built-in regular expressions, use the -p switch. When using a built-in expression, use the value in the Name column. For example, to look for email addresses, use this command:

bstrings -f <some file> --lr email

bstrings also allows searching for several strings or regular expressions at once using the --fr and --fs switches