Comments on “Windows Registry Forensic Tool Specification” (Public Draft 1 of Version 1.0)

Section 5

Page 4, lines 172-177:

It should be noted that some registry hives do not have a visible mount point, these registry hives are called “application hives” [1]. The Amcache hive in Windows 8 and later versions of Windows NT is an example of such a registry hive.

Page 4, figure 1:

The figure indicates that there are trailing empty blocks at the end of the hive file. These blocks, when present, may contain remnant data with deleted keys and values, so they cannot be considered empty in all cases. Also, the size of these blocks is not necessary a multiple of 4 KB, the size is, in fact, arbitrary.

The same figure states that there is the “regf” hive header in the beginning of the file. This name is not an official one, the official name is “base block”.

Page 4, lines 188-189:

The “named key”, “value key”, “security key”, and “data block” are not official names of corresponding structures. The official names are: “key node”, “key value”, “key security”, and “big data”.

Page 5, section 5.2, list of conditions:

The list states that “The date and time value is stored in a FILETIME (UTC) structure”. This is true, except for the “Last reorganized timestamp” in the base block – it is actually the FILETIME timestamp, but with a lower accuracy: the last two bits of this timestamp are used to encode the reorganization type (either the hive defragmentation or access bits zeroing).

The same list states that “A key name has a limit of 255 characters”. This is incorrect because it is possible to store 256 characters in a key name using a regular API function (and the official documentation is wrong with this limit).

The same list states that “ASCII strings are Single Byte Character (SBC) or Multi Byte Character (MBC) string stored with a codepage”. This is incorrect, key names and value names are stored either in ASCII (extended) or Unicode (UTF-16LE). ASCII (extended) strings are stored in the Latin-1 encoding without any code page marks (some sources state that these strings are stored in the Windows-1252 encoding or in the system-specific encoding, but this is also incorrect).

Page 6, table 1:

The table lists the “%SystemRoot%\System32\Config\DEFAULT” file as a “Template file for NTUSER.DAT registry”. But the DEFAULT hive is not a template one, it is used “for the Local System account and is an alias for HKEY_USERS\S-1-5-18” [2].

Notes:

The section does not refer to transaction log files, which are an important part of the low-level registry storage. There are two transaction log formats: a legacy log (“old format”), which is used in all versions of Windows NT up to and including Windows 8, and an incremental log (“new format”), which is used starting from Windows 8.1.

When the incremental log is used, a kernel may delay writing to a primary file up to an hour (by default) according to the unbiased kernel timer. Thus, if an examiner works with a primary file only, he/she will not see recently updated registry data [3].

Since the unbiased kernel timer is paused when an operating system is hibernated (this includes the hybrid shutdown, which is enabled by default on many computers) or sleeping, updates to a hive may be stored outside of a primary file for multiple days [4].

This behavior is not encountered when the legacy log is used, because in this case all changes to registry data are written to a transaction log file and then, without a delay, to a primary file.

The format of primary files, legacy (“old”) and incremental (“new”) transaction log files is known and documented [5].

Section 7

Please, consider adding the following core requirements:

- A Windows registry forensic tool shall support applying incremental transaction log files without modification of original input data when a hive file is in the dirty state.
- A Windows registry forensic tool shall support unusual but valid characters in key names and value names *(a null byte in a value name, “/” in a key name, “\” and “/” in a value name)*, unusual but valid key names and value names *(“.” and “..”)*, and keys having a subkey and a value sharing the same name *(these cases are important because some tools treat the registry as a file system, especially when such a tool mounts a registry hive using the FUSE subsystem)*.

Also, the following optional requirements seem reasonable:

- A Windows registry forensic tool shall support applying a legacy transaction log file without modification of original input data when a hive file is in the dirty state.
- A Windows registry forensic tool shall support:
  - applying log entries from incremental transaction log files one by one when a hive file is in the dirty state (without modification of original input data); and
  - giving an examiner access to a current hive state each time when a log entry has been applied or collecting relevant registry data each time when a log entry has been applied *(for example, it is possible to extract many timestamps for a single registry key by looking at registry data in individual log entries)*.
- A Windows registry forensic tool shall support hive files truncated in the middle of hive bins data *(for example, when running against carved registry files, because truncated primary files require a different parsing approach to account missing subkeys lists)*.

References

1. Microsoft – RegLoadAppKey function (Windows)
   URL: https://msdn.microsoft.com/en-us/library/windows/desktop/ms724886(v=vs.85).aspx
2. Raymond Chen – The .Default user is not the default user
   URL: https://blogs.msdn.microsoft.com/oldnewthing/20070302-00/?p=27783
3. Maxim Suhanov – Flush strategies in the Windows registry

URL: https://github.com/msuhanov/regf-samples/tree/master/8.1-unreconciled
4. Mark Spencer – Unique Windows Registry data in Fast Boot hibernation and hive transaction logs
   URL: https://insights.arsenalexperts.com/2018/02/unique-windows-registry-data-fast-boot-hibernation-hive-transaction-logs/
5. Maxim Suhanov – Windows registry file format specification
   URL: https://github.com/msuhanov/regf/blob/master/Windows%20registry%20file%20format%20specification.md