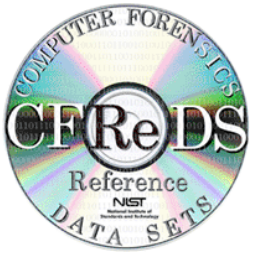


NIST CFReDS
A Windows Registry Dataset
[cfreds-2017-winreg]

Software and Systems Division
Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899

May 2018

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

Table of Contents

1. PROJECT OVERVIEW

The *Windows registry* is a system-defined database in which applications and system components store and retrieve configuration data. The Windows operating system provides registry APIs to retrieve, modify, or delete registry items such as keys, values and data. Note that the Windows registry in this specification means Windows NT registry (i.e. not Windows 3.1 or Windows 95/98/ME).

From digital forensics point of view, the Windows registry is one of primary targets for Windows forensics as a treasure box including not only configurations of the operating system and user installed applications, but also meaningful artifacts that can be useful for identifying users' behaviors and reconstructing their past events. Although Windows registry analysis techniques are already generally being used in Windows forensics, there is a lack of objective and scientific evaluation efforts on digital forensic tools (dedicated registry forensic tools as well as digital forensic suites having registry-related features), which can parse and interpret Windows registry internals. In this situation, NIST/CFTT (Computer Forensic Tool Testing) project aims to enhance the reliability of Windows registry-related forensics by establishing methodologies for conformance testing and quality testing together with NIST/CFReDS (Computer Forensic Reference Data Sets) project.

For achieving the overall aim described above, the CFReDS project first develops a reference Windows registry dataset. The purpose of this work is to provide reference data for research, development and training activities of digital forensic techniques on Windows registry, and furthermore to establish ground truth data for the digital forensic tool testing. The reference dataset developed here will be published on the project website[1] for digital forensics-related communities.

Table 1. Research and tool testing considerations on Windows registry

Research and tool testing considerations	User-generated registry hives	System-generated registry hives	Note
Supporting various input types	V	V	Hive set[2], backup hives[3]
Parsing normal registry hives	V	V	
Parsing corrupted registry hives	V	-	
Recovering deleted registry data	V	V	
Interpreting well-known registry data	-	V	Interpreting artifacts
Countering anti-forensics	V	-	Manipulated structures

As shown in **Table 1**, developing reference Windows registry hives is divided by two types of user-generated and system-generated registry hives, and considers all possible cases for supporting sophisticated tool testing works. (Of course, it can be updated along with the advancement of digital forensic techniques.)

Firstly, user-generated registry hives are synthetic data created experimentally by NIST CFReDS project. These data include various types of registry items that even some of them could not be happened normally in the real Windows OS environment. As a result of this work, we provide not only normal registry hives, but also corrupted and manipulated registry hives in order to support more sophisticated tool testing works.

[1] NIST CFReDS (Computer Forensic Reference Data Sets) - http://www.cfreds.nist.gov
[2] A hive set generally consists of SAM, SYSTEM, SOFTWARE, SECURITY and pairs of [NTUSER, USRCLASS] for each Windows account.
[3] Multiple hive sets from Restore Points (XP or lower) and Volume Shadow Copies (Vista or higher).

For creating user-generated registry hives, we developed several scripts (.REG[4] and PowerShell[5]) which are working with Windows registry API for adding, changing and deleting Windows registry entries. In addition to this, additional programs with open-sources for handling Windows registry internals without calling Windows registry API were developed for understanding differences with Windows registry API.

Secondly, system-generated registry hives are feasible data extracted from reference Windows systems. These data can be utilized for supporting tool testing works from various perspectives including the interpretation of well-known registry data. For this, we first defined possible user actions related to Windows registry artifacts including but not limited to creating accounts, logging on/off accounts, connecting/disconnecting devices, opening/closing/traversing files (or directories), searching keywords, sharing directories, and installing/executing/terminating/uninstalling user applications.

And then, we created a simple scenario depicting user behaviors using defined actions, and developed reference Windows systems with meaningful artifacts based on the scenario. Regarding these reference systems, virtual machines were used for generating artifacts with various Windows OS versions. For the efficient conduct of this task, we tried to automate a large part of user behaviors in order to conveniently create multiple virtual machines with a common scenario. Section 3 will describe details about how to populate virtual machines for registry tool testing works.

[4] Microsoft, How to add, modify, or delete registry subkeys and values by using a .reg file (https://support.microsoft.com/en-us/kb/310516)
[5] Microsoft, Windows PowerShell User's Guide - Working with Registry Entries (https://technet.microsoft.com/en-us/library/dd315394.aspx)

2. USER-GENERATED REFERENCE WINDOWS REGISTRY DATA

This sub-section describes a detailed information including generation strategies relating to user-generated reference registry hives. As shown in **Table 2**, user-generated hives can be utilized for all tool testing points except interpreting well-known registry data.

Table 2. User-generated registry hives and tool testing points

Research and tool testing considerations	User-generated registry hives	System-generated registry hives	Note
Supporting various input types	V	V	Hive set, backup hives
Parsing normal registry hives	V	V	
Parsing corrupted registry hives	V	-	
Recovering deleted registry data	V	V	
Interpreting well-known registry data	-	V	Interpreting artifacts
Countering anti-forensics	V	-	Manipulated structures

2.1. GENERATION STRATEGY

All generation strategies explained in this section consider the following fundamental limitations on Windows registry hive format.

Fundamental Limitations on Windows Registry Entries [6] [7] [8] [9] [10]
- A key name has a limit of 255 characters. - A value name has a limit of 16,383 characters. - A registry tree can be 512 levels deep.

There are four different categories of user-generated registry hives as described in **Table 3**. Category codes at the first column of the table will be used for naming generated reference hive files.

Table 3. Categories of user-generated registry hives

Category codes	Description
NR	Normal registry hive file
NRD	Normal registry hive file with deleted registry data
CR	Corrupted registry hive file
MR	Manipulated registry hive file (including possible anti-forensic activities)

[6] Microsoft, Windows registry information for advanced users (https://support.microsoft.com/en-us/kb/256986)
[7] Microsoft, Registry Element Size Limits (https://msdn.microsoft.com/en-us/library/windows/desktop/ms724872(v=vs.85).aspx)
[8] Peter Norris, The Internal Structure of the Windows Registry. M.S. thesis. Cranfield Univ., UK.
[9] Maxim Suhanov, Windows registry file format specification (https://github.com/msuhanov/regf)
[10] Joachim Metz, Windows NT Registry File (REGF) format specification (https://github.com/libyal/libregf/tree/master/documentation)

2.1.1. Category #1 - Normal registry hive file

The first 'NR' category includes normal registry hives. It means general and benign registry hive files based on the fundamental limitations of the hive format described above. The detailed types included in this category are shown in **Fig. 1**.

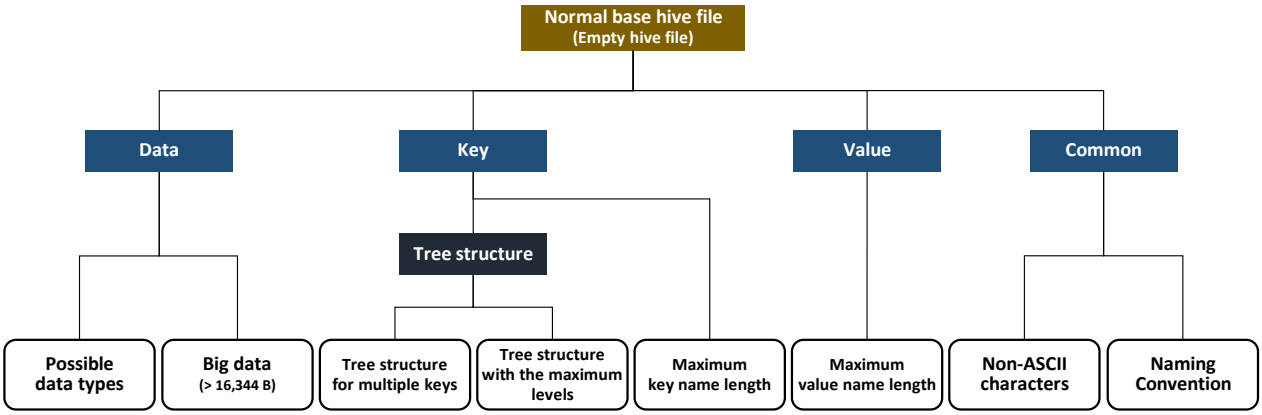


Figure 1. Category #1 – Normal registry hive file

There are 8 different types associated with registry items such as keys and values. In the registry format, the registry items are structured in a tree structure. Each node in the tree is called a key, and each key can contain both subkeys and values which are name/data pairs stored within the key.

Regarding the data stored in a value, we consider all supported data types (such as STRING, BINARY and DWORD) and even big data (> 16,344 bytes in the hive format version 1.5) for creating every possible data objects. In addition, very long key/value names and simple/complicated tree structures are created based on the fundamental limitations. Finally, a common class handles string objects (key name, value name, and data) containing ASCII as well as non-ASCII characters like UTF-16LE, which is a basic encoding in the Windows registry hive format. More specifically, the following characteristics are considered for the 'Naming Convention' type: (1) if the name length field of a 'key value' (vk) structure is ZERO, a tool (like RegEdit.exe) handles it as a default value usually printed as '*(Default)*', (2) unlike file systems, the Windows registry allows a key to have a subkey and a value sharing an identical name, (3) the slash(/) character is allowed to be used for naming keys and values, (4) the backslash(\) is not allowed for naming keys, (5) the dot(.) and double dots(..) are allowed to be used for naming keys and values, and finally (6) ASCII and UTF-16LE characters are allowed to be used for naming keys and values. It should be noted that 0x00 (NULL) and 0x5C (backslash) are not allowed for naming keys.

For creating normal registry hives, we developed several registration entry files (.REG files) and Python scripts, which are working with Windows registry API for adding Windows registry entries. Handling these .REG files is a feature of Windows registry editor (RegEdit.exe)[11], and so registry subkeys and values can be imported with 'RegEdit.exe'.

[11] Microsoft, How to add, modify, or delete registry subkeys and values by using a .reg file (https://support.microsoft.com/en-us/kb/310516)

In addition to this, we tried to develop additional programs with a well-known open-source Hivex[12] library for handling Windows registry internals without calling Windows registry API in order to understand differences between Windows API and other implementations.

Note that for this category the difference between the hive format versions needs to be considered because there are structural changes on managing data stream according to the format version. So, this work considers two different hive format version 1.3 and 1.5 that are mainly used in Windows NT and its successors.

2.1.2. Category #2 - Normal registry hive file with deleted registry data

The second 'NRD' category includes normal registry hives with deleted registry data. A hive file of this category will have unallocated areas as a result of deleting registry items. The detailed types included in this category are depicted in **Fig. 2**.

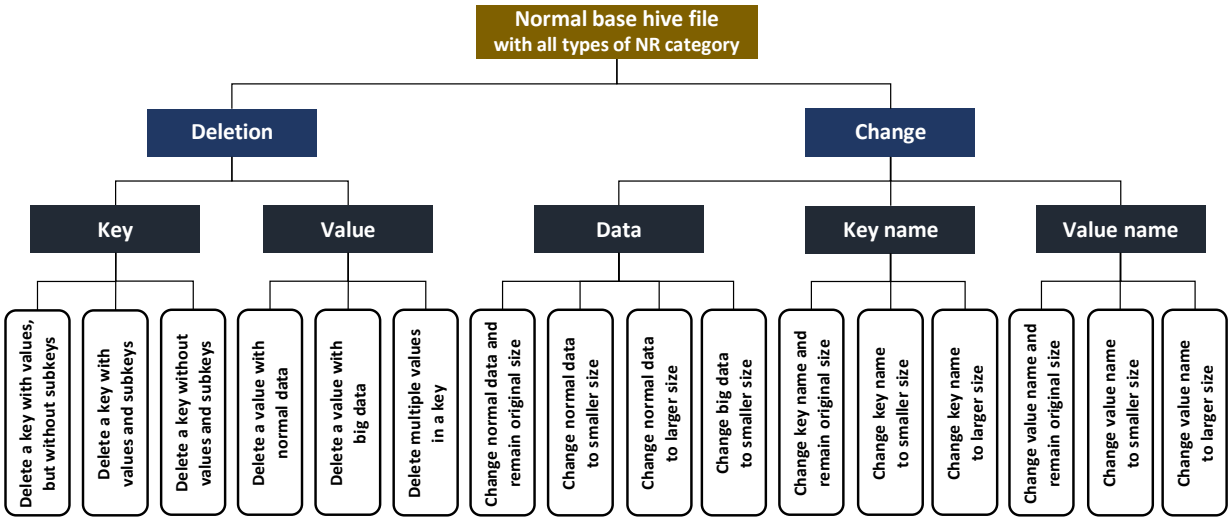


Figure 2. Category #2 – Normal registry hive file with deleted registry data

There are 16 types associated with deleted registry items. In this category, we define two operations 'deletion' and 'change' as user activities related to deleting registry data.

Firstly, the deletion activities are relevant to key and value items with different sub-conditions. Users may not only be able to delete a key with or without sub-items (values or subkeys), but also to delete a value with normal or big data. Secondly, users can change data streams and key/value names with various size conditions. In detail, the change operations for data streams and key/value names are able to be performed while remaining their original size or to smaller/larger size. Note that, in the case of changing big data, this work considers the 'to smaller size' condition only since other conditions are already included for normal data streams.

[12] Red Hat, Hivex – Library for reading and writing Windows Registry 'hive' binary files (http://libguestfs.org)

For creating normal registry hives with deleted registry data, we developed several registration entry files (.REG files), which are working with Windows registry API for deleting and changing Windows registry entries. We also developed PowerShell[13] scripts and utilized Windows registry editor (RegEdit.exe) manually for renaming keys and values because the renaming feature is not supported by a .REG file.

In addition to this, we tried to develop additional programs with an open-source Hivex library for handling Windows registry internals without calling Windows registry API in order to understand differences between Windows API and other implementations.

2.1.3. Category #3 - Corrupted registry hive file

Each hive file included in this 'CR' category will have one or more corrupted blocks. For your guidance, a hive block means the basic unit of allocation for the registry hive format. In the format version 1.3 and 1.5, a hive block is 0x1000 (4,096) bytes, and this is the same with a default cluster size of NTFS (New Technology File System) in Windows NT or its successors.

Fig. 3 shows the internal structure of a registry hive file. The figure depicts an abstract version of the complicated format in order to explain corruption types simply. A registry hive file consists of a hive header (base block) and multiple hive bins, and each hive bin has a hive bin header and hive cell(s).

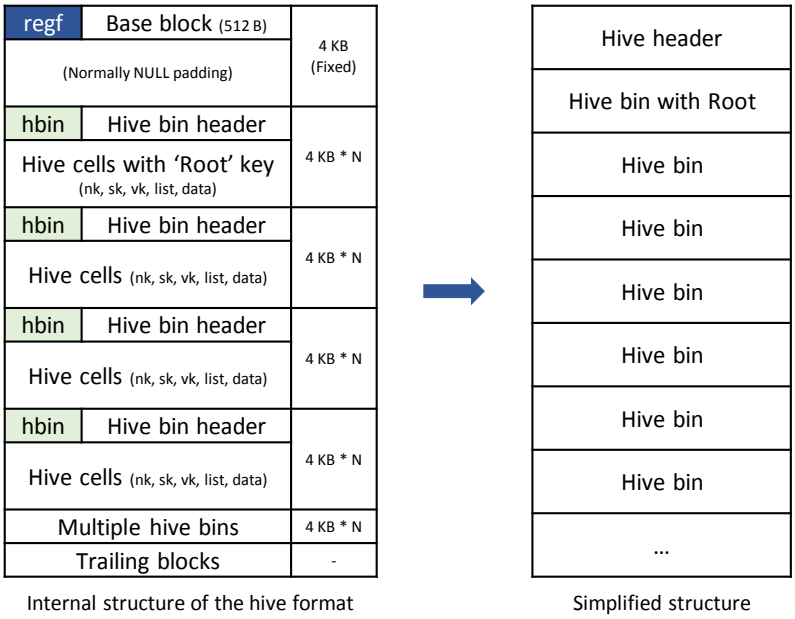


Figure 3. Windows registry hive format internals

[13] Microsoft, Windows PowerShell User's Guide - Working with Registry Entries (https://technet.microsoft.com/en-us/library/dd315394.aspx)

The corruption on a registry hive file can occur in a variety of situations. For example, when Windows system has abnormally shut down, when some parts of compressed or encrypted hive files for the further process could not be decompressed or decrypted properly, when hive files are carved incompletely from unallocated and unused areas of a file systems or binary dump, when there are partial registry data in the physical memory related areas, when a storage media is damaged, etc. The detailed types included in this category are shown in **Fig. 4** and **Fig. 5**.

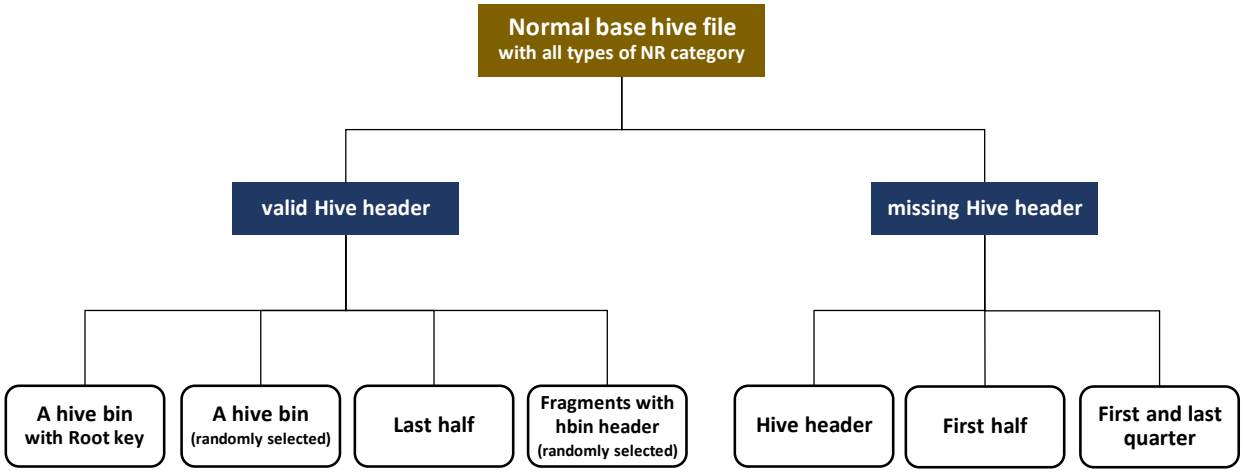


Figure 4. Category #3 – Corrupted registry hive file

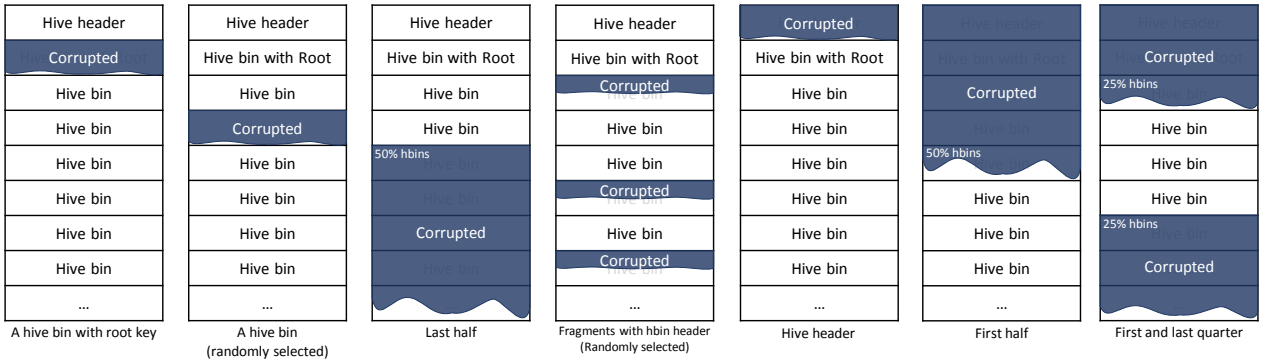


Figure 5. Types of the hive file corruption

In this category, we define two primary conditions, valid and missing hive header (base block), as the key factor of corrupting registry hive files. The corruption is generally likely to occur at a specific fragment unit level such as the sector, cluster or block rather than the bit (or byte) level. Thus, this work tries to overwrite selected hive block(s) in a hive file according to various corruption types depicted in **Fig. 5**. As mentioned above, because a hive block size 0x1000 (4,096) is also the same with a default cluster size in NTFS, these types of corruption can happen in the real environment.

The first four types are to corrupt multiple hive blocks except the hive header block. As you can understand, because there is a valid hive header at the head position, hive files associated with these types of corruption can be identified at the initial step of processing the hive format. However, the fact that they have invalid data can cause exception errors during the parsing of the hive format. So in this case, if there is an automated

tool which can parse hive files, it should provide appropriate exception handling functions, and further it should be able to recover meaningful data from still valid parts if the tool supports it.

Similarly, the last three types try to corrupt multiple hive blocks including the hive header block. In these types, if there is an automated tool which can handle the hive format, it should also support appropriate exception handling functions like the first four types. Of course, although hive files probably won't be identified as the hive format if they don't have a valid hive header, there are still possibilities of recovering meaningful data from undamaged hive blocks.

We developed several python scripts for creating corrupted registry hives. Each script overwrites hive block(s) in a copy of normal base hive file with all types of 'NR' category.

2.1.4. Category #4 - Manipulated registry hive file

The fourth 'MR' category includes manipulated registry hives, which are related to possible anti-forensic activities that can be used to confuse digital forensic tools. As you can imagine, there will be a lot of manipulation methods on the hive format since it is similar to find unpredictable vulnerabilities from unknown executables. So, for limiting the scope, we will define several feasible anti-forensic activities for developing simple automated manipulation methods rather than developing sophisticated fuzzing algorithms for finding all possible manipulation (or attack) points. **Fig. 6** depicts the defined activities and their sub-classes for this category.

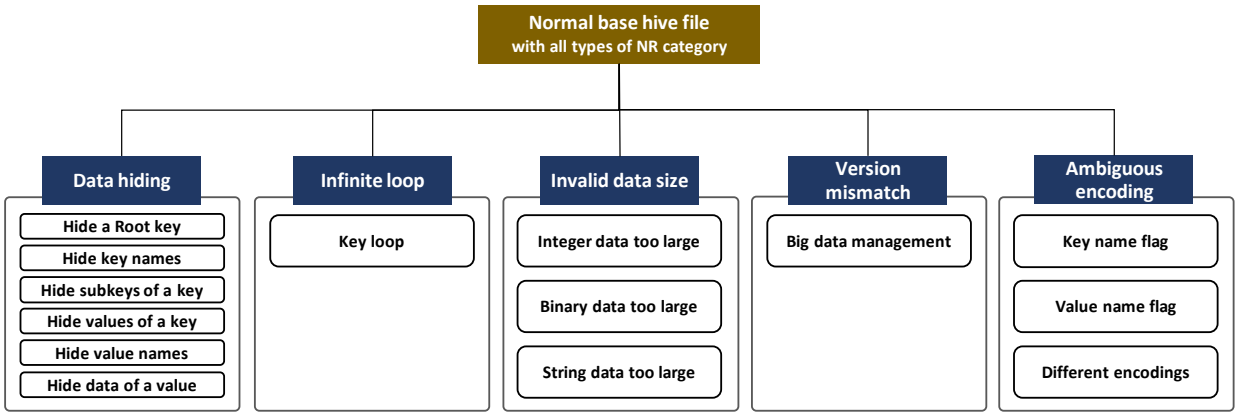


Figure 6. Category #4 – Manipulated registry hive file

As shown in the figure, this category consists of five primary classes. The first class is to hide registry items such as keys, values and its data. This data hiding can be achieved through manipulating the offset and size related to various hive structures including the hive header, key cell (nk), subkey-list cell (lf, lh, ri, li), value-list cell, value cell (vk) and data cell. The second class includes infinite loop (or endless loop) types. For this class, the loop can be created simply because the key cells in the hive format are stored as a tree structure. The third class is related to the data size that is an important factor for getting data properly. In this class, the data consists of more bytes than the proper size of it as a result of editing the data size in the value (vk) cell structure. The fourth class is about the hive version mismatch. Since the big data (> 16,344)

management scheme is introduced from the hive version 1.5, we can patch the version information of the hive header structure from 1.5 to 1.3 (or vice versa) in order to confuse parsing hive files with big data. The final fifth class is to store strings with ambiguous encodings. For existing key/value names, we can simply create ambiguity through changing the encoding flag in the key (nk) and value (vk) cell structure. Also, this class includes adding keys, values and data encoded by various encodings such as ISO8859-15, EUC-KR, KOI8-R, GB18030, EUC-JP, UTF-8, etc.

Table 4 describes the detailed generation strategies for each class described above. For clear understanding, you may need to know about the hive format in detail from existing literature and open-source projects on the Windows registry.

Table 4. Generation strategy for the category #4

Primary Class	Secondary Class	Generation Strategy
Data hiding	Hide a root key (1)	- Edit 'root cell offset' in the hive header structure ◦ *root cell offset ← the 1st subkey offset of the original root key cell* - Remain the original checksum value
	Hide a root key (2)	- Edit 'root cell offset' in the hive header structure ◦ *root cell offset ← the 1st subkey offset of the original root key cell* - Update the checksum value
	Hide key names (1)	- Edit 'key name size' in the key (nk) cell structure ◦ *select a key cell stored at the 1st subkey offset of the root key* ◦ *key name size ← a half of the original key name size*
	Hide key names (2)	- Edit 'key cell size' in the key (nk) cell structure ◦ *select a key cell stored at the 1st subkey offset of the root key* ◦ *adjust = a half of the original key name size* ◦ *cell size ← the original cell size - adjust*
	Hide subkeys of a key (1)	- Edit 'number of subkeys' in the key (nk) cell structure ◦ *select a key cell which has at least 7 subkeys* ◦ *adjust = 4* ◦ *number of subkeys ← the original number of subkeys - adjust*
	Hide subkeys of a key (2)	- Edit 'subkey-list cell size' in the subkey-list cell structure ◦ *select a key cell which has at least 7 subkeys* ◦ *go to the subkey-list cell offset* ◦ *adjust = 4 * sizeof(a key offset item)* ◦ *cell size ← the original cell size - adjust*
	Hide subkeys of a key (3)	- Edit 'number of subkeys' in the subkey-list cell structure ◦ *select a key cell which has at least 7 subkeys* ◦ *go to the subkey-list cell offset* ◦ *adjust = 4* ◦ *number of subkeys ← the original number of subkeys - adjust*
	Hide subkeys of a key (4)	- Edit 'subkey offsets' in the subkey-list cell structure ◦ *select a key cell which has at least 7 subkeys* ◦ *go to the subkey-list cell offset* ◦ *the last 4 subkeys' offsets ← NULL*
	Hide values of a key (1)	- Edit 'number of values' in the key (nk) cell structure ◦ *select a key cell which has at least 7 values* ◦ *adjust = 4* ◦ *number of values ← the original number of values - adjust*
	Hide values of a key (2)	- Edit 'value-list cell size' in the value-list cell structure ◦ *select a key cell which has at least 7 values* ◦ *go to the value-list cell offset* ◦ *adjust = 4 * sizeof(a value offset item)* ◦ *cell size ← the original cell size - adjust*
	Hide values of a key (3)	- Edit 'value offsets' in the value-list cell structure ◦ *select a key cell which has at least 7 subkeys* ◦ *go to the value-list cell offset* ◦ *the last 4 values' offsets ← NULL*

	Hide value names (1)	- Edit 'value name size' in the value (vk) cell structure ◦ *target = the 1st value offset of the 1st subkey of the root key* ◦ *select a value cell stored at the target* ◦ *value name size ← a half of the original value name size*
	Hide value names (2)	- Edit 'value cell size' in the value (vk) cell structure ◦ *target = the 1st value offset of the 1st subkey of the root key* ◦ *select a value cell stored at the target* ◦ *adjust = a half of the original value name size* ◦ *cell size ← the original cell size - adjust*
	Hide data of a value (1)	- Edit 'data size' in the value (vk) cell structure ◦ *select a value cell which has BINARY type data (<= 16,344)* ◦ *data size ← 0*
	Hide data of a value (2)	- Edit 'data cell size' in the data cell structure ◦ *select a value cell which has BINARY type data (<= 16,344)* ◦ *go to the data cell offset* ◦ *cell size ← 4*
	Hide data of a value (3)	- Edit 'data offset' in the value (vk) cell structure ◦ *select a value cell which has BINARY type data (<= 16,344)* ◦ *data offset ← NULL*
	Hide data of a value (4)	- Edit 'data type' in the value (vk) cell structure ◦ *select a value cell which has BINARY type data (<= 16,344)* ◦ *data type ← SZ (UTF-16LE NULL-terminated string)*
	Hide big data of a value	- Edit 'data size' in the value (vk) cell structure ◦ *select a value cell which has BINARY type data (> 16,344)* ◦ *data size ← 0*
Infinite loop	Key loop	- Set one of subkey offsets of a key to point the key or its parent key ◦ *select a key ('A') cell (except root cell) which has subkeys* ◦ *go to the subkey-list cell offset* ◦ *the last subkey offset ← A's parent key offset (building a loop)*
Invalid data size	Integer data too large	- Edit 'data size' in the value (vk) cell structure ◦ *select a value cell which has DWORD_LE type data* ◦ *data size ← 8 (more than 4 bytes)*
	Binary data too large	- Edit 'data size' in the value (vk) cell structure ◦ *select a value cell which has BINARY type data* ◦ *data size ← twice as long as the original string size*
	String data too large	- Edit 'data size' in the value (vk) cell structure ◦ *select a value cell which has SZ type data* ◦ *data size ← twice as long as the original string size*
Version mismatch	Big data management	- Patch v1.3 format to v1.5 format or vice versa ◦ *change the minor version value '3' to '5' or vice versa ('5' to '3')* ◦ *update the checksum value*
Ambiguous encoding	Key name flag	- Edit 'encoding flag' in the key (nk) cell structure ◦ *select a key which has an UNICODE key name* ◦ *change the flag 'UNICODE' to 'ASCII'*
	Value name flag	- Edit 'encoding flag' in the value (vk) cell structure ◦ *select a value which has an UNICODE value name* ◦ *change the flag 'UNICODE' to 'ASCII'*
	Different encodings	- Add keys, values and data encoded by various encodings ◦ *'¡Hola!' encoded by ISO- 8859-15* ◦ *'안녕하세요' encoded by EUC-KR* ◦ *'Здравствуйте' encoded by KOI8-R* ◦ *'您好' encoded by GB18030* ◦ *'こんにちは' encoded by EUC-JP* ◦ *'नमस्ते' encoded by UTF-8*

The classes described in **Table 4** are only part of many possible manipulation methods for confusing digital forensic tools. Of course, automated digital forensic tools which can handle registry hive files may be able to process manipulated registry hives without exceptions if they have been well-implemented with

appropriate exception handlers. Even so, the tools won't be able to identify and present all manipulated data if they support just basic parsing functions. In other words, if the tools support additional functions such as carving data from unallocated areas in the hive file format, a portion of the hidden data may be detected and extracted.

The primary value of this category is as follows. Although it is difficult or sometimes impossible to identify perfectly manipulated (or corrupted unexpectedly) hive files under the specification, they can still be identified in many imperfect cases through the verification process on internal structures because of a special characteristic of the registry hive format that multiple structures (cells) manage similar or same values (mainly size and count) in different positions. That is to say, if users can get any alert messages from digital forensic tools when abnormal values or structures are detected, it would be possible to lead them to do a more detailed analysis.

2.2. GENERATION METHODS FOR CATEGORY #1 (NORMAL REGISTRY HIVE FILE)

2.2.1. Possible data types

Type (1) .REG file
Filename: [nr]-01-1_possible-data-types.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x01_TYPE1_DATA-TYPES]
"VALUE 0x00 (NONE)"     = hex(0):6E,6F,6E,65
"VALUE 0x01 (SZ)"       = "UTF-16LE NULL-terminated string"
"VALUE 0x02 (EXP_SZ)"   = hex(2):25,00,53,00,79,00,73,00,74,00,65,00,6D,00,52,00,\
                                  6F,00,6F,00,74,00,25,00,00,00
"VALUE 0x03 (BINARY)"   = hex(3):62,69,6E,61,72,79,20,64,61,74,61
"VALUE 0x04 (DWORD-LE)" = hex(4):04,00,00,00
"VALUE 0x05 (DWORD-BE)" = hex(5):00,00,00,04
"VALUE 0x06 (LINK)"     = hex(6):53,00,79,00,6D,00,62,00,6F,00,6C,00,69,00,63,00,\
                                  20,00,6C,00,69,00,6E,00,6B,00,00,00
"VALUE 0x07 (MULTI_SZ)" = hex(7):52,00,45,00,47,00,5F,00,4D,00,55,00,4C,00,54,00,\
                                  49,00,5F,00,53,00,5A,00,5F,00,31,00,00,00,\
                                  52,00,45,00,47,00,5F,00,4D,00,55,00,4C,00,54,00,\
                                  49,00,5F,00,53,00,5A,00,5F,00,32,00,00,00,\
                                  52,00,45,00,47,00,5F,00,4D,00,55,00,4C,00,54,00,\
                                  49,00,5F,00,53,00,5A,00,5F,00,33,00,00,00
"VALUE 0x08 (RES_LIST)" = hex(8):72,65,73,6F,75,72,63,65,5F,6C,69,73,74
"VALUE 0x09 (RES_DESC)" = hex(9):72,65,73,6F,75,72,63,65,5F,64,65,73,63,72,69,70,74,6F,72
"VALUE 0x0A (REQ_LIST)" = hex(A):72,65,71,75,69,72,65,6D,65,6E,74,73,5F,6C,69,73,74
"VALUE 0x0B (QWORD-LE)" = hex(B):08,00,00,00,00,00,00,00
```

Type (2) Python script using 'Hivex' library
Filename: [nr]-01-2_possible-data-types.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

child = h.node_add_child(h.root(), "0x01_TYPE2_DATA-TYPES")
assert child

values = [
    { "key"  : "VALUE 0x00 (NONE)",
      "t"    : 0,
      "value": b"\x6E\x6F\x6E\x65" },
    { "key"  : "VALUE 0x01 (SZ)",
      "t"    : 1,
      "value": u"UTF-16LE NULL-terminated string\0".encode('utf-16') },
    { "key"  : "VALUE 0x02 (EXP_SZ)",
      "t"    : 2,
      "value": u"%SystemRoot%\0".encode('utf-16') },
    { "key"  : "VALUE 0x03 (BINARY)",
      "t"    : 3,
      "value": b"\x62\x69\x6E\x61\x72\x79\x20\x64\x61\x74\x61" },
    { "key"  : "VALUE 0x04 (DWORD-LE)",
      "t"    : 4,
      "value": b"\x04\x00\x00\x00" },
    { "key"  : "VALUE 0x05 (DWORD-BE)",
      "t"    : 5,
      "value": b"\x00\x00\x00\x04" },
    { "key"  : "VALUE 0x06 (LINK)",
      "t"    : 6,
      "value": u"Symbolic link\0".encode('utf-16') },
    { "key"  : "VALUE 0x07 (MULTI_SZ)",
```

```
      "t"    : 7,
      "value": u"REG_MULTI_SZ_1\0REG_MULTI_SZ_2\0REG_MULTI_SZ_3\0".encode('utf-16') },
    { "key"  : "VALUE 0x08 (RES_LIST)",
      "t"    : 8,
      "value": b"resource_list" },
    { "key"  : "VALUE 0x09 (RES_DESC)",
      "t"    : 9,
      "value": b"resource_descriptor" },
    { "key"  : "VALUE 0x0A (REQ_LIST)",
      "t"    : 10,
      "value": b"requirements_list" },
    { "key"  : "VALUE 0x0B (QWORD-LE)",
      "t"    : 11,
      "value": b"\x08\x00\x00\x00\x00\x00\x00\x00" },
]

h.node_set_values(child, values)
h.commit(sys.argv[1])
```

2.2.2. Simple tree structure

Type (1) .REG file
Filename: [nr]-02-1_simple-tree-structure.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE]

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_1]
@ = hex:01

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_1\Node_1-1]
@ = hex:01,01

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_1\Node_1-1\Node_1-1-1]
@ = hex:01,01,01

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_1\Node_1-2]
@ = hex:01,02

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_1\Node_1-2\Node_1-2-1]
@ = hex:01,02,01

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2]
@ = hex:02

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-1]
@ = hex:02,01

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-1\Node_2-1-1]
@ = hex:02,01,01

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-1\Node_2-1-2]
@ = hex:02,01,02

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-2]
@ = hex:02,02

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-2\Node_2-2-1]
@ = hex:02,02,01

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-2\Node_2-2-1\Node_2-2-1-1]

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-2\Node_2-2-2]
@ = hex:02,02,02
```

[HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-2\Node_2-2-2\Node_2-2-2-1]

Type (2) Python script using 'Hivex' library
Filename: [nr]-02-2_simple-tree-structure.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x02_TYPE2_TREE")

node_1 = h.node_add_child(tree, "Node_1")
set_value(node_1, "", 3, b"\x01")

node_11 = h.node_add_child(node_1, "Node_1-1")
set_value(node_11, "", 3, b"\x01\x01")

node_111 = h.node_add_child(node_11, "Node_1-1-1")
set_value(node_111, "", 3, b"\x01\x01\x01")

node_12 = h.node_add_child(node_1, "Node_1-2")
set_value(node_12, "", 3, b"\x01\x02")

node_121 = h.node_add_child(node_12, "Node_1-2-1")
set_value(node_121, "", 3, b"\x01\x02\x01")

node_2 = h.node_add_child(tree, "Node_2")
set_value(node_2, "", 3, b"\x02")

node_21 = h.node_add_child(node_2, "Node_2-1")
set_value(node_21, "", 3, b"\x02\x01")

node_211 = h.node_add_child(node_21, "Node_2-1-1")
set_value(node_211, "", 3, b"\x02\x01\x01")

node_212 = h.node_add_child(node_21, "Node_2-1-2")
set_value(node_212, "", 3, b"\x02\x01\x02")

node_22 = h.node_add_child(node_2, "Node_2-2")
set_value(node_22, "", 3, b"\x02\x02")

node_221 = h.node_add_child(node_22, "Node_2-2-1")
set_value(node_221, "", 3, b"\x02\x02\x01")

node_2211 = h.node_add_child(node_221, "Node_2-2-1-1")

node_222 = h.node_add_child(node_22, "Node_2-2-2")
set_value(node_222, "", 3, b"\x02\x02\x02")

node_2221 = h.node_add_child(node_222, "Node_2-2-2-1")

h.commit(sys.argv[1])
```

2.2.3. Tree structure with the maximum levels

Type (1) .REG file
Filename: [nr]-03-1_tree-structure-with-the-maximum-levels.reg[14]

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x03_TYPE1_TREE-MAX\006\007\008\……(skip)……\510\511\512]
@ = "A Registry tree can be 512 levels deep"
```

Type (2) Python script using 'Hivex' library
Filename: [nr]-03-2_tree-structure-with-the-maximum-levels.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x03_TYPE2_TREE-MAX")
child = tree

for idx in range(3, 513):
    nk_name = '{:03d}'.format(idx)
    child = h.node_add_child(child, nk_name)

set_value(child, "", 1, "A Registry tree can be 512 levels deep\0".encode('utf-16'))

h.commit(sys.argv[1])
```

Type (3) Python script using 'Hivex' library for creating a number of levels
Filename: [nr]-03-3_tree-structure-with-a-number-of-levels.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x03_TYPE3_TREE-MAX")
child = tree

for idx in range(3, 1000001):  # One million levels deep (experimental)
```

[14] We expected that the .REG file would create 512 levels, but it was possible to create 509 levels deep only. It may be due to several virtual keys for mounting a user defined hive file. That is why the subkey of '0x03_TYPE1_TREE-MAX' key is '006' instead of '003'.

```
    nk_name = '{:03d}'.format(idx)
    child = h.node_add_child(child, nk_name)

set_value(child, "", 1, "A Registry tree do not have a limitation on levels theoretically\0".encode('utf-16'))

h.commit(sys.argv[1])
```

2.2.4. Maximum key name length

Type (1) .REG file
Filename: [nr]-04-1_maximum-key-name-length.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x04_TYPE1_KEY-NAME-MAX]
@ = "Root Node - a key name has a limit of 255 characters"

[HKEY_LOCAL_MACHINE\ROOT\0x04_TYPE1_KEY-NAME-MAX\Node_1111111…(skip)…11111111111-255]
@ = "1st Node - a key name has a limit of 255 characters"

[HKEY_LOCAL_MACHINE\ROOT\0x04_TYPE1_KEY-NAME-MAX\Node_2222222…(skip)…22222222222-255]
@ = "2nd Node - a key name has a limit of 255 characters"

[HKEY_LOCAL_MACHINE\ROOT\0x04_TYPE1_KEY-NAME-MAX\Node_222…(skip)…222-255\Node_@@@…(skip)…@@@-255]

[HKEY_LOCAL_MACHINE\ROOT\0x04_TYPE1_KEY-NAME-MAX\Node_222…(skip)…222-255\Node_$$$…(skip)…$$$-255]

[HKEY_LOCAL_MACHINE\ROOT\0x04_TYPE1_KEY-NAME-MAX\Node_3333333…(skip)…333333333333-256]
@ = "3rd Node - the maximum length of a key name is 256 characters if there is no NULL character"
```

Type (2) Python script using 'Hivex' library
Filename: [nr]-04-2_maximum-key-name-length.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x04_TYPE2_KEY-NAME-MAX")
child = tree

nk_name = "Node_"
for idx in range(0, 246): nk_name += "1"
nk_name += "-255"
node_1 = h.node_add_child(tree, nk_name)
set_value(node_1, "", 1, "1st Node - a key name has a limit of 255 characters\0".encode('utf-16'))

nk_name = "Node_"
for idx in range(0, 246): nk_name += "2"
nk_name += "-255"
node_2 = h.node_add_child(tree, nk_name)
set_value(node_2, "", 1, "2nd Node - a key name has a limit of 255 characters\0".encode('utf-16'))

nk_name = "Node_"
for idx in range(0, 246): nk_name += "@"
```

```
nk_name += "-255"
h.node_add_child(node_2, nk_name)

nk_name = "Node_"
for idx in range(0, 246): nk_name += "$"
nk_name += "-255"
h.node_add_child(node_2, nk_name)

nk_name = "Node_"
for idx in range(0, 247): nk_name += "3"
nk_name += "-256"
node_3 = h.node_add_child(tree, nk_name)
set_value(node_3, "", 1, "3rd Node - the maximum length of a key name may be 256 characters if there is
no NULL character\0".encode('utf-16'))

h.commit(sys.argv[1])
```

Type (3) Python script using 'Hivex' library for creating key name beyond the limitation
Filename: [nr]-04-3_maximum-key-name-length-beyond-limitation.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x04_TYPE3_KEY-NAME-MAX")
child = tree

nk_name = "Node_"
for idx in range(0, 248): nk_name += "7"
nk_name += "-257"
node_7 = h.node_add_child(tree, nk_name)
set_value(node_7, "", 1, "1st Node - a key name length is 257 characters\0".encode('utf-16'))

nk_name = "Node_"
for idx in range(0, 503): nk_name += "8"
nk_name += "-512"
node_8 = h.node_add_child(tree, nk_name)
set_value(node_8, "", 1, "2nd Node - a key name length is 512 characters\0".encode('utf-16'))

nk_name = "Node_"
for idx in range(0, 1014): nk_name += "9"
nk_name += "-1024"
node_9 = h.node_add_child(tree, nk_name)
set_value(node_9, "", 1, "3rd Node - a key name length is 1024 characters\0".encode('utf-16'))

h.commit(sys.argv[1])
```

2.2.5. Maximum value name length

Type (1) .REG file
Filename: [nr]-05-1_maximum-value-name-length.reg

```
Windows Registry Editor Version 5.00
```

```
[HKEY_LOCAL_MACHINE\ROOT\0x05_TYPE1_VALUE-NAME-MAX]
@ = "Root Node - a value name has a limit of 16,383 characters"

[HKEY_LOCAL_MACHINE\ROOT\0x05_TYPE1_VALUE-NAME-MAX\Value_V]
"VVVVV…(skip)…VVVVV-16383" = "V Node - a value name has a limit of 16,383 characters"

[HKEY_LOCAL_MACHINE\ROOT\0x05_TYPE1_VALUE-NAME-MAX\Value_V\Value_W]
"WWWWW…(skip)…WWWWW-16383" = "W Node - a value name has a limit of 16,383 characters"
```

Type (2) Python script using 'Hivex' library
Filename: [nr]-05-2_maximum-value-name-length.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x05_TYPE2_VALUE-NAME-MAX")
set_value(tree, "", 1, "Root Node - a value name has a limit of 16,383 characters\0".encode('utf-16'))

nk_name = "Value_V"
node_v = h.node_add_child(tree, nk_name)
vk_name = ""
for idx in range(0, 16377): vk_name += "V"
vk_name += "-16383"
set_value(node_v, vk_name, 1, "V Node - a value name has a limit of 16,383 characters\0".encode('utf-
16'))

nk_name = "Value_W"
node_w = h.node_add_child(node_v, nk_name)
vk_name = ""
for idx in range(0, 16377): vk_name += "W"
vk_name += "-16383"
set_value(node_w, vk_name, 1, "W Node - a value name has a limit of 16,383 characters\0".encode('utf-
16'))

h.commit(sys.argv[1])
```

Type (3) Python script using 'Hivex' library for creating value name beyond the limitation
Filename: [nr]-05-3_maximum-value-name-length-beyond-limitation.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })
```

```
tree = h.node_add_child(h.root(), "0x05_TYPE3_VALUE-NAME-MAX")
set_value(tree, "", 1, "Root Node - a value name has a limit of 16,383 characters\0".encode('utf-16'))

nk_name = "Value_Y"
node_y = h.node_add_child(tree, nk_name)
vk_name = ""
for idx in range(0, 16378): vk_name += "Y"
vk_name += "-16384"
set_value(node_y, vk_name, 1, "Y Node - a value name length is 16,384 characters\0".encode('utf-16'))

nk_name = "Value_Z"
node_z = h.node_add_child(node_y, nk_name)
vk_name = ""
for idx in range(0, 32760): vk_name += "Z"
vk_name += "-32766"
set_value(node_z, vk_name, 1, "Z Node - a value name length is 32,766 characters\0".encode('utf-16'))

h.commit(sys.argv[1])
```

2.2.6. Big-data (> 16,344 bytes)

Type (1) .REG file
Filename: [nr]-06-1_big-data.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x06_TYPE1_BIG-DATA]
"BINARY 16344" = hex(3):41,41…(total 16,344 bytes)…41,41"
"BINARY 16345" = hex(3):42,42…(total 16,345 bytes)…42,42"
"BINARY 20440" = hex(3):43,43…(total 20,440 bytes = 16344 +  4096)…43,43"
"BINARY 32688" = hex(3):44,44…(total 32,668 bytes = 16344 + 16344)…44,44"
"BINARY 1MB-4" = hex(3):45,45…(total 1,048,572 bytes = 1048576(1MiB)-4)…45,45"
"BINARY 1MB-3" = hex(3):46,46…(total 1,048,573 bytes = 1048576(1MiB)-3)…46,46"
```

Type (2) Python script using 'Hivex' library
Filename: [nr]-06-2_big-data.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x06_TYPE2_BIG-DATA")

vk_name = "BINARY 16344"
vk_data = b""
for idx in range(0, 16344): vk_data += "\x41"
set_value(tree, vk_name, 3, vk_data)

vk_name = "BINARY 16345"
vk_data = b""
for idx in range(0, 16345): vk_data += "\x42"
set_value(tree, vk_name, 3, vk_data)
```

```
vk_name = "BINARY 20440"
vk_data = b""
for idx in range(0, 20440): vk_data += "\x43"
set_value(tree, vk_name, 3, vk_data)

vk_name = "BINARY 32688"
vk_data = b""
for idx in range(0, 32688): vk_data += "\x44"
set_value(tree, vk_name, 3, vk_data)

# hivex-internal.h (line 329)
# define HIVEX_MAX_ALLOCATION  1000000
vk_name = "BINARY 1000000-4"
vk_data = b""
for idx in range(0, 1000000-4): vk_data += "\x45"
set_value(tree, vk_name, 3, vk_data)

h.commit(sys.argv[1])
```

2.2.7. Non-ASCII characters

Type (1) .REG file
Filename: [nr]-07-1_non-ascii-characters.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII]
"TEST 1" = "Registry hive structure parsing"
"TEST 2" = "Non-ASCII characters"

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\Hello]
@ = "English"
"Hello" = "Hello : English"

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\¡Hola!]
@ = "Spanish"
"¡Hola!" = "¡Hola! : Spanish"

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\안녕하세요]
@ = "Korean"
"안녕하세요" = "안녕하세요 : Korean"

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\Здравствуйте]
@ = "Russia"
"Здравствуйте" = "Здравствуйте : Russia"

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\您好]
@ = "Chinese"
"您好" = "您好 : Chinese"

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\こんにちは]
@ = "Japanese"
"こんにちは" = "こんにちは : Japanese"

[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\नमस्ते]
@ = "Hindi"
"नमस्ते" = "नमस्ते : Hindi"
```

Type (2) Python script using 'Hivex' library
Filename: [nr]-07-2_non-ascii-characters.py

```
import sys
import os
```

```
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

tree = h.node_add_child(h.root(), "0x07_TYPE1_NON-ASCII")
set_value(tree, "TEST 1", 1, u"Registry hive structure parsing\0".encode('utf-16'))
set_value(tree, "TEST 2", 1, u"Non-ASCII characters\0".encode('utf-16'))

items = [
    { "word"    : u"Hello",
      "language": u"English\0" },
    { "word"    : u"¡Hola!",
      "language": u"Spanish\0" },
    { "word"    : u"안녕하세요",
      "language": u"Korean\0" },
    { "word"    : u"Здравствуйте",
      "language": u"Russian\0" },
    { "word"    : u"您好",
      "language": u"Chinese\0" },
    { "word"    : u"こんにちは",
      "language": u"Japanese\0" },
    { "word"    : u"नमस्ते",
      "language": u"Hindi\0" },
]

for item in items:
    name = item['word'].encode('utf-8')
    data = item['word'] + " : " + item['language']
    data = data.encode('utf-16')
    node = h.node_add_child(tree, name)
    set_value(node, "", 1, item['language'].encode('utf-16'))
    set_value(node, name, 1, data)

h.commit(sys.argv[1])
```

2.2.8. Naming Convention

Type (1) Python script
Filename: [nr]-08_key-value-naming-convention.py

```
import winreg

PRE_DEFINED = winreg.HKEY_LOCAL_MACHINE

def set_reg(key_path, value_name="", value_type="", value_data=""):
    try:
        winreg.CreateKey(PRE_DEFINED, key_path)
        if value_name != "":
            key = winreg.OpenKey(PRE_DEFINED, key_path, 0, winreg.KEY_WRITE)
            winreg.SetValueEx(key, value_name, 0, value_type, value_data)
            winreg.CloseKey(key)
    except WindowsError:
        print("set_reg(): Error detected.")
        return False
    return True
```

```
SEP = "\\"
REG_SZ = winreg.REG_SZ
REG_BINARY = winreg.REG_BINARY
base = "ROOT\\0x08_NAMING-CONVENTION\\"
set_reg(base)

# ============================================================================
# If the name length field of a 'key value' (vk) structure is ZERO,
# a tool may handle it as a default value.
# In the case of RegEdit.exe, it will be printed as '(Default)'.
key = base + "0x01_Default-Value\\"

set_reg(key, None, REG_SZ, "Value name is \'NULL\' (= Name length is 0)")
set_reg(key, "(Default)", REG_SZ, "Value name is \'(Default)\' (= Name length is 9)")

# ============================================================================
# Unlike file systems, the Windows registry allows a key to have
# a subkey and a value sharing an identical name.
# < Expected Structure > ------------------------
#   +- [NK] 0x08_NAMING-CONVENTION
#       +- [NK] 0x02_Identical_Key_and_Value_Names
#           +- [VK] CFTT ("[Level 1]")
#           +- [NK] CFTT
#               +- [VK] CFTT ("[Level 2]")
#               +- [NK] CFTT
#                   +- [VK] CFTT ("[Level 3]")
#                   +- [NK] CFTT
key = base + "0x02_Identical-Key-and-Value-Names\\"

set_reg(key, "CFTT", REG_SZ, "[Level 1] CFTT")
set_reg(key + "CFTT", "CFTT", REG_SZ, "[Level 2] CFTT")
set_reg(key + "CFTT\\CFTT", "CFTT", REG_SZ, "[Level 3] CFTT")
set_reg(key + "CFTT\\CFTT\\CFTT")  # There is no value

# ============================================================================
# The slash(/) is allowed to be used for naming keys and values.
# Note that the backslash(\) is not allowed for naming keys.
key = base + "0x03_Slash\\"

set_reg(key, "/", REG_SZ, "[Level 1] Forward slash x1 \"/\"")
set_reg(key + "/", "/", REG_SZ, "[Level 2] Forward slash x1 \"/\"")
set_reg(key + "/" + SEP + "/subkey1", "/", REG_SZ, "[Level 3] Forward slash x1 \"/\" of subkey1")
set_reg(key + "/" + SEP + "/subkey2", "/", REG_SZ, "[Level 3] Forward slash x1 \"/\" of subkey2")

set_reg(key, "//", REG_SZ, "[Level 1] Forward slash x2 \"//\"")
set_reg(key + "//", "//", REG_SZ, "[Level 2] Forward slash x2 \"//\"")
set_reg(key + "//" + SEP + "//subkey1", "//", REG_SZ, "[Level 3] Forward slash x2 \"//\" of subkey1")
set_reg(key + "//" + SEP + "//subkey2", "//", REG_SZ, "[Level 3] Forward slash x2 \"//\" of subkey2")

set_reg(key, "///", REG_SZ, "[Level 1] Forward slash x3 \"///\"")
set_reg(key + "///", "///", REG_SZ, "[Level 2] Forward slash x3 \"///\"")
set_reg(key + "///" + SEP + "///subkey1", "///", REG_SZ, "[Level 3] Forward slash x3 \"///\" of subkey1")
set_reg(key + "///" + SEP + "///subkey2", "///", REG_SZ, "[Level 3] Forward slash x3 \"///\" of subkey2")

set_reg(key, "\\", REG_SZ, "[Level 1] Backslash x1 \"\\\"")
set_reg(key, "\\\\", REG_SZ, "[Level 1] Backslash x2 \"\\\\\"")
set_reg(key, "\\\\\\", REG_SZ, "[Level 1] Backslash x3 \"\\\\\\\"")

# ============================================================================
# The dot(.) and double dots(..) are allowed to be used for naming keys and values.
key = base + "0x04_Dot\\"

set_reg(key, ".", REG_SZ, "[Level 1] Dot \".\"")
set_reg(key + ".", ".", REG_SZ, "[Level 2] Dot \".\"")
set_reg(key + "." + SEP + ".subkey1", ".", REG_SZ, "[Level 3] Dot \".\" of subkey1")
set_reg(key + "." + SEP + ".subkey2", ".", REG_SZ, "[Level 3] Dot \".\" of subkey2")

set_reg(key, "..", REG_SZ, "[Level 1] Double dots \"..\"")
set_reg(key + "..", "..", REG_SZ, "[Level 2] Double dots \"..\"")
set_reg(key + ".." + SEP + "..subkey1", "..", REG_SZ, "[Level 3] Double dots \"..\" of subkey1")
set_reg(key + ".." + SEP + "..subkey2", "..", REG_SZ, "[Level 3] Double dots \"..\" of subkey2")
```

```
set_reg(key + ".././..", ".././..", REG_SZ, "Multiple dots and slashes \".././..\"")

# ============================================================================
# ASCII characters are allowed to be used for naming keys and values.
# Note that 0x00(NULL) and 0x5C(backslash) are not allowed for naming keys.
# In the range of 0x81~0xFF, they probably need to be handled according to ISO/IEC 8859-1.
key = base + "0x05_ASCII-256-Characters\\"

for i in range(0, 0xFF+1):
    if i == 0x00 or i == 0x5C:  # excluding 0x00 and backslash
        continue
    s = "ASCII_0x{0:02X}_({1:c})".format(i, i)
    set_reg(key + s, None,
            REG_SZ, "ASCII \"{0:c}\" (0x{0:02X})".format(i))
    set_reg(key + s, "(Binary Data of \"{0:c}\")".format(i),
            REG_BINARY, i.to_bytes(1, byteorder='little'))
    set_reg(key + s, "{0:c}".format(i),
            REG_SZ, "ASCII \"{0:c}\"".format(i))
    set_reg(key + s, "{0:c}{0:c}".format(i),
            REG_SZ, "ASCII \"{0:c}{0:c}\"".format(i))
    set_reg(key + s, "{0:c} ".format(i),
            REG_SZ, "ASCII \"{0:c}(space)\"".format(i))
    set_reg(key + s, " {0:c}".format(i),
            REG_SZ, "ASCII \"(space){0:c}\"".format(i))

    set_reg(key + s + SEP + "{0:c}".format(i))
    set_reg(key + s + SEP + "{0:c}{0:c}".format(i))

# ============================================================================
# UTF-16LE characters are allowed to be used for naming keys and values.
# For naming keys, '☆' (0x2606) is used for enforcing to use UTF-16LE encoding.
# Note that the value names in the ASCII range(0x00~0xFF) are stored as ASCII characters.
# Because those ASCII value names are already covered by the '0x04_ASCII-256-Characters' class,
# this class will focus on the range of 0x0100~0x0400.
key = base + "0x06_UTF-16LE-First-1024-Characters\\"

for i in range(0, 0x0400+1):
    if i == 0x00 or i == 0x5C:  # excluding 0x00 and backslash
        continue
    s = "UTF-16LE_☆_0x{0:04X}_({1:s})".format(i, chr(i))  # '☆' for enforcing to use UTF-16LE
    set_reg(key + s, None,
            REG_SZ, "UTF-16LE \"{0:s}\" (0x{1:04X})".format(chr(i), i))
    set_reg(key + s, "(Binary Data of \"{0:s}\")".format(chr(i)),
            REG_BINARY, i.to_bytes(2, byteorder='little'))
    set_reg(key + s, "{0:s}".format(chr(i)),
            REG_SZ, "UTF-16LE \"{0:s}\"".format(chr(i)))
    set_reg(key + s, "{0:s}{0:s}".format(chr(i)),
            REG_SZ, "UTF-16LE \"{0:s}{0:s}\"".format(chr(i)))
    set_reg(key + s, "{0:s}{1:s}{2:s}".format(chr(i), chr(i + 1), chr(i + 2)),
            REG_SZ, "UTF-16LE \"{0:s}{1:s}{2:s}\"".format(chr(i), chr(i + 1), chr(i + 2)))
    set_reg(key + s, "{0:s} ".format(chr(i)),
            REG_SZ, "UTF-16LE \"{0:s}(space)\"".format(chr(i)))
    set_reg(key + s, " {0:s}".format(chr(i)),
            REG_SZ, "UTF-16LE \"(space){0:s}\"".format(chr(i)))

    set_reg(key + s + SEP + "{0:s}".format(chr(i)))
    set_reg(key + s + SEP + "{0:s}{0:s}".format(chr(i)))
    set_reg(key + s + SEP + "{0:s}{1:s}{2:s}".format(chr(i), chr(i + 1), chr(i + 2)))

# ============================================================================
# The backslash(\) character is used to escape characters that have a special meaning.
# The followings are considered here:
#   \a (07): Bell
#   \b (08): Backspace
#   \t (09): Horizontal tap
#   \n (10): Line feed
#   \v (11): Vertical tap
#   \f (12): Form feed
#   \r (13): Carriage return
# Note that although the '0x04_ASCII-256-Characters' class already includes those characters,
# this class will be used to reveal how forensic tools handle escape sequences in more detail.
key = base + "0x07_Escape-Sequences\\"
```

```
set_reg(key + "\abell\a", "\abell\a",
        REG_SZ, "Bell (\\a) \"\abell\a\"")
set_reg(key + "\bbackspace\b", "\bbackspace\b",
        REG_SZ, "Backspace (\\b) \"\bbackspace\b\"")
set_reg(key + "\thorizontal-tap\t", "\thorizontal-tap\t",
        REG_SZ, "Horizontal tap (\\t) \"\thorizontal-tap\t\"")
set_reg(key + "\nline-feed\n", "\nline-feed\n",
        REG_SZ, "Line feed (\\n) \"\nline-feed\n\"")
set_reg(key + "\vvertical-tap\v", "\vvertical-tap\v",
        REG_SZ, "Vertical tap (\\v) \"\vvertical-tap\v\"")
set_reg(key + "\fform-feed\f", "\fform-feed\f",
        REG_SZ, "Form feed (\\f) \"\fform-feed\f\"")
set_reg(key + "\rcarriage-return\r", "\rcarriage-return\r",
        REG_SZ, "Carriage return (\\r) \"\rcarriage-return\r\"")
```

2.3. GENERATION METHODS FOR CATEGORY #2 (NORMAL REGISTRY HIVE FILE WITH DELETED REGISTRY DATA)

2.3.1. Delete keys with values, but without subkeys

Type (1) .REG file
Filename: [nrd]-01-1_delete-keys-with-values-but-without-subkeys.reg

```
Windows Registry Editor Version 5.00

[-HKEY_LOCAL_MACHINE\ROOT\0x01_TYPE1_DATA-TYPES]

[-HKEY_LOCAL_MACHINE\ROOT\0x06_TYPE1_BIG-DATA]
```

Type (2) Python script using 'Hivex' library
Filename: [nrd]-01-2_delete-keys-with-values-but-without-subkeys.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

child = h.node_get_child(h.root(), "0x01_TYPE2_DATA-TYPES")
h.node_delete_child(child)

child = h.node_get_child(h.root(), "0x06_TYPE2_BIG-DATA")
h.node_delete_child(child)

h.commit(sys.argv[1])
```

2.3.2. Delete keys with values and subkeys

Type (1) .REG file
Filename: [nrd]-02-1_delete-keys-with-values-and-subkeys.reg

```
Windows Registry Editor Version 5.00

[-HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_1]

[-HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2]

[-HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII]
```

Type (2) Python script using 'Hivex' library
Filename: [nrd]-02-2_delete-keys-with-values-and-subkeys.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

child = h.node_get_child(h.root(), "0x02_TYPE2_TREE")
child = h.node_get_child(child, "Node_1")
h.node_delete_child(child)

child = h.node_get_child(h.root(), "0x02_TYPE2_TREE")
child = h.node_get_child(child, "Node_2")
```

```
h.node_delete_child(child)

child = h.node_get_child(h.root(), "0x07_TYPE2_NON-ASCII")
h.node_delete_child(child)

h.commit(sys.argv[1])
```

2.3.3. Delete keys without values and subkeys

Type (1) .REG file
Filename: [nrd]-03-1_delete-keys-without-values-and-subkeys.reg

```
Windows Registry Editor Version 5.00

[-HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-2\Node_2-2-1\Node_2-2-1-1]

[-HKEY_LOCAL_MACHINE\ROOT\0x02_TYPE1_TREE\Node_2\Node_2-2\Node_2-2-2\Node_2-2-2-1]

[-HKEY_LOCAL_MACHINE\ROOT\0x04_TYPE1_KEY-NAME-MAX\Node_22…(skip)…22-255\Node_@@…(skip)…@@-255]
```

Type (2) Python script using 'Hivex' library
Filename: [nrd]-03-2_delete-keys-without-values-and-subkeys.py

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

child = h.node_get_child(h.root(), "0x02_TYPE2_TREE")
child = h.node_get_child(child, "Node_2")
child = h.node_get_child(child, "Node_2-2")
child = h.node_get_child(child, "Node_2-2-1")
child = h.node_get_child(child, "Node_2-2-1-1")
h.node_delete_child(child)

child = h.node_get_child(h.root(), "0x02_TYPE2_TREE")
child = h.node_get_child(child, "Node_2")
child = h.node_get_child(child, "Node_2-2")
child = h.node_get_child(child, "Node_2-2-2")
child = h.node_get_child(child, "Node_2-2-2-1")
h.node_delete_child(child)

child = h.node_get_child(h.root(), "0x04_TYPE2_KEY-NAME-MAX")
child = h.node_get_child(child, "Node_22222…(skip)…22222-255")
child = h.node_get_child(child, "Node_@@@@@…(skip)…@@@@@-255")
h.node_delete_child(child)

h.commit(sys.argv[1])
```

2.3.4. Delete values with normal data

Type (1) .REG file
Filename: [nrd]-04-1_delete-a-value-with-normal-data.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x05_TYPE1_VALUE-NAME-MAX\Value_V]
"VVVVV…(skip)…VVVVV-16383" =-
```

```
[HKEY_LOCAL_MACHINE\ROOT\0x07_TYPE1_NON-ASCII\Hello]
"Hello" =-
```

*** Python script using 'Hivex' library**

```
- This test could not be performed because Hivex library not supports the value deletion.
  (libguestfs-tools (v1.32.2) + python-hivex (v1.3.13))
```

2.3.5. Delete values with big data

Type (1) .REG file
Filename: [nrd]-05-1_delete-a-value-with-big-data.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x06_TYPE1_BIG-DATA]
"BINARY 16345" =-
"BINARY 20440" =-
```

*** Python script using 'Hivex' library**

```
- This test could not be performed because Hivex library not supports the value deletion.
  (libguestfs-tools (v1.32.2) + python-hivex (v1.3.13))
```

2.3.6. Delete multiple values in a key

Type (1) .REG file
Filename: [nrd]-06-1_delete-multiple-values-in-a-key.reg

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x01_TYPE1_DATA-TYPES]
"VALUE 0x00 (NONE)"     =-
"VALUE 0x01 (SZ)"       =-
"VALUE 0x02 (EXP_SZ)"   =-
"VALUE 0x03 (BINARY)"   =-
"VALUE 0x04 (DWORD-LE)" =-
"VALUE 0x05 (DWORD-BE)" =-
"VALUE 0x06 (LINK)"     =-
"VALUE 0x07 (MULTI_SZ)" =-
"VALUE 0x08 (RES_LIST)" =-
"VALUE 0x09 (RES_DESC)" =-
"VALUE 0x0A (REQ_LIST)" =-
"VALUE 0x0B (QWORD-LE)" =-
```

*** Python script using 'Hivex' library**

```
- This test could not be performed because Hivex library not supports the value deletion.
  (libguestfs-tools (v1.32.2) + python-hivex (v1.3.13))
```

2.3.7. Change normal data and remain original size

Type (1) .REG file
Filename: [nrd]-07-1_change-normal-data-and-remain-original-size.reg

```
* Refer to the script for creating data on Section 2.2.1. Possible data types – Type (1)

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x01_TYPE1_DATA-TYPES]
```

```
"VALUE 0x01 (SZ)"       = "UTF-EEEE NULL-terminated string"
"VALUE 0x03 (BINARY)"   = hex(3):62,69,EE,EE,EE,EE,20,64,61,74,61
```

Type (2) Python script using 'Hivex' library
Filename: [nrd]-07-2_change-normal-data-and-remain-original-size.py

* Refer to the script for creating data on Section 2.2.1. Possible data types – Type (2)

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

child = h.node_get_child(h.root(), "0x01_TYPE2_DATA-TYPES")

vk_name = "VALUE 0x01 (SZ)"
vk_data = u"UTF-EEEE NULL-terminated string\0".encode('utf-16')
set_value(child, vk_name, 1, vk_data)

vk_name = "VALUE 0x03 (BINARY)"
vk_data = b"\x62\x69\xEE\xEE\xEE\xEE\x20\x64\x61\x74\x61"
set_value(child, vk_name, 3, vk_data)

h.commit(sys.argv[1])
```

2.3.8. Change normal data to smaller size

Type (1) .REG file
Filename: [nrd]-08-1_change-normal-data-to-smaller-size.reg

* Refer to the script for creating data on Section 2.2.1. Possible data types – Type (1)

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x01_TYPE1_DATA-TYPES]
"VALUE 0x01 (SZ)"       = "UTF-EE NULL-terminated string"
"VALUE 0x03 (BINARY)"   = hex(3):62,69,EE,EE,20,64,61,74,61
```

Type (2) Python script using 'Hivex' library
Filename: [nrd]-08-2_change-normal-data-to-smaller-size.py

* Refer to the script for creating data on Section 2.2.1. Possible data types – Type (2)

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
```

```
    })

child = h.node_get_child(h.root(), "0x01_TYPE2_DATA-TYPES")

vk_name = "VALUE 0x01 (SZ)"
vk_data = u"UTF-EE NULL-terminated string\0".encode('utf-16')
set_value(child, vk_name, 1, vk_data)

vk_name = "VALUE 0x03 (BINARY)"
vk_data = b"\x62\x69\xEE\xEE\x20\x64\x61\x74\x61"
set_value(child, vk_name, 3, vk_data)

h.commit(sys.argv[1])
```

2.3.9. Change normal data to larger size

Type (1) .REG file
Filename: [nrd]-09-1_change-normal-data-to-larger-size.reg

```
* Refer to the script for creating data on Section 2.2.1. Possible data types – Type (1)
```

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x01_DATA-TYPES]
"VALUE 0x01 (SZ)"       = "UTF-16LE NULL-terminated string UNICODE"
"VALUE 0x03 (BINARY)"   = hex(3):62,69,6E,61,72,79,20,64,61,74,61,20,55,4E,49,43,4F,44,45
```

Type (2) Python script using 'Hivex' library
Filename: [nrd]-09-2_change-normal-data-to-larger-size.py

```
* Refer to the script for creating data on Section 2.2.1. Possible data types – Type (2)
```

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

child = h.node_get_child(h.root(), "0x01_TYPE2_DATA-TYPES")

vk_name = "VALUE 0x01 (SZ)"
vk_data = u"UTF-16LE NULL-terminated string UNICODE\0".encode('utf-16')
set_value(child, vk_name, 1, vk_data)

vk_name = "VALUE 0x03 (BINARY)"
vk_data = b"\x62\x69\x6E\x61\x72\x79\x20\x64\x61\x74\x61\x20\x55\x4E\x49\x43\x4F\x44\x45"
set_value(child, vk_name, 3, vk_data)

h.commit(sys.argv[1])
```

2.3.10. Change big data to smaller size

Type (1) .REG file
Filename: [nrd]-10-1_change-big-data-to-smaller-size.reg

```
* Refer to the script for creating data on Section 2.2.6. Big-data – Type (1)
```

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\ROOT\0x06_TYPE1_BIG-DATA]
"BINARY 20440" = hex(3):EE,EE,EE,EE,EE,EE,EE,EE,EE,EE,EE
```

Type (2) Python script using 'Hivex' library
Filename: [nrd]-10-2_change-big-data-to-smaller-size.py

* Refer to the script for creating data on Section 2.2.6. Big-data – Type (2)

```
import sys
import os
import hivex

assert not (len(sys.argv) != 2)
h = hivex.Hivex (sys.argv[1], verbose = True, debug = True, write = True)
assert h

def set_value(nk, vk_name, vk_type, vk_data):
    global h
    h.node_set_value(nk, {
        "key":   vk_name,
        "t":     vk_type,
        "value": vk_data
    })

child = h.node_get_child(h.root(), "0x06_TYPE2_BIG-DATA")

vk_name = "BINARY 20440"
vk_data = b""
for idx in range(0, 11): vk_data += "\xEE"
set_value(child, vk_name, 3, vk_data)

h.commit(sys.argv[1])
```

2.3.11. Change key name[15] and remain original size

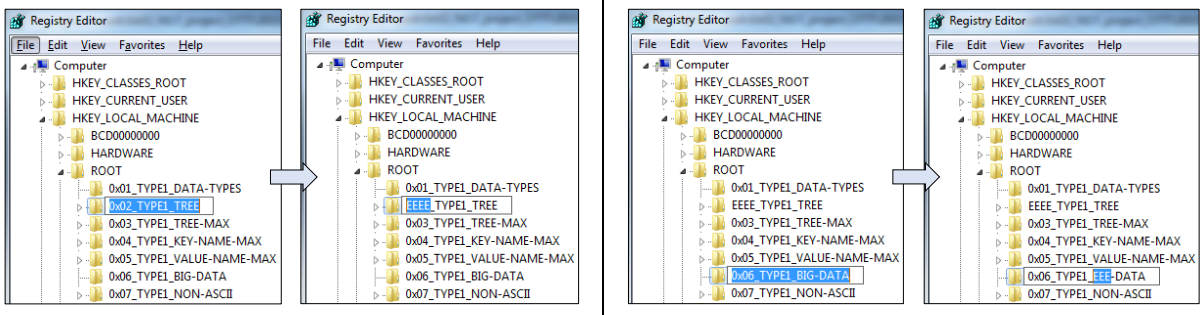
Type (1) PowerShell script
Filename: [nrd]-11-1_change-key-name-and-remain-original-size.ps1

* Refer to the script for creating data on Section 2.2.2 and Section 2.2.6.

```
Rename-Item "HKLM:\ROOT\0x02_TYPE1_TREE" -NewName "EEEE_TYPE1_TREE"

Rename-Item "HKLM:\ROOT\0x06_TYPE1_BIG-DATA" -NewName "0x06_TYPE1_EEE-DATA"
```

Type (2) Manual Editing with Windows Registry Editor (RegEdit.exe)

* Refer to the script for creating data on Section 2.2.2 and Section 2.2.6.



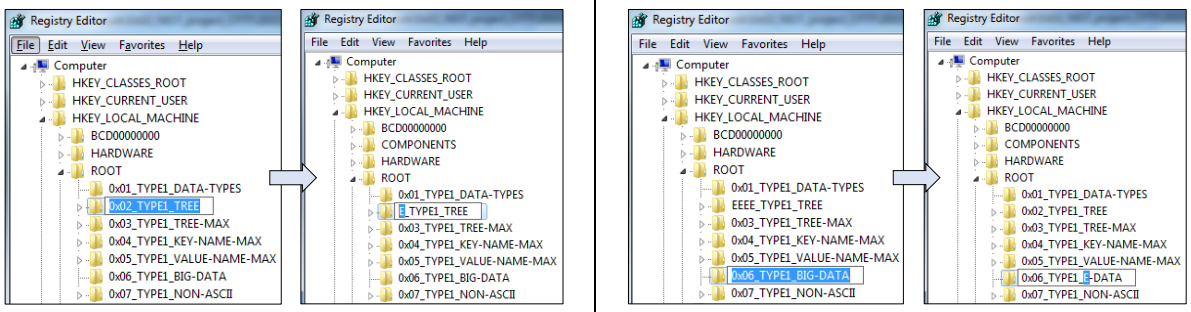
[15] We performed two types of experiments using Windows PowerShell scripts and Windows registry editor (RegEdit.exe) for renaming keys and values because the renaming feature is not supported by a .REG script and an external library 'Hivex'.

2.3.12. Change key name to smaller size

Type (1) PowerShell script
Filename: [nrd]-12-1_change-key-name-to-smaller-size.ps1
* Refer to the script for creating data on Section 2.2.2 and Section 2.2.6.

```
Rename-Item "HKLM:\ROOT\0x02_TYPE1_TREE" -NewName "E_TYPE1_TREE"

Rename-Item "HKLM:\ROOT\0x06_TYPE1_BIG-DATA" -NewName "0x06_TYPE1_E-DATA"
```

Type (2) Manual Editing with Windows Registry Editor (RegEdit.exe)
* Refer to the script for creating data on Section 2.2.2 and Section 2.2.6.

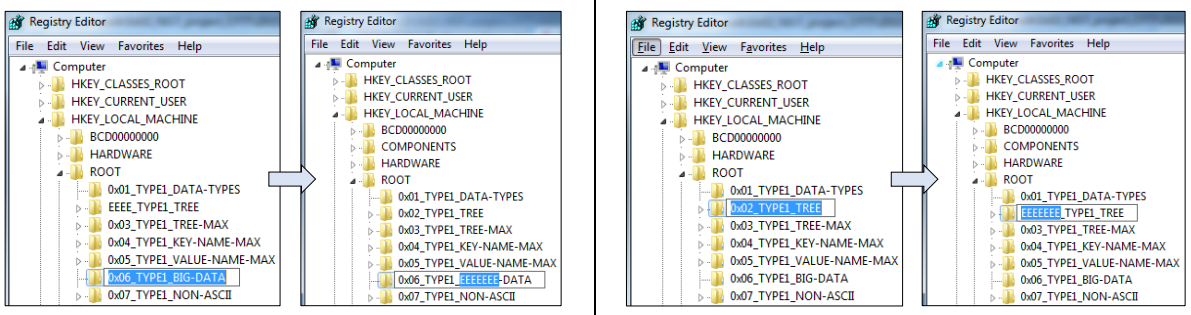


2.3.13. Change key name to larger size

Type (1) PowerShell script
Filename: [nrd]-13-1_change-key-name-to-larger-size.ps1
* Refer to the script for creating data on Section 2.2.2 and Section 2.2.6.

```
Rename-Item "HKLM:\ROOT\0x06_TYPE1_BIG-DATA" -NewName "0x06_TYPE1_EEEEEEE-DATA"

Rename-Item "HKLM:\ROOT\0x02_TYPE1_TREE" -NewName "EEEEEEE_TYPE1_TREE"
```

Type (2) Manual Editing with Windows Registry Editor (RegEdit.exe)
* Refer to the script for creating data on Section 2.2.2 and Section 2.2.6.



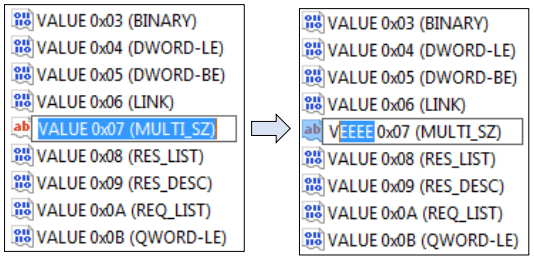
2.3.14. Change value name and remain original size

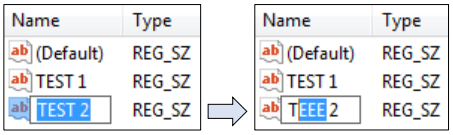
Type (1) PowerShell script
Filename: [nrd]-14-1_change-value-name-and-remain-original-size.ps1
* Refer to the script for creating data on Section 2.2.1 and Section 2.2.7.

```
Rename-ItemProperty -Path "HKLM:\ROOT\0x01_TYPE1_DATA-TYPES" -Name "VALUE 0x07 (MULTI_SZ)" -NewName "VEEEE 0x07 (MULTI_SZ)"

Rename-ItemProperty -Path "HKLM:\ROOT\0x07_TYPE1_NON-ASCII" -Name "TEST 2" -NewName "TEEE 2"
```

Type (2) Manual Editing with Windows Registry Editor (RegEdit.exe)

* Refer to the script for creating data on Section 2.2.1 and Section 2.2.7.





2.3.15. Change value name to smaller size

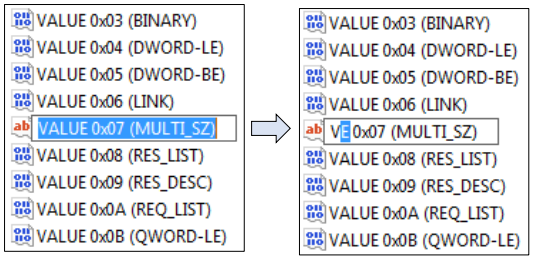
Type (1) PowerShell script
Filename: [nrd]-15-1_change-value-name-to-smaller-size.ps1

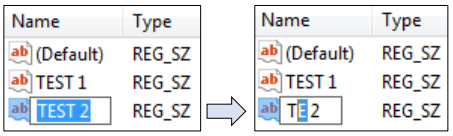
* Refer to the script for creating data on Section 2.2.1 and Section 2.2.7.

```
Rename-ItemProperty -Path "HKLM:\ROOT\0x01_TYPE1_DATA-TYPES" -Name "VALUE 0x07 (MULTI_SZ)" -NewName "VE 0x07 (MULTI_SZ)"

Rename-ItemProperty -Path "HKLM:\ROOT\0x07_TYPE1_NON-ASCII" -Name "TEST 2" -NewName "TE 2"
```

Type (2) Manual Editing with Windows Registry Editor (RegEdit.exe)

* Refer to the script for creating data on Section 2.2.1 and Section 2.2.7.





2.3.16. Change value name to larger size

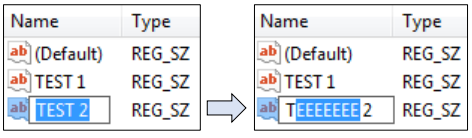
Type (1) PowerShell script
Filename: [nrd]-16-1_change-value-name-to-larger-size.ps1

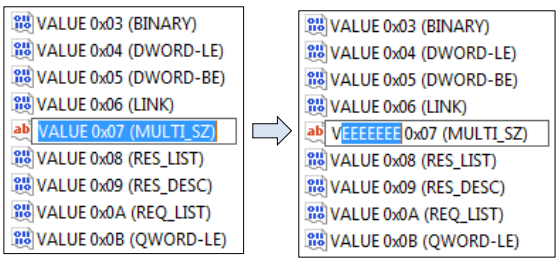
* Refer to the script for creating data on Section 2.2.1 and Section 2.2.7.

```
Rename-ItemProperty -Path "HKLM:\ROOT\0x07_TYPE1_NON-ASCII" -Name "TEST 2" -NewName "TEEEEEEE 2"

Rename-ItemProperty -Path "HKLM:\ROOT\0x01_TYPE1_DATA-TYPES" -Name "VALUE 0x07 (MULTI_SZ)" -NewName "VEEEEEEE 0x07 (MULTI_SZ)"
```

Type (2) Manual Editing with Windows Registry Editor (RegEdit.exe)

* Refer to the script for creating data on Section 2.2.1 and Section 2.2.7.





2.4. GENERATION METHODS FOR CATEGORY #3 (CORRUPTED REGISTRY HIVE FILE)

2.4.1. A hive bin with root key

Refer to the script '[cr]-01_a-hive-bin-with-root-key.py' and Section 2.1.3.

2.4.2. A hive bin randomly selected

Refer to the script '[cr]-02_a-hive-bin-randomly-selected.py' and Section 2.1.3.

2.4.3. Last half

Refer to the script '[cr]-03_last-half.py' and Section 2.1.3.

2.4.4. Fragments with hive bin header randomly selected

Refer to the script '[cr]-04_fragments-with-hive-bin-header-randomly-selected' and Section 2.1.3.

2.4.5. Hive header

Refer to the script '[cr]-05_hive-header' and Section 2.1.3.

2.4.6. First half

Refer to the script '[cr]-06_first-half.py' and Section 2.1.3.

2.4.7. First and last quarter

Refer to the script '[cr]-07_first-and-last-quarter.py' and Section 2.1.3.

2.5. GENERATION METHODS FOR CATEGORY #4 (MANIPULATED REGISTRY HIVE FILE)

2.5.1. Data hiding

2.5.1.1. Hide a root key

```
Refer to the scripts
          ‘[mr]-01.(1)_hide-a-root-key.py’
          ‘[mr]-01.(2)_hide-a-root-key.py’
         and Section 2.1.4.
```

2.5.1.2. Hide key names

```
Refer to the scripts
          ‘[mr]-02.(1)_hide-key-names.py’
          ‘[mr]-02.(2)_hide-key-names.py’
         and Section 2.1.4.
```

2.5.1.3. Hide subkeys of a key

```
Refer to the scripts
          ‘[mr]-03.(1)_hide-subkeys-of-a-key.py’
          ‘[mr]-03.(2)_hide-subkeys-of-a-key.py’
          ‘[mr]-03.(3)_hide-subkeys-of-a-key.py’
          ‘[mr]-03.(4)_hide-subkeys-of-a-key.py’
         and Section 2.1.4.
```

2.5.1.4. Hide values of a key

```
Refer to the scripts
          ‘[mr]-04.(1)_hide-values-of-a-key.py’
          ‘[mr]-04.(2)_hide-values-of-a-key.py’
          ‘[mr]-04.(3)_hide-values-of-a-key.py’
         and Section 2.1.4.
```

2.5.1.5. Hide value names

```
Refer to the scripts
          ‘[mr]-05.(1)_hide-value-names.py’
          ‘[mr]-05.(2)_hide-value-names.py’
         and Section 2.1.4.
```

2.5.1.6. Hide data of a value

```
Refer to the scripts
          ‘[mr]-06.(1)_hide-data-of-a-value.py’
          ‘[mr]-06.(2)_hide-data-of-a-value.py’
```

```
          '[mr]-06.(3)_hide-data-of-a-value.py'
          '[mr]-06.(4)_hide-data-of-a-value.py'
        and Section 2.1.4.
```

2.5.1.7. Hide big data (> 16,344 bytes) of a value

```
Refer to the script '[mr]-07_hide-big-data-of-a-value.py' and Section 2.1.4.
```

2.5.2. Infinite loop

2.5.2.1. Key loop

```
Refer to the script '[mr]-08_key-loop.py' and Section 2.1.4.
```

2.5.3. Invalid data size

2.5.3.1. Integer data too large

```
Refer to the script '[mr]-09_integer-data-too-large.py' and Section 2.1.4.
```

2.5.3.2. Binary data too large

```
Refer to the script '[mr]-10_binary-data-too-large.py' and Section 2.1.4.
```

2.5.3.3. String data too large

```
Refer to the script '[mr]-11_string-data-too-large.py' and Section 2.1.4.
```

2.5.4. Version mismatch

2.5.4.1. Big data management

```
Refer to the script '[mr]-12_big-data-management.py' and Section 2.1.4.
```

2.5.5. Ambiguous encoding

2.5.5.1. Key name flag

```
Refer to the script '[mr]-13_key-name-flag.py' and Section 2.1.4.
```

2.5.5.2. Value name flag

```
Refer to the script '[mr]-14_value-name-flag.py' and Section 2.1.4.
```

2.5.5.3. Different encodings

```
Refer to the script '[mr]-15_different-encodings.py' and Section 2.1.4.
```

2.6. INTEGRATED GENERATION OF USER-GENERATED REFERENCE DATA

We offer an integrated automated execution method (a shell script) to execute fully automated assistance tools at one time. The following indicates pre-requirements for each operating system used in this work.

Requirements

```
- Windows environment
   ✓ PowerShell
   ✓ Python 2.7 (c:\python27\python.exe)
   ✓ Python 3.4 (c:\python34\python.exe)

- Linux environment
   ✓ Python 2.7
   ✓ libguestfs-tools (including hivex library)
   ✓ python-hivex (python binding for hivex)
```

In Windows environment, a Windows batch script file ('ug-automation-windows.bat') will create user-generated reference registry hive files based on the generation strategies described above Sections. Note that six hive files of the 'NRD' category associated with the type #2 methods from Sections 2.3.11 to 2.3.16 won't be created by executing the script file. They should be created manually by using Windows registry editor ('RegEdit.exe').

Windows Environment

```
# Script execution

   - Note that the current directory is a directory that includes generation scripts and related files (.REG, PowerShell, Python…).

   $ ug-automation-windows.bat
```

```
# Execution messages

 * User-Generated Reference Registry Hive File Generator in Windows Environment

 * Developed and managed by
   - NIST CFTT   (Computer Forensic Tool Testing)         www.cftt.nist.gov
   - NIST CFReDS (Computer Forensic Reference Data Sets)  www.cfreds.nist.gov

  ------------------------------------------------------------------------------------------
  == Set global variables
  ------------------------------------------------------------------------------------------
  == Execute 'NR' scripts for creating each test case on the format v13
     (NR means normal registry hives)
     |
     |--- [nr]-01-1_possible-data-types.reg
     |     |--- Load the hive [nr]-01-1_v13.hive
     |     |--- Launch the .REG file [nr]-01-1_possible-data-types.reg
     |     |--- Unload the hive
     |     |--- Duplicate result files [nr]-01-1_v13.hive*
     |--- [nr]-02-1_simple-tree-structure.reg
     |     |--- Load the hive [nr]-02-1_v13.hive
     |     |--- Launch the .REG file [nr]-02-1_simple-tree-structure.reg
     |     |--- Unload the hive
     |     |--- Duplicate result files [nr]-02-1_v13.hive*
    …(skip)…
  ------------------------------------------------------------------------------------------
  == Execute 'NR' scripts for creating a single all-in-one hive on the format v13
     |
     |--- Launch all NR scripts
     |     |--- Load the hive [nr]-##-1_all-in-one_v13.hive
     |     |--- Launch the .REG file [nr]-01-1_possible-data-types.reg
     |     |--- Launch the .REG file [nr]-02-1_simple-tree-structure.reg
     |     |--- Launch the .REG file [nr]-03-1_tree-structure-with-the-maximum-levels.reg
     |     |--- Launch the .REG file [nr]-04-1_maximum-key-name-length.reg
     |     |--- Launch the .REG file [nr]-05-1_maximum-value-name-length.reg
     |     |--- Launch the .REG file [nr]-06-1_big-data.reg
     |     |--- Launch the .REG file [nr]-07-1_non-ascii-characters.reg
     |     |--- Unload the hive
     |     |--- Duplicate result files [nr]-##-1_all-in-one_v13.hive*
  ------------------------------------------------------------------------------------------
```

```
== Execute 'NRD' scripts for creating each test case on the format v13
   (NRD means normal registry hives with deleted registry data)
   |
   |--- [nrd]-01-1_delete-keys-with-values-but-without-subkeys.reg
   |    |--- Load the hive [nrd]-01-1_v13.hive
   |    |--- Launch the .REG file [nrd]-01-1_delete-keys-with-values-but-without-subkeys.reg
   |    |--- Unload the hive
   |    |--- Duplicate result files [nrd]-01-1_v13.hive*
   |--- [nrd]-02-1_delete-keys-with-values-and-subkeys.reg
   |    |--- Load the hive [nrd]-02-1_v13.hive
   |    |--- Launch the .REG file [nrd]-02-1_delete-keys-with-values-and-subkeys.reg
   |    |--- Unload the hive
   |    |--- Duplicate result files [nrd]-02-1_v13.hive*
   …(skip)…
--------------------------------------------------------------------------------
== Execute 'CR' scripts for creating each test case on the format v13
   (CR means corrupted registry hives)
   |
   |--- [cr]-01_a-hive-bin-with-root-key.py
   |    |--- Launch the Python script [cr]-01_a-hive-bin-with-root-key.py
   |    |--- Duplicate result files [cr]-01_v13.hive*
   |--- [cr]-02_a-hive-bin-randomly-selected.py
   |    |--- Launch the Python script [cr]-02_a-hive-bin-randomly-selected.py
   |    |--- Duplicate result files [cr]-02_v13.hive*
   …(skip)…
--------------------------------------------------------------------------------
== Execute 'MR' scripts for creating each test case on the format v13
   (MR means manipulated registry hives)
   |
   |--- [mr]-01.(1)_hide-a-root-key.py
   |    |--- Launch the Python script [mr]-01.(1)_hide-a-root-key.py
   |    |--- Duplicate result files [mr]-01.(1)_v13.hive*
   |--- [mr]-01.(2)_hide-a-root-key.py
   |    |--- Launch the Python script [mr]-01.(2)_hide-a-root-key.py
   |    |--- Duplicate result files [mr]-01.(2)_v13.hive*
   |--- [mr]-02.(1)_hide-key-names.py
   |    |--- Launch the Python script [mr]-02.(1)_hide-key-names.py
   |    |--- Duplicate result files [mr]-02.(1)_v13.hive*
   …(skip)…
--------------------------------------------------------------------------------
== Execute 'NR' scripts for creating each test case on the format v15
   (NR means normal registry hives)
   |
   |--- [nr]-01-1_possible-data-types.reg
   |    |--- Load the hive [nr]-01-1_v15.hive
   |    |--- Unload the hive
   |    |--- Patch the current hive format from v1.3 to v1.5
   |    |--- Load the hive [nr]-01-1_v15.hive
   |    |--- Launch the .REG file [nr]-01-1_possible-data-types.reg
   |    |--- Unload the hive
   |    |--- Duplicate result files [nr]-01-1_v15.hive*
   |--- [nr]-02-1_simple-tree-structure.reg
   |    |--- Load the hive [nr]-02-1_v15.hive
   |    |--- Unload the hive
   |    |--- Patch the current hive format from v1.3 to v1.5
   |    |--- Load the hive [nr]-02-1_v15.hive
   |    |--- Launch the .REG file [nr]-02-1_simple-tree-structure.reg
   |    |--- Unload the hive
   |    |--- Duplicate result files [nr]-02-1_v15.hive*
   …(skip)…
--------------------------------------------------------------------------------
== Execute 'NR' scripts for creating a single all-in-one hive on the format v15
   |
   |--- Launch all NR scripts
   |    |--- Load the hive [nr]-##-1_all-in-one_v15.hive
   |    |--- Unload the hive
   |    |--- Patch the current hive format from v1.3 to v1.5
   |    |--- Load the hive [nr]-##-1_all-in-one_v15.hive
   |    |--- Launch the .REG file [nr]-01-1_possible-data-types.reg
   |    |--- Launch the .REG file [nr]-02-1_simple-tree-structure.reg
   |    |--- Launch the .REG file [nr]-03-1_tree-structure-with-the-maximum-levels.reg
   |    |--- Launch the .REG file [nr]-04-1_maximum-key-name-length.reg
   |    |--- Launch the .REG file [nr]-05-1_maximum-value-name-length.reg
   |    |--- Launch the .REG file [nr]-06-1_big-data.reg
   |    |--- Launch the .REG file [nr]-07-1_non-ascii-characters.reg
   |    |--- Unload the hive
   |    |--- Duplicate result files [nr]-##-1_all-in-one_v15.hive*
--------------------------------------------------------------------------------
== Execute 'NRD' scripts for creating each test case on the format v15
   (NRD means normal registry hives with deleted registry data)
   |
   |--- [nrd]-01-1_delete-keys-with-values-but-without-subkeys.reg
   |    |--- Load the hive [nrd]-01-1_v15.hive
   |    |--- Launch the .REG file [nrd]-01-1_delete-keys-with-values-but-without-subkeys.reg
   |    |--- Unload the hive
   |    |--- Duplicate result files [nrd]-01-1_v15.hive*
   |--- [nrd]-02-1_delete-keys-with-values-and-subkeys.reg
   |    |--- Load the hive [nrd]-02-1_v15.hive
   |    |--- Launch the .REG file [nrd]-02-1_delete-keys-with-values-and-subkeys.reg
   |    |--- Unload the hive
   |    |--- Duplicate result files [nrd]-02-1_v15.hive*
   …(skip)…
```

```
  ------------------------------------------------------------------------------------------
  == Execute 'CR' scripts for creating each test case on the format v15
     (CR means corrupted registry hives)
     |
     |--- [cr]-01_a-hive-bin-with-root-key.py
     |    |--- Launch the Python script [cr]-01_a-hive-bin-with-root-key.py
     |    |--- Duplicate result files [cr]-01_v15.hive*
     |--- [cr]-02_a-hive-bin-randomly-selected.py
     |    |--- Launch the Python script [cr]-02_a-hive-bin-randomly-selected.py
     |    |--- Duplicate result files [cr]-02_v15.hive*
     …(skip)…
  ------------------------------------------------------------------------------------------
  == Execute 'MR' scripts for creating each test case on the format v15
     (MR means manipulated registry hives)
     |
     |--- [mr]-01.(1)_hide-a-root-key.py
     |    |--- Launch the Python script [mr]-01.(1)_hide-a-root-key.py
     |    |--- Duplicate result files [mr]-01.(1)_v15.hive*
     |--- [mr]-01.(2)_hide-a-root-key.py
     |    |--- Launch the Python script [mr]-01.(2)_hide-a-root-key.py
     |    |--- Duplicate result files [mr]-01.(2)_v15.hive*
     |--- [mr]-02.(1)_hide-key-names.py
     |    |--- Launch the Python script [mr]-02.(1)_hide-key-names.py
     |    |--- Duplicate result files [mr]-02.(1)_v15.hive*
     …(skip)…
  ------------------------------------------------------------------------------------------
  Time taken 00:00:41
```

Result data generated by the script

```
  - Assume that %SCRIPT_ROOT% is a directory the script file resides.
  - '+---' prefix means DIRECTORY, otherwise FILE.

  +---%SCRIPT_ROOT%\[YYYY-MM-DD HH.MM.SS] User-Generated Registry Hives using WinAPI
      |
      +---[nr]-01-1_possible-data-types_v13
      |       [nr]-01-1_v13.hive
      |       [nr]-01-1_v13.hive.LOG1
      |       [nr]-01-1_v13.hive.LOG2
      |       [nr]-01-1_v13.hive{GUID}.TM.blf
      |       [nr]-01-1_v13.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
      |       [nr]-01-1_v13.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
      |
      +---[nr]-01-1_possible-data-types_v15
      |       [nr]-01-1_v15.hive
      |       [nr]-01-1_v15.hive.LOG1
      |       [nr]-01-1_v15.hive.LOG2
      |       [nr]-01-1_v15.hive{GUID}.TM.blf
      |       [nr]-01-1_v15.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
      |       [nr]-01-1_v15.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
      |
      +---[nr]-02-1_simple-tree-structure_v13
      |       [nr]-02-1_v13.hive
      |       [nr]-02-1_v13.hive.LOG1
      |       [nr]-02-1_v13.hive.LOG2
      |       [nr]-02-1_v13.hive{GUID}.TM.blf
      |       [nr]-02-1_v13.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
      |       [nr]-02-1_v13.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
      |
      +---[nr]-02-1_simple-tree-structure_v15
      |       [nr]-02-1_v15.hive
      |       [nr]-02-1_v15.hive.LOG1
      |       [nr]-02-1_v15.hive.LOG2
      |       [nr]-02-1_v15.hive{GUID}.TM.blf
      |       [nr]-02-1_v15.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
      |       [nr]-02-1_v15.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
      |
      …(skip)…
      |
      +---[nr]-##-1_all-in-one_v13
      |       [nr]-##-1_all-in-one_v13.hive
      |       [nr]-##-1_all-in-one_v13.hive.LOG1
      |       [nr]-##-1_all-in-one_v13.hive.LOG2
      |       [nr]-##-1_all-in-one_v13.hive{GUID}.TM.blf
      |       [nr]-##-1_all-in-one_v13.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
      |       [nr]-##-1_all-in-one_v13.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
      |
      +---[nr]-##-1_all-in-one_v15
      |       [nr]-##-1_all-in-one_v15.hive
      |       [nr]-##-1_all-in-one_v15.hive.LOG1
      |       [nr]-##-1_all-in-one_v15.hive.LOG2
      |       [nr]-##-1_all-in-one_v15.hive{GUID}.TM.blf
      |       [nr]-##-1_all-in-one_v15.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
      |       [nr]-##-1_all-in-one_v15.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
      |
      +---[nrd]-01-1_delete-keys-with-values-but-without-subkeys_v13
      |       [nrd]-01-1_v13.hive
      |       [nrd]-01-1_v13.hive.LOG1
      |       [nrd]-01-1_v13.hive.LOG2
      |       [nrd]-01-1_v13.hive{GUID}.TM.blf
```

```
|       [nrd]-01-1_v13.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
|       [nrd]-01-1_v13.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
|
+---[nrd]-01-1_delete-keys-with-values-but-without-subkeys_v15
|       [nrd]-01-1_v15.hive
|       [nrd]-01-1_v15.hive.LOG1
|       [nrd]-01-1_v15.hive.LOG2
|       [nrd]-01-1_v15.hive{GUID}.TM.blf
|       [nrd]-01-1_v15.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
|       [nrd]-01-1_v15.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
|
+---[nrd]-02-1_delete-keys-with-values-and-subkeys_v13
|       [nrd]-02-1_v13.hive
|       [nrd]-02-1_v13.hive.LOG1
|       [nrd]-02-1_v13.hive.LOG2
|       [nrd]-02-1_v13.hive{GUID}.TM.blf
|       [nrd]-02-1_v13.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
|       [nrd]-02-1_v13.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
|
+---[nrd]-02-1_delete-keys-with-values-and-subkeys_v15
|       [nrd]-02-1_v15.hive
|       [nrd]-02-1_v15.hive.LOG1
|       [nrd]-02-1_v15.hive.LOG2
|       [nrd]-02-1_v15.hive{GUID}.TM.blf
|       [nrd]-02-1_v15.hive{GUID}.TMContainer00000000000000000001.regtrans-ms
|       [nrd]-02-1_v15.hive{GUID}.TMContainer00000000000000000002.regtrans-ms
|
…(skip)…
|
+---[cr]-01_a-hive-bin-with-root-key_v13
|       [cr]-01_v13.hive
|       [cr]-01_v13.hive.txt
|
+---[cr]-01_a-hive-bin-with-root-key_v15
|       [cr]-01_v15.hive
|       [cr]-01_v15.hive.txt
|
+---[cr]-02_a-hive-bin-randomly-selected_v13
|       [cr]-02_v13.hive
|       [cr]-02_v13.hive.txt
|
+---[cr]-02_a-hive-bin-randomly-selected_v15
|       [cr]-02_v15.hive
|       [cr]-02_v15.hive.txt
|
+---[cr]-03_last-half_v13
|       [cr]-03_v13.hive
|       [cr]-03_v13.hive.txt
|
+---[cr]-03_last-half_v15
|       [cr]-03_v15.hive
|       [cr]-03_v15.hive.txt
|
…(skip)…
|
+---[mr]-01.(1)_hide-a-root-key_v13
|       [mr]-01.(1)_v13.hive
|       [mr]-01.(1)_v13.hive.txt
|
+---[mr]-01.(1)_hide-a-root-key_v15
|       [mr]-01.(1)_v15.hive
|       [mr]-01.(1)_v15.hive.txt
|
+---[mr]-01.(2)_hide-a-root-key_v13
|       [mr]-01.(2)_v13.hive
|       [mr]-01.(2)_v13.hive.txt
|
+---[mr]-01.(2)_hide-a-root-key_v15
|       [mr]-01.(2)_v15.hive
|       [mr]-01.(2)_v15.hive.txt
|
+---[mr]-02.(1)_hide-key-names_v13
|       [mr]-02.(1)_v13.hive
|       [mr]-02.(1)_v13.hive.txt
|
+---[mr]-02.(1)_hide-key-names_v15
|       [mr]-02.(1)_v15.hive
|       [mr]-02.(1)_v15.hive.txt
|
…(skip)…
```

In Linux environment, a bash script file (‘ug-automation-linux.sh’) will create user-generated reference registry hive files related to several ‘NR’ and ‘NRD’ categories.

Linux Environment

Script execution

- Note that the current directory is a directory that includes generation scripts and related files (Python scripts, clean hive…).

$ *ug-automation-linux.sh*

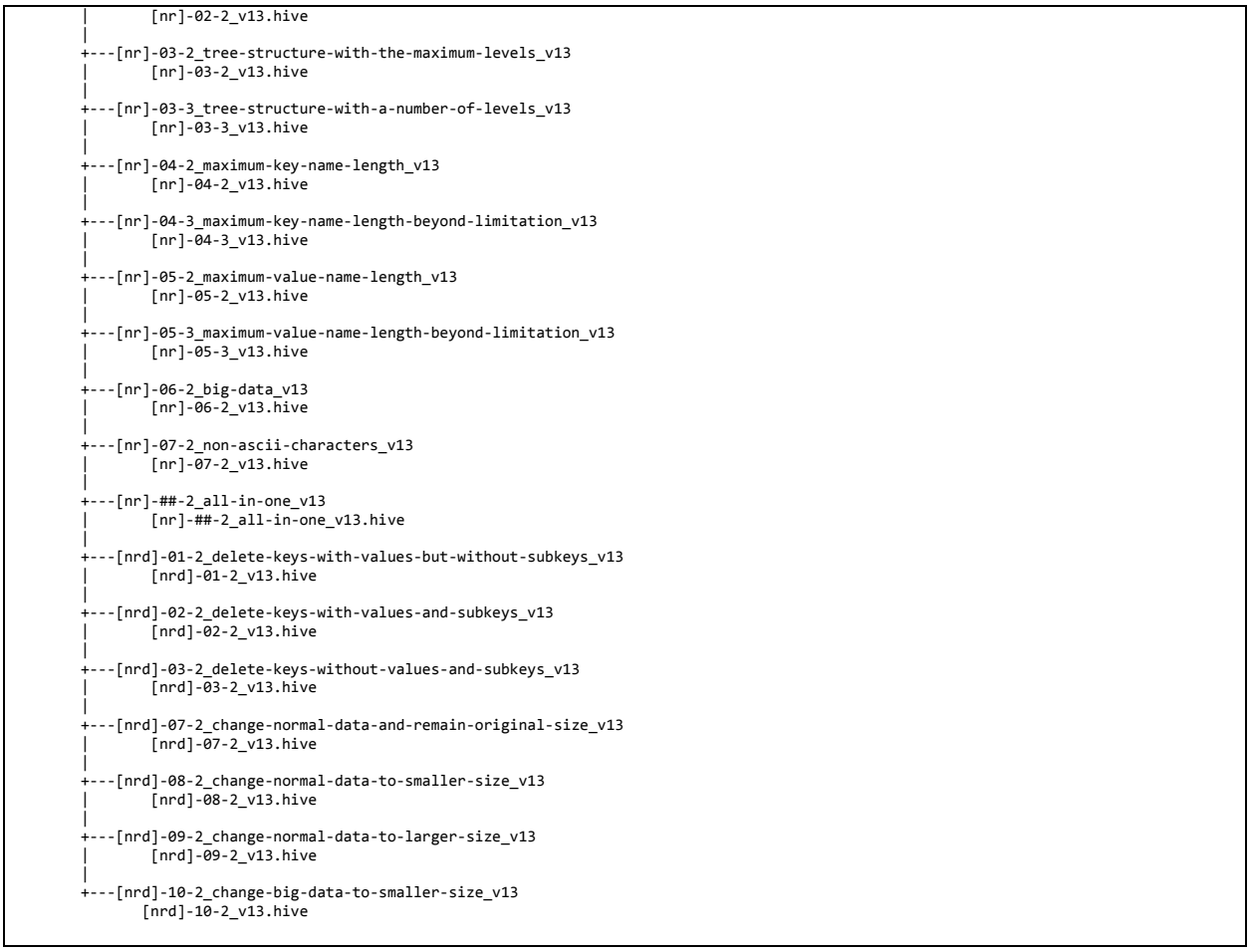
Execution messages

```
* User-Generated Reference Registry Hive File Generator in Linux Environment

* Developed and managed by
  - NIST CFTT   (Computer Forensic Tool Testing)         www.cftt.nist.gov
  - NIST CFReDS (Computer Forensic Reference Data Sets)  www.cfreds.nist.gov

-------------------------------------------------------------------------------------------
== Set global variables
-------------------------------------------------------------------------------------------
== Execute 'NR' scripts for creating each test case
   (NR means normal registry hives)
   |
   |--- [nr]-01-2_possible-data-types.py
   |    |--- Launch the python script [nr]-01-2_possible-data-types.py
   |    |--- Duplicate the result file [nr]-01-2_v13.hive
   |--- [nr]-02-2_simple-tree-structure.py
   |    |--- Launch the python script [nr]-02-2_simple-tree-structure.py
   |    |--- Duplicate the result file [nr]-02-2_v13.hive
   |--- [nr]-03-2_tree-structure-with-the-maximum-levels.py
   |    |--- Launch the python script [nr]-03-2_tree-structure-with-the-maximum-levels.py
   |    |--- Duplicate the result file [nr]-03-2_v13.hive
   |--- [nr]-03-3_tree-structure-with-a-number-of-levels.py
   |    |--- Launch the python script [nr]-03-3_tree-structure-with-a-number-of-levels.py
   |    |--- Duplicate the result file [nr]-03-3_v13.hive
   |--- [nr]-04-2_maximum-key-name-length.py
   |    |--- Launch the python script [nr]-04-2_maximum-key-name-length.py
   |    |--- Duplicate the result file [nr]-04-2_v13.hive
  …(skip)…
-------------------------------------------------------------------------------------------
== Execute 'NR' scripts for creating a single all-in-one hive
   |
   |--- Launch all NR scripts
   |    |--- Launch the python script [nr]-01-2_possible-data-types.py
   |    |--- Launch the python script [nr]-02-2_simple-tree-structure.py
   |    |--- Launch the python script [nr]-03-2_tree-structure-with-the-maximum-levels.py
   |    |--- Launch the python script [nr]-04-2_maximum-key-name-length.py
   |    |--- Launch the python script [nr]-05-2_maximum-value-name-length.py
   |    |--- Launch the python script [nr]-06-2_big-data.py
   |    |--- Launch the python script [nr]-07-2_non-ascii-characters.py
   |    |--- Duplicate the result file [nr]-##-2_all-in-one_v13.hive
-------------------------------------------------------------------------------------------
== Execute 'NRD' scripts for creating each test case
   (NRD means normal registry hives with deleted registry data)
   |
   |--- [nrd]-01-2_delete-keys-with-values-but-without-subkeys.py
   |    |--- Launch the python script [nrd]-01-2_delete-keys-with-values-but-without-subkeys.py
   |    |--- Duplicate the result file [nrd]-01-2_v13.hive
   |--- [nrd]-02-2_delete-keys-with-values-and-subkeys.py
   |    |--- Launch the python script [nrd]-02-2_delete-keys-with-values-and-subkeys.py
   |    |--- Duplicate the result file [nrd]-02-2_v13.hive
   |--- [nrd]-03-2_delete-keys-without-values-and-subkeys.py
   |    |--- Launch the python script [nrd]-03-2_delete-keys-without-values-and-subkeys.py
   |    |--- Duplicate the result file [nrd]-03-2_v13.hive
   |--- [nrd]-07-2_change-normal-data-and-remain-original-size.py
   |    |--- Launch the python script [nrd]-07-2_change-normal-data-and-remain-original-size.py
   |    |--- Duplicate the result file [nrd]-07-2_v13.hive
  …(skip)…
-------------------------------------------------------------------------------------------
Time taken 00:00:15
```

Result data generated by the script

- Assume that %SCRIPT_ROOT% is a directory the script file resides.
- ‘+---’ prefix means DIRECTORY, otherwise FILE.

```
+---%SCRIPT_ROOT%\[YYYY-MM-DD HH.MM.SS] User-Generated Registry Hives using Hivex
    |
    +---[nr]-01-2_possible-data-types_v13
    |       [nr]-01-2_v13.hive
    |
    +---[nr]-02-2_simple-tree-structure_v13
```


[nr]-02-2_v13.hive
+---[nr]-03-2_tree-structure-with-the-maximum-levels_v13
[nr]-03-2_v13.hive
+---[nr]-03-3_tree-structure-with-a-number-of-levels_v13
[nr]-03-3_v13.hive
+---[nr]-04-2_maximum-key-name-length_v13
[nr]-04-2_v13.hive
+---[nr]-04-3_maximum-key-name-length-beyond-limitation_v13
[nr]-04-3_v13.hive
+---[nr]-05-2_maximum-value-name-length_v13
[nr]-05-2_v13.hive
+---[nr]-05-3_maximum-value-name-length-beyond-limitation_v13
[nr]-05-3_v13.hive
+---[nr]-06-2_big-data_v13
[nr]-06-2_v13.hive
+---[nr]-07-2_non-ascii-characters_v13
[nr]-07-2_v13.hive
+---[nr]-##-2_all-in-one_v13
[nr]-##-2_all-in-one_v13.hive
+---[nrd]-01-2_delete-keys-with-values-but-without-subkeys_v13
[nrd]-01-2_v13.hive
+---[nrd]-02-2_delete-keys-with-values-and-subkeys_v13
[nrd]-02-2_v13.hive
+---[nrd]-03-2_delete-keys-without-values-and-subkeys_v13
[nrd]-03-2_v13.hive
+---[nrd]-07-2_change-normal-data-and-remain-original-size_v13
[nrd]-07-2_v13.hive
+---[nrd]-08-2_change-normal-data-to-smaller-size_v13
[nrd]-08-2_v13.hive
+---[nrd]-09-2_change-normal-data-to-larger-size_v13
[nrd]-09-2_v13.hive
+---[nrd]-10-2_change-big-data-to-smaller-size_v13
[nrd]-10-2_v13.hive

2.7. GENERATED REFERENCE DATA INFORMATION

For listing the generation results in this work, **Table 5** defines the table structure for user-generated registry data. This structure will be used for explaining each registry hive file on **Table 6**.

Table 5. Definition of the table structure for user-generated registry data

Data code	Detailed Information	
Prefix of a hive file name	Class name	Class name (refer to Sections 2.2 to 2.5)
	Generation method	One of below generation methods: - [Windows] .REG file - [Windows] RegEdit (manual operation) - [Windows] PowerShell script - [Windows] Python script - [Linux] Python script using hivex library
	File paths	File paths based on the '`ug-reference-hives`' directory as the current root
	Comments	Special comments about this registry data (N/A if there are no comments)

Note that operational behaviors described in comments of **Table 6** are results using a workstation having Windows 7 Enterprise SP1 (64-bits, English) and a specific version of hivex library in a Linux system.

Table 6. Summary of user-generated Windows registry data

Data code	Detailed Information
[nr]-01-1	Possible data types
	[Windows] .REG file

	`\[nr]-01-1_possible-data-types_v13\[nr]-01-1_v13.hive`
	`\[nr]-01-1_possible-data-types_v15\[nr]-01-1_v15.hive`
	(Transaction log files are also created such as LOG# and Transactional Registry (TxR))

	N/A
[nr]-01-2	Possible data types
	[Linux] Python script using hivex library
	`\[nr]-01-2_possible-data-types_v13\[nr]-01-2_v13.hive`
	N/A
[nr]-02-1	Simple tree structure
	[Windows] .REG file

	`\[nr]-02-1_simple-tree-structure_v13\[nr]-02-1_v13.hive`
	`\[nr]-02-1_simple-tree-structure_v15\[nr]-02-1_v15.hive`
	(Transaction log files are also created such as LOG# and Transactional Registry (TxR))

	N/A
[nr]-02-2	Simple tree structure
	[Linux] Python script using hivex library
	`\[nr]-02-2_simple-tree-structure_v13\[nr]-02-2_v13.hive`
	N/A
[nr]-03-1	Tree structure with the maximum levels
	[Windows] .REG file

	`\[nr]-03-1_tree-structure-with-the-maximum-levels_v13\[nr]-03-1_v13.hive`
	`\[nr]-03-1_tree-structure-with-the-maximum-levels_v15\[nr]-03-1_v15.hive`
	(Transaction log files are also created such as LOG# and Transactional Registry (TxR))

	N/A
[nr]-03-2	Tree structure with the maximum levels
	[Linux] Python script using hivex library
	`\[nr]-03-2_tree-structure-with-the-maximum-levels_v13\[nr]-03-2_v13.hive`
	N/A
[nr]-03-3	Tree structure with a number of levels
	[Linux] Python script using hivex library
	`\[nr]-03-3_tree-structure-with-a-number-of-levels_v13\[nr]-03-3_v13.hive`
	N/A
[nr]-04-1	Maximum key name length
	[Windows] .REG file
	`\[nr]-04-1_maximum-key-name-length_v13\[nr]-04-1_v13.hive` `\[nr]-04-1_maximum-key-name-length_v15\[nr]-04-1_v15.hive`

	(Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nr]-04-2	Maximum key name length
	[Linux] Python script using hivex library
	\[nr]-04-2_maximum-key-name-length_v13\[nr]-04-2_v13.hive
	N/A
[nr]-04-3	Maximum key name length beyond the limitation
	[Linux] Python script using hivex library
	\[nr]-04-3_maximum-key-name-length-beyond-limitation_v13\[nr]-04-3_v13.hive
	N/A
[nr]-05-1	Maximum value name length
	[Windows] .REG file
	\[nr]-05-1_maximum-value-name-length_v13\[nr]-05-1_v13.hive \[nr]-05-1_maximum-value-name-length_v15\[nr]-05-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nr]-05-2	Maximum value name length
	[Linux] Python script using hivex library
	\[nr]-05-2_maximum-value-name-length_v13\[nr]-05-2_v13.hive
	N/A
[nr]-05-3	Maximum value name length beyond the limitation
	[Linux] Python script using hivex library
	\[nr]-05-3_maximum-value-name-length-beyond-limitation_v13\[nr]-05-3_v13.hive
	N/A
[nr]-06-1	Big data
	[Windows] .REG file
	\[nr]-06-1_big-data_v13\[nr]-06-1_v13.hive \[nr]-06-1_big-data_v15\[nr]-06-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nr]-06-2	Big data
	[Linux] Python script using hivex library
	\[nr]-06-2_big-data_v13\[nr]-06-2_v13.hive
	N/A
[nr]-07-1	Non-ASCII characters
	[Windows] .REG file
	\[nr]-07-1_non-ascii-characters_v13\[nr]-07-1_v13.hive \[nr]-07-1_non-ascii-characters_v15\[nr]-07-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nr]-07-2	Non-ASCII characters
	[Linux] Python script using hivex library
	\[nr]-07-2_non-ascii-characters_v13\[nr]-07-2_v13.hive
	N/A
[nr]-08-1	Naming Convention
	[Windows] Python script
	\[nr]-08_key-value-naming-convention_v13\[nr]-08_v13.hive \[nr]-08_key-value-naming-convention_v15\[nr]-08_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nr]-##-1	All-in-one hive with all types of NR types
	[Windows] .REG file
	\[nr]-##-1_all-in-one_v13\[nr]-##-1_all-in-one_v13.hive \[nr]-##-1_all-in-one_v15\[nr]-##-1_all-in-one_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	See [nr]-*-1
[nr]-##-2	All-in-one hive with all types of NR types
	[Linux] Python script using hivex library
	\[nr]-##-2_all-in-one_v13\[nr]-##-2_all-in-one_v13.hive
	See [nr]-*-2
[nrd]-01-1	Delete keys with values, but without subkeys
	[Windows] .REG file
	\[nrd]-01-1_delete-keys-with-values-but-without-subkeys_v13\[nrd]-01-1_v13.hive \[nrd]-01-1_delete-keys-with-values-but-without-subkeys_v15\[nrd]-01-1_v15.hive

	(Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nrd]-01-2	Delete keys with values, but without subkeys
	[Linux] Python script using hivex library
	\[nrd]-01-2_delete-keys-with-values-but-without-subkeys_v13\[nrd]-01-2_v13.hive
	N/A
[nrd]-02-1	Delete a key with values and subkeys
	[Windows] .REG file
	\[nrd]-02-1_delete-keys-with-values-and-subkeys_v13\[nrd]-02-1_v13.hive \[nrd]-02-1_delete-keys-with-values-and-subkeys_v15\[nrd]-02-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nrd]-02-2	Delete a key with values and subkeys
	[Linux] Python script using hivex library
	\[nrd]-02-2_delete-keys-with-values-and-subkeys_v13\[nrd]-02-2_v13.hive
	N/A
[nrd]-03-1	Delete a key without values and subkeys
	[Windows] .REG file
	\[nrd]-03-1_delete-keys-without-values-and-subkeys_v13\[nrd]-03-1_v13.hive \[nrd]-03-1_delete-keys-without-values-and-subkeys_v15\[nrd]-03-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nrd]-03-2	Delete a key without values and subkeys
	[Linux] Python script using hivex library
	\[nrd]-03-2_delete-keys-without-values-and-subkeys_v13\[nrd]-03-2_v13.hive
	N/A
[nrd]-04-1	Delete a value with normal data
	[Windows] .REG file
	\[nrd]-04-1_delete-a-value-with-normal-data_v13\[nrd]-04-1_v13.hive \[nrd]-04-1_delete-a-value-with-normal-data_v15\[nrd]-04-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nrd]-05-1	Delete a value with big data
	[Windows] .REG file
	\[nrd]-05-1_delete-a-value-with-big-data_v13\[nrd]-05-1_v13.hive \[nrd]-05-1_delete-a-value-with-big-data_v15\[nrd]-05-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nrd]-06-1	Delete multiple values in a key
	[Windows] .REG file
	\[nrd]-06-1_delete-multiple-values-in-a-key_v13\[nrd]-06-1_v13.hive \[nrd]-06-1_delete-multiple-values-in-a-key_v15\[nrd]-06-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	N/A
[nrd]-07-1	Change normal data and remain original size
	[Windows] .REG file
	\[nrd]-07-1_change-normal-data-and-remain-original-size_v13\[nrd]-07-1_v13.hive \[nrd]-07-1_change-normal-data-and-remain-original-size_v15\[nrd]-07-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	- Original data is overwritten by new data in the existing data cell.
[nrd]-07-2	Change normal data and remain original size
	[Linux] Python script using hivex library
	\[nrd]-07-2_change-normal-data-and-remain-original-size_v13\[nrd]-07-2_v13.hive
	- All value and data cells of a key which has the target value are copied, and then the target data is updated. - New cells are allocated for storing data through searching areas from the beginning of the file. - That is, all value and data cells related to the target are newly added to the file. - After the original value and data cells are deleted, they are marked as unallocated areas. - So, there is a possibility of remaining residual data at the position of the original data cell if the area is not overwritten by another allocation.
[nrd]-08-1	Change normal data to smaller size
	[Windows] .REG file
	\[nrd]-08-1_change-normal-data-to-smaller-size_v13\[nrd]-08-1_v13.hive \[nrd]-08-1_change-normal-data-to-smaller-size_v15\[nrd]-08-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))

	- Original data is overwritten by new data in the existing data cell. - There will be residual data at the tail of the existing data cell.
[nrd]-08-2	Change normal data to smaller size
	[Linux] Python script using hivex library
	\[nrd]-08-2_change-normal-data-to-smaller-size_v13\[nrd]-08-2_v13.hive
	Same as the comments of [nrd]-07-2
[nrd]-09-1	Change normal data to larger size
	[Windows] .REG file
	\[nrd]-09-1_change-normal-data-to-larger-size_v13\[nrd]-09-1_v13.hive \[nrd]-09-1_change-normal-data-to-larger-size_v15\[nrd]-09-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	- A new data cell is allocated for storing data through searching areas from the beginning of the file. - After the original data cell is erased, it is marked as an unallocated area. - There is a possibility of remaining residual data at the position of the original data cell if the area is not overwritten by another allocation.
[nrd]-09-2	Change normal data to larger size
	[Linux] Python script using hivex library
	\[nrd]-09-2_change-normal-data-to-larger-size_v13\[nrd]-09-2_v13.hive
	Same as the comments of [nrd]-07-2
[nrd]-10-1	Change big data to smaller size
	[Windows] .REG file
	\[nrd]-10-1_change-big-data-to-smaller-size_v13\[nrd]-10-1_v13.hive \[nrd]-10-1_change-big-data-to-smaller-size_v15\[nrd]-10-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-08-1
[nrd]-10-2	Change big data to smaller size
	[Linux] Python script using hivex library
	\[nrd]-10-2_change-big-data-to-smaller-size_v13\[nrd]-10-2_v13.hive
	Same as the comments of [nrd]-07-2
[nrd]-11-1	Change key name and remain original size
	[Windows] PowerShell script
	\[nrd]-11-1_change-key-name-and-remain-original-size_v13\[nrd]-11-1_v13.hive \[nrd]-11-1_change-key-name-and-remain-original-size_v15\[nrd]-11-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	- A key (nk) cell with the new key name is allocated through searching areas from the beginning of the file. - Cells for subkeys and values (+ data) of the original key are also newly allocated in the same manner. (That is, all cells are added to the tail position if there are no unallocated areas. Note that new cells are not always located at the end of a hive file because their actual locations are determined according to various factors including unallocated areas, required bytes (length) and available spaces in each hive bin.) - Then, the original key and its belongings (subkeys, values and data) are erased and marked as unallocated areas.
[nrd]-11-2	Change key name and remain original size
	[Windows] RegEdit (manual operation)
	\[nrd]-11-2_change-key-name-and-remain-original-size_v13\[nrd]-11-2_v13.hive \[nrd]-11-2_change-key-name-and-remain-original-size_v15\[nrd]-11-2_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	- A key (nk) cell with the new key name is allocated through searching areas from the beginning of the file. (That is, all cells are added to the tail position if there are no unallocated areas. Note that new cells are not always located at the end of a hive file because their actual locations are determined according to various factors including unallocated areas, required bytes (length) and available spaces in each hive bin.) - The subkey-list offset of the original key is copied to the field of the new key. - The parent key offset of the subkeys are changed for pointing to the new key.
[nrd]-12-1	Change key name to smaller size
	[Windows] PowerShell script
	\[nrd]-12-1_change-key-name-to-smaller-size_v13\[nrd]-12-1_v13.hive \[nrd]-12-1_change-key-name-to-smaller-size_v15\[nrd]-12-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-11-1
[nrd]-12-2	Change key name to smaller size
	[Windows] RegEdit (manual operation)
	\[nrd]-12-2_change-key-name-to-smaller-size_v13\[nrd]-12-2_v13.hive \[nrd]-12-2_change-key-name-to-smaller-size_v15\[nrd]-12-2_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-11-2
[nrd]-13-1	Change key name to larger size

	[Windows] PowerShell script
	\[nrd]-13-1_change-key-name-to-larger-size_v13\[nrd]-13-1_v13.hive \[nrd]-13-1_change-key-name-to-larger-size_v15\[nrd]-13-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-11-1
[nrd]-13-2	Change key name to larger size
	[Windows] RegEdit (manual operation)
	\[nrd]-13-2_change-key-name-to-larger-size_v13\[nrd]-13-2_v13.hive \[nrd]-13-2_change-key-name-to-larger-size_v15\[nrd]-13-2_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-11-2
[nrd]-14-1	Change value name and remain original size
	[Windows] PowerShell script
	\[nrd]-14-1_change-value-name-and-remain-original-size_v13\[nrd]-14-1_v13.hive \[nrd]-14-1_change-value-name-and-remain-original-size_v15\[nrd]-14-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	- A value (vk) cell with the new value name is allocated through searching areas from the beginning of the file. - A data cell of the original value is also newly allocated in the same manner. (That is, all cells are added to the tail position if there are no unallocated areas. Note that new cells are not always located at the end of a hive file because their actual locations are determined according to various factors including unallocated areas, required bytes (length) and available spaces in each hive bin.) - Then, the original value and its data are erased and marked as unallocated areas.
[nrd]-14-2	Change value name and remain original size
	[Windows] RegEdit (manual operation)
	\[nrd]-14-2_change-value-name-and-remain-original-size_v13\[nrd]-14-2_v13.hive \[nrd]-14-2_change-value-name-and-remain-original-size_v15\[nrd]-14-2_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-14-1
[nrd]-15-1	Change value name to smaller size
	[Windows] PowerShell script
	\[nrd]-15-1_change-value-name-to-smaller-size_v13\[nrd]-15-1_v13.hive \[nrd]-15-1_change-value-name-to-smaller-size_v15\[nrd]-15-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-14-1
[nrd]-15-2	Change value name to smaller size
	[Windows] RegEdit (manual operation)
	\[nrd]-15-2_change-value-name-to-smaller-size_v13\[nrd]-15-2_v13.hive \[nrd]-15-2_change-value-name-to-smaller-size_v15\[nrd]-15-2_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-14-1
[nrd]-16-1	Change value name to larger size
	[Windows] PowerShell script
	\[nrd]-16-1_change-value-name-to-larger-size_v13\[nrd]-16-1_v13.hive \[nrd]-16-1_change-value-name-to-larger-size_v15\[nrd]-16-1_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-14-1
[nrd]-16-2	Change value name to larger size
	[Windows] RegEdit (manual operation)
	\[nrd]-16-2_change-value-name-to-larger-size_v13\[nrd]-16-2_v13.hive \[nrd]-16-2-change-value-name-to-larger-size_v15\[nrd]-16-2_v15.hive (Transaction log files are also created such as LOG# and Transactional Registry (TxR))
	Same as the comments of [nrd]-14-1
[cr]-01	A hive bin with Root key
	[Windows] Python script
	\[cr]-01_a-hive-bin-with-root-key_v13\[cr]-01_v13.hive \[cr]-01_a-hive-bin-with-root-key_v15\[cr]-01_v15.hive (A log file including the corruption information is also created for each class.)
	N/A
[cr]-02	A hive bin randomly selected
	[Windows] Python script
	\[cr]-02_a-hive-bin-randomly-selected_v13\[cr]-02_v13.hive \[cr]-02_a-hive-bin-randomly-selected_v15\[cr]-02_v15.hive (A log file including the corruption information is also created for each class.)
	N/A
[cr]-03	Last half

	[Windows] Python script
	\[cr]-03_last-half_v13\[cr]-03_v13.hive \[cr]-03_last-half_v15\[cr]-03_v15.hive (A log file including the corruption information is also created for each class.)
	N/A
[cr]-04	Fragments with hbin header randomly selected
	[Windows] Python script
	\[cr]-04_fragments-with-hive-bin-header-randomly-selected_v13\[cr]-04_v13.hive \[cr]-04_fragments-with-hive-bin-header-randomly-selected_v15\[cr]-04_v15.hive (A log file including the corruption information is also created for each class.)
	N/A
[cr]-05	Hive header
	[Windows] Python script
	\[cr]-05_hive-header_v13\[cr]-05_v13.hive \[cr]-05_hive-header_v15\[cr]-05_v15.hive (A log file including the corruption information is also created for each class.)
	N/A
[cr]-06	First half
	[Windows] Python script
	\[cr]-06_first-half_v13\[cr]-06_v13.hive \[cr]-06_first-half_v15\[cr]-06_v15.hive (A log file including the corruption information is also created for each class.)
	N/A
[cr]-07	First and last quarter
	[Windows] Python script
	\[cr]-07_first-and-last-quarter_v13\[cr]-07_v13.hive \[cr]-07_first-and-last-quarter_v15\[cr]-07_v15.hive (A log file including the corruption information is also created for each class.)
	N/A
[mr]-01	< Data hiding > Hide a root key
	[Windows] Python script
	\[mr]-01.(1)_hide-a-root-key_v13\[mr]-01.(1)_v13.hive \[mr]-01.(1)_hide-a-root-key_v15\[mr]-01.(1)_v15.hive \[mr]-01.(2)_hide-a-root-key_v13\[mr]-01.(2)_v13.hive \[mr]-01.(2)_hide-a-root-key_v15\[mr]-01.(2)_v15.hive (A log file including the modified information is also created for each class.)
	N/A
[mr]-02	< Data hiding > Hide key names
	[Windows] Python script
	\[mr]-02.(1)_hide-key-names_v13\[mr]-02.(1)_v13.hive \[mr]-02.(1)_hide-key-names_v15\[mr]-02.(1)_v15.hive \[mr]-02.(2)_hide-key-names_v13\[mr]-02.(2)_v13.hive \[mr]-02.(2)_hide-key-names_v15\[mr]-02.(2)_v15.hive (A log file including the modified information is also created for each class.)
	Target key: ROOT\0x01_TYPE1_DATA-TYPES\
[mr]-03	< Data hiding > Hide subkeys of a key
	[Windows] Python script
	\[mr]-03.(1)_hide-subkeys-of-a-key_v13\[mr]-03.(1)_v13.hive \[mr]-03.(1)_hide-subkeys-of-a-key_v15\[mr]-03.(1)_v15.hive \[mr]-03.(2)_hide-subkeys-of-a-key_v13\[mr]-03.(2)_v13.hive \[mr]-03.(2)_hide-subkeys-of-a-key_v15\[mr]-03.(2)_v15.hive \[mr]-03.(3)_hide-subkeys-of-a-key_v13\[mr]-03.(3)_v13.hive \[mr]-03.(3)_hide-subkeys-of-a-key_v15\[mr]-03.(3)_v15.hive \[mr]-03.(4)_hide-subkeys-of-a-key_v13\[mr]-03.(4)_v13.hive \[mr]-03.(4)_hide-subkeys-of-a-key_v15\[mr]-03.(4)_v15.hive (A log file including the modified information is also created for each class.)
	Target key: ROOT\0x07_TYPE1_NON-ASCII\
[mr]-04	< Data hiding > Hide values of a key
	[Windows] Python script
	\[mr]-04.(1)_hide-values-of-a-key_v13\[mr]-04.(1)_v13.hive \[mr]-04.(1)_hide-values-of-a-key_v15\[mr]-04.(1)_v15.hive \[mr]-04.(2)_hide-values-of-a-key_v13\[mr]-04.(2)_v13.hive \[mr]-04.(2)_hide-values-of-a-key_v15\[mr]-04.(2)_v15.hive \[mr]-04.(3)_hide-values-of-a-key_v13\[mr]-04.(3)_v13.hive \[mr]-04.(3)_hide-values-of-a-key_v15\[mr]-04.(3)_v15.hive (A log file including the modified information is also created for each class.)
	Target values: ROOT\0x01_TYPE1_DATA-TYPES*
[mr]-05	< Data hiding > Hide value names

	[Windows] Python script
	\[mr]-05.(1)_hide-value-names_v13\[mr]-05.(1)_v13.hive \[mr]-05.(1)_hide-value-names_v15\[mr]-05.(1)_v15.hive \[mr]-05.(2)_hide-value-names_v13\[mr]-05.(2)_v13.hive \[mr]-05.(2)_hide-value-names_v15\[mr]-05.(2)_v15.hive (A log file including the modified information is also created for each class.)
	Target value: ROOT\0x01_TYPE1_DATA-TYPES\VALUE 0x00 (NONE)
[mr]-06	< Data hiding > Hide data of a value
	[Windows] Python script
	\[mr]-06.(1)_hide-data-of-a-value_v13\[mr]-06.(1)_v13.hive \[mr]-06.(1)_hide-data-of-a-value_v15\[mr]-06.(1)_v15.hive \[mr]-06.(2)_hide-data-of-a-value_v13\[mr]-06.(2)_v13.hive \[mr]-06.(2)_hide-data-of-a-value_v15\[mr]-06.(2)_v15.hive \[mr]-06.(3)_hide-data-of-a-value_v13\[mr]-06.(3)_v13.hive \[mr]-06.(3)_hide-data-of-a-value_v15\[mr]-06.(3)_v15.hive \[mr]-06.(4)_hide-data-of-a-value_v13\[mr]-06.(4)_v13.hive \[mr]-06.(4)_hide-data-of-a-value_v15\[mr]-06.(4)_v15.hive (A log file including the modified information is also created for each class.)
	Target value: ROOT\0x01_TYPE1_DATA-TYPES\VALUE 0x03 (BINARY)
[mr]-07	< Data hiding > Hide big data of a value
	[Windows] Python script
	\[mr]-07_hide-big-data-of-a-value_v13\[mr]-07_v13.hive \[mr]-07_hide-big-data-of-a-value_v15\[mr]-07_v15.hive (A log file including the modified information is also created for each class.)
	Target value: ROOT\0x06_TYPE1_BIG-DATA\BINARY 16345
[mr]-08	< Infinite loop > Key loop
	[Windows] Python script
	\[mr]-08_key-loop_v13\[mr]-08_v13.hive \[mr]-08_key-loop_v15\[mr]-08_v15.hive (A log file including the modified information is also created for each class.)
	Infinite loop: 'ROOT\0x02_TYPE1_TREE\Node_2\' → 'ROOT\'
[mr]-09	< Invalid data size > Integer data too large
	[Windows] Python script
	\[mr]-09_integer-data-too-large_v13\[mr]-09_v13.hive \[mr]-09_integer-data-too-large_v15\[mr]-09_v15.hive (A log file including the modified information is also created for each class.)
	Target value: ROOT\0x01_TYPE1_DATA-TYPES\VALUE 0x04 (DWORD-LE)
[mr]-10	< Invalid data size > Binary data too large
	[Windows] Python script
	\[mr]-10_binary-data-too-large_v13\[mr]-10_v13.hive \[mr]-10_binary-data-too-large_v15\[mr]-10_v15.hive (A log file including the modified information is also created for each class.)
	Target value: ROOT\0x01_TYPE1_DATA-TYPES\VALUE 0x03 (BINARY)
[mr]-11	< Invalid data size > String data too large
	[Windows] Python script
	\[mr]-11_string-data-too-large_v13\[mr]-11_v13.hive \[mr]-11_string-data-too-large_v15\[mr]-11_v15.hive (A log file including the modified information is also created for each class.)
	Target value: ROOT\0x01_TYPE1_DATA-TYPES\VALUE 0x01 (SZ)
[mr]-12	< Version mismatch > Big data management
	[Windows] Python script
	\[mr]-12_big-data-management_v13\[mr]-12_v13.hive \[mr]-12_big-data-management_v15\[mr]-12_v15.hive (A log file including the modified information is also created for each class.)
	N/A
[mr]-13	< Ambiguous encoding > Key name flag
	[Windows] Python script
	\[mr]-13_key-name-flag_v13\[mr]-13_v13.hive \[mr]-13_key-name-flag_v15\[mr]-13_v15.hive (A log file including the modified information is also created for each class.)
	Target key: ROOT\0x07_TYPE1_NON-ASCII\Здравствуйте\
[mr]-14	< Ambiguous encoding > Value name flag
	[Windows] Python script
	\[mr]-14_value-name-flag_v13\[mr]-14_v13.hive \[mr]-14_value-name-flag_v15\[mr]-14_v15.hive (A log file including the modified information is also created for each class.)
	Target value: ROOT\0x07_TYPE1_NON-ASCII\Здравствуйте\Здравствуйте

[mr]-15	< Ambiguous encoding > Different encodings
	[Windows] Python script
	\[mr]-15_different-encodings_v13\[mr]-15_v13.hive \[mr]-15_different-encodings_v15\[mr]-15_v15.hive (A log file including the modified information is also created for each class.)
	N/A

3. SYSTEM-GENERATED REFERENCE WINDOWS REGISTRY DATA

This sub-section describes details about system-generated reference Windows registry data. As mentioned in Section 1, a term 'system-generated' means that this kind of data is to be extracted from actual Windows systems populated by user activities. As shown in **Table 7**, system-generated Windows registry hive files can be utilized for various tool testing cases including the interpretation of well-known registry data.

Table 7. System-generated registry hives and tool testing points

Research and tool testing considerations	User-generated registry hives	System-generated registry hives	Note
Supporting various input types	V	V	Hive set, backup hives
Parsing normal registry hives	V	V	
Parsing corrupted registry hives	V	-	
Recovering deleted registry data	V	V	
Interpreting well-known registry data	-	V	Interpreting artifacts
Countering anti-forensics	V	-	Manipulated structures

3.1. GENERATION STRATEGY

The overall generation strategy for developing system-generated reference Windows registry data considers the following requirements. It is necessary to note that these requirements can be applied to generating any kind of reference datasets related to Microsoft Windows systems and other operating systems as well.

Requirements for System-Generated Reference Windows Registry Data
- The reference Windows systems developed here should consist of various versions of Windows. - The reference Windows systems should not contain any license information for commercial software. - A common scenario should be developed upon consideration of a variety of forensically meaningful user actions to populate the reference Windows systems. - A common scenario should be applied to all reference systems in order to identify differences between various Windows versions. Note that some actions may be valid only in specific Windows versions. - The reference data should include various well-known registry hive files from the reference Windows systems. It should also include those files from a system partition and its backups as well (e.g., Volume Shadow Copies in the case of Windows Vista and higher).

The practical approaches to perform actual generation processes may vary according to supported environments, available resources and purposes of generating datasets. In this project, we propose a systematic procedure for the efficient conduct of the generation processes based on the above requirements.

3.1.1. Overall procedure

The **Fig. 7** depicts the overall procedure for developing system-generated reference data in this project. The procedure is divided into five steps like shown in the figure. It is necessary to note that this procedure can be applied generally for developing any types of system-generated reference data including Windows registry in this project.

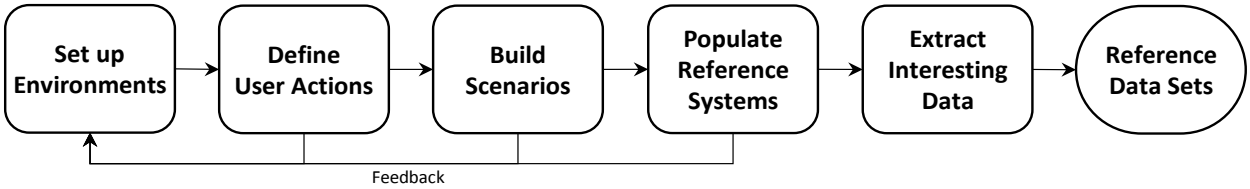


Figure 7. Procedure for developing system-generated reference Windows registry data

The first step is to set up execution environments based on the purpose of generating datasets. It includes but not limited to operating systems, external devices, software/hardware tools and other related resources. For example, this project requires at least one physical workstation for executing related programs, a type 2 hypervisor (like VirtualBox) for managing virtual machines, three external devices with the USB interface, and six different versions of Windows operating systems. As shown in the **Fig. 7**, this step is deeply related to the following steps. So, the execution environment needs to be built clearly based on the purpose, and of course afterward it can be continuously updated by feedbacks.

The next step is to define user actions related to reference data that are currently being made based on the purpose (obtaining reference Windows registry data in this project) and execution environments established in the first step. Because this step is closely related with the third step for building common scenarios, user actions defined here should be performed without any issues on at least one or more target systems. In addition to this condition, it is also necessary to consider a variety of meaningful actions depicting actual users' behaviors with various versions of operating systems and applications.

In the third step, common scenarios (a scenario consists of a sequence of user actions defined in the second step) depicting user behaviors are developed for populating reference systems. This step should produce at least one or more scenarios for performing the next population step. For the efficient conduct of the next step, the scenario should be developed as much detail as possible. Additionally, user actions in a scenario can be grouped into multiple action stages if necessary for classifying related user behaviors. These action stages will be helpful to improve the applicability as a reference data, because it provides interesting intervals between user actions.

The fourth step is to populate reference operating systems and applications according to the scenario developed in the previous step. In this step, it is required to select appropriate population strategies based on the execution environments established in the first step. To populate the target systems (Windows OSes in this project), we utilize *pyvmpop*[16] that supports the virtual machine automation as well as forensic data extraction based on the concept of VMPOP (virtual machine population) framework. This framework offers repeatability and reproducibility by representing detailed user actions as programming codes, and also make the population process traceable through providing dedicated logging features. These advantages will enhance the completeness and usefulness of reference data. Note that all the previous steps can be updated by feedbacks during the actual population process of this step.

[16] *pyvmpop*: A Python implementation of VMPOP (virtual machine population) framework (https://github.com/jungheum/pyvmpop)

The final step is to extract forensically interesting data from reference operating systems populated by the previous step. This step is not mandatory if the purpose is to obtain reference data as a form of disk image formats because the result of the previous step is the form itself. Thus, this optional step can be executed according to the need. For example, the purpose of this project is to develop reference Windows registry data, so it is required to extract specific files (Windows registry hive files) from virtual machine images.

3.2. SETTING UP EXECUTION ENVIRONMENTS

To develop system-generated reference registry data, the initial step is to set up common execution environments for the following steps. The environments established for this project include the following items:

- A physical workstation with Windows 7 (x64, v6.1.7601)
- Oracle VirtualBox (v5.0+) as a type-2 hypervisor
- Base virtual machine images with Windows operating systems (Vista, 7, 8, 8.1, 10 and 10RS1) downloaded from Microsoft
- Three USB storage devices with sample files and applications

3.2.1. NAT network NatCFReDS in VirtualBox

For supporting user actions associated with network drives and remote desktops, a NAT network is required to enable communications between running virtual machines in VirtualBox. It may be configured manually using graphical user interfaces of the VirtualBox Manager program, or users can also utilize a command line tool 'VBoxManage' that is installed by default during the installation process of VirtualBox. The following command line creates a NAT network for this project.

```
VBoxManage natnetwork add --netname NatCFReDS --network "10.11.11.0/24" –enable
```

3.2.2. Common Windows server within NatCFReDS

A common virtual machine is used as a server system to provide a network drive and remote desktop for all base virtual machines. Although this virtual machine may be created by manual operations, in this project, we utilized a batch script of **Table 8** developed for semi-automated procedures.

Table 8. Batch script for importing a common VM: 'cfreds-server'

```
@echo off
:: Reference - https://www.virtualbox.org/manual/ch08.html

:: Import a VM using an OVA file from Microsoft
set vmname="cfreds-server"
set image="C:\VMs\MSEdge.Win10_RS1.Stable.14.14393.VirtualBox\MSEdge - Win10_preview.ova"
set cpus=1
set memory=1024
VBoxManage import %image% --vsys 0 --vmname %vmname% --cpus %cpus% --memory %memory%

:: Set configurations: network (nat), audio (hda) and usb (3.0)
set natname="NatCFReDS"
VBoxManage modifyvm %vmname% --audio dsound --audiocontroller hda --nic1 natnetwork --nat-
network1 %natname% --nicpromisc1 allow-vms --usbxhci on

:: Configure the clipboard and drag-and-drop settings
VBoxManage modifyvm %vmname% --clipboard   hosttoguest
VBoxManage modifyvm %vmname% --draganddrop hosttoguest

:: Start the imported virtual machine and wait for booting (manual check is required)
VBoxManage startvm %vmname%
pause
```

```
:: Set the resulotion (1024 x 768)
VBoxManage controlvm %vmname% setvideomodehint 1024 768 32
pause

:: The following procedures should be done by manual
:: (1) Update Guest Additions (the default guest agent for VirtualBox)
:: (2) Disable Windows Update
:: (3) Set NIC - IP(10.11.11.127) and DNS(8.8.8.8 / 8.8.4.4)
:: (4) Add an admin account (ID: cfreds-server1, PW: cs1nist)
:: (5) Logout and login using the 'cfreds-server1' account
:: (6) Copy sample files to Desktop (‘NETWORK_DIR’ directory)
:: (7) Set a shared directory (\\10.11.11.127\NETWORK_DIR)
:: (8) Enable the remote desktop feature

:: Shutdown and wait for done (manual check is required)
VBoxManage controlvm %vmname% acpipowerbutton
pause

:: Set the second IDE drive (DVD) to empty
VBoxManage storageattach %vmname% --storagectl "IDE Controller" --port 1 --device 0 --medium emptydrive

:: Take a snapshot for storing the current state
VBoxManage snapshot %vmname% take "Snapshot 1" --description "init"
```

At the first stage of the script, a virtual machine titled in ‘cfreds-server’ is added by importing an image file with Window 10 (14.14393). Right after that, there is a command line for setting basic configurations including Network, Audio and USB. When a virtual machine is added, it will be started through VirtualBox automatically, and then we should wait until the booting process is done. At this point, several procedures should be done by manual in order to make the virtual machine available as a server system. The procedures include updating the default guest agent, setting the primary network interface card (NIC) for the common NAT network, adding an administrator account, configuring a shared directory with sample files, and finally enabling the remote desktop feature. Afterward, the script will shut down the virtual machine, and take a snapshot for storing the current state. Detailed options and variables for each procedure are available from **Table 8**.

3.2.3. Base Windows virtual machines

In this project, public virtual machine images downloaded from Microsoft[17] are used as sources of base virtual machines. Microsoft is providing free virtual machines for supporting testing works relating to web-browsers (IE and Edge) installed in various versions of Windows systems. As listed in **Table 9**, we prepared six different versions including Windows Vista, 7, 8, 8.1, 10 (13.10586), and 10 RS1 (14.14393). This virtual machine list includes various versions and they also do not have any license information, so it satisfies basic requirements described in Section 3.1.

Table 9. Target virtual machines from Microsoft

Platform	Windows Name	Arch.	Edition	SHA-1 (downloaded OVA files from Microsoft)	Note
VirtualBox	Windows Vista	x86	Enterprise	92CC40D375D05623544BE0E08A01DBBBA0C9DB76	SP2, IE 7
	Windows 7	x86	Enterprise	E6831392D268937F4122EE54ECD6B860C2721D94	SP1, IE 9
	Windows 8	x86	Enterprise Evaluation	652F383F8A6D87BF84668DEB2BA67FF7451F01BC	IE 10
	Windows 8.1	x86	Enterprise Evaluation	58418F724916EFBA9BACD880CC96736B4A09AEA2	IE 11
	Windows 10 (13.10586)	x64	Enterprise Evaluation	4D482EC8FF60E27246648C4AA236091CF2FB40BF	IE 11, Edge
	Windows 10 (14.14393)	x64	Enterprise Evaluation	5EE09F4F93E1C5642203EEF0921924E6F93785A4	IE 11, Edge

These base virtual machines should be imported to VirtualBox before proceeding to the next steps. To provide convenient and repeatable methods for doing that, we developed simple batch scripts based on a script described in Section 3.2.2. **Table 10** shows an example of the scripts for importing a Windows 7 machine. As shown in the table, the script requires a manual procedure to install the latest version of the guest agent (Guest Additions in the case of VirtualBox).

Table 10. Batch script for importing a base VM: 'Win_7_IE09_(CFReDS)'

```
@echo off

:: Import a VM using an OVA file from Microsoft
set vmname="Win_7_IE09_(CFReDS)"
set image="C:\VMs\IE9.Win7.VirtualBox\IE9 - Win7.ova"
set cpus=2
set memory=1024
VBoxManage import %image% --vsys 0 --vmname %vmname% --cpus %cpus% --memory %memory%

:: Set configurations: network (nat), audio (hda) and usb (2.0)
set natname="NatCFReDS"
VBoxManage modifyvm %vmname% --audio dsound --audiocontroller hda --nic1 natnetwork --nat-
network1 %natname% --nicpromisc1 allow-vms --usbehci on

:: Start the imported virtual machine and wait for booting (manual check is required)
VBoxManage startvm %vmname%
pause

:: The following procedures should be done by manual
:: - Update Guest Additions (the default guest agent for VirtualBox)

:: Shutdown and wait for done (manual check is required)
VBoxManage controlvm %vmname% acpipowerbutton
pause

:: Set the second IDE drive (DVD) to empty
VBoxManage storageattach %vmname% --storagectl "IDE" --port 1 --device 0 --medium emptydrive

:: Take a snapshot for storing the current state
VBoxManage snapshot %vmname% take "Snapshot 1" --description "init"
```

[17] Free virtual machines for testing Microsoft IE and Edge (https://developer.microsoft.com/en-us/microsoft-edge/tools/vms/)

The overall procedures for all the virtual machines are basically the same, but Window 8 and higher requires additional procedures to add a new SATA controller and attach an existing disk image to the SATA port 0. That is because Windows 8 introduced 'File History' to allow users to back up files to an external storage drive, so this project uses a prepared disk image file with an NTFS partition (5 GB) for the feature. As an example, **Table 11** shows a script for importing a Windows 10 (RS1) machine.

Table 11. Batch script for importing a base VM: 'Win10RS1_14393_IE11+Edge_(CFReDS)'

```
@echo off

:: Import a VM using an OVA file from Microsoft
set vmname="Win10RS1_14393_IE11+Edge_(CFReDS)"
set image="C:\VMs\MSEdge.Win10_RS1.Stable.14.14393.VirtualBox\MSEdge - Win10_preview.ova"
set cpus=2
set memory=2048
VBoxManage import %image% --vsys 0 --vmname %vmname% --cpus %cpus% --memory %memory%

:: Set configurations: network (nat), audio (hda) and usb (3.0)
set natname="NatCFReDS"
VBoxManage modifyvm %vmname% --audio dsound --audiocontroller hda --nic1 natnetwork --nat-
network1 %natname% --nicpromisc1 allow-vms --usbxhci on

:: Add a new SATA controller
VBoxManage storagectl %vmname% --name "SATA" --add SATA

:: Add an existing disk to SATA 0,0 (for enabling File History feature)
set disk="C:\pyvmpop\example\cfreds-2017-winreg\cfreds_2017_winreg_tiny_disk.vmdk"
VBoxManage storageattach %vmname% --storagectl "SATA" --port 0 --device 0 --type HDD --medium %disk%

:: Start the imported virtual machine and wait for booting (manual check is required)
VBoxManage startvm %vmname%
pause

:: The following procedures should be done by manual
:: - Update Guest Additions (the default guest agent for VirtualBox)

:: Shutdown and wait for done (manual check is required)
VBoxManage controlvm %vmname% acpipowerbutton
pause

:: Set the second IDE drive (DVD) to empty
VBoxManage storageattach %vmname% --storagectl "IDE Controller" --port 1 --device 0 --medium emptydrive

:: Take a snapshot for storing the current state
VBoxManage snapshot %vmname% take "Snapshot 1" --description "init"
```

3.2.4. Removable storage devices

To support user actions relating to external devices, three USB thumb drives are prepared with sample files (NIST/CFReDS and Govdoc1[18]) and applications (without license information). **Table 12** summarizes information pertaining to the removable storage devices. For ease of terminology in this project, we refer to the USB thumb drives as RM1, RM2 and RM3 respectively. In addition, the detailed information including device names, volume names and serial numbers can be used for identifying forensic artifacts from reference Windows registry data as a result of this project.

[18] Govdocs1 (http://digitalcorpora.org/corpora/govdocs)

Table 12. List of prepared removable storage devices

Name	Device information	Volume name	S/N	Description
RM1	SanDisk Cruzer Fit 4GB	RM1-MBR&NTFS	4C530012550531106501	RM1 includes sample files
RM2	SanDisk Cruzer Fit 4GB	RM2-MBR&FAT	4C530012450531101593	RM2 includes applications (including executables and installers) and sample files
RM3	SanDisk Cruzer Fit 4GB	RM3-GPT&NTFS	4C530012230531102000	RM3 includes sample files

As listed in **Table 12**, each removable storage device consists of multiple directories and files to allow users to generate various forensic artifacts. **Table 13-15** show detailed file lists (including version information of applications if available) stored in the storage devices. In the case of RM1 and RM3, there exist sample files categorized by file formats, so these files can be used for supporting some user actions such as opening files and launching programs. In the case of RM2, there are additional files relating to well-known applications, which are grouped by application types including web-browsers, documents, archives, multimedia, cloud-services, P2P (peer-to-peer) and anti-forensics. Using these application-related files, it is possible to perform various user actions such as installing applications and launching them with user-controlled arguments.

Table 13. File list stored in 'RM1' that includes sample files

Directory tree and files	Note
RM1+Samples	
RM1+Samples\dir-1\executable1.exe	HxD Hex Editor v1.7.7.0
RM1+Samples\dir-1\executable2.exe	Sysinternals Process Explorer v16.5.0.0
RM1+Samples\dir-1\p1.torrent RM1+Samples\dir-1\p2.torrent	Torrent files
RM1+Samples\dir-1\dir-1-1\audio1.mp3 RM1+Samples\dir-1\dir-1-1\audio2.mp3 RM1+Samples\dir-1\dir-1-1\audio3.wav RM1+Samples\dir-1\dir-1-1\audio4.wav	Audio files
RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video1.mp4 RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video2.mp4 RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video3.avi RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video4.avi RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video5.mov RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video6.MOV RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video7.wmv RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video8.wmv RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video9.3gp RM1+Samples\dir-1\dir-1-1\dir-1-1-1\video10.3gp	Video files
RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image1.png RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image2.png RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image3.tiff RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image4.tiff RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image5.gif RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image6.gif RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image7.jpg RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image8.jpg RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image9.bmp RM1+Samples\dir-1\dir-1-1\dir-1-1-2\image10.bmp	Image files
RM1+Samples\dir-1\dir-1-2\document1.pdf RM1+Samples\dir-1\dir-1-2\document2.pdf RM1+Samples\dir-1\dir-1-2\document3.pptx RM1+Samples\dir-1\dir-1-2\document4.pptx RM1+Samples\dir-1\dir-1-2\document5.docx RM1+Samples\dir-1\dir-1-2\document6.docx RM1+Samples\dir-1\dir-1-2\document7.xlsx RM1+Samples\dir-1\dir-1-2\document8.xlsx	Document files
RM1+Samples\dir-1\dir-1-3\text1.txt RM1+Samples\dir-1\dir-1-3\text2.txt RM1+Samples\dir-1\dir-1-3\text3.html RM1+Samples\dir-1\dir-1-3\text4.html RM1+Samples\dir-1\dir-1-3\text5.xml RM1+Samples\dir-1\dir-1-3\text6.xml	Text files
RM1+Samples\dir-1\dir-1-4\archive1.7z RM1+Samples\dir-1\dir-1-4\archive2.bz2 RM1+Samples\dir-1\dir-1-4\archive3.gz	Archive files

RM1+Samples\dir-1\dir-1-4\archive4.tar RM1+Samples\dir-1\dir-1-4\archive5.rar RM1+Samples\dir-1\dir-1-4\archive6.zip	

Table 14. File list stored in 'RM2' that includes applications and sample files

Directory tree and files	Note
RM2+Apps\#1_web-browser	
RM2+Apps\#1_web-browser\ChromeStandaloneSetup.exe	Google chrome v1.3.31.5
RM2+Apps\#2_document	
RM2+Apps\#2_document\npp.7.1.Installer.exe	Notepad++ v7.1
RM2+Apps\#2_document\Setup.x64.en-us_ProfessionalRetail_NKGG6-WBPCC-HXWMY-6DQGJ-CPQVG_act_1_.exe RM2+Apps\#2_document\Setup.x86.en-us_ProfessionalRetail_NKGG6-WBPCC-HXWMY-6DQGJ-CPQVG_act_1_.exe	Microsoft Office 2016 Public Review
RM2+Apps\#2_document\AdbeRdr11000_mui_Std\Setup.exe ...	Adobe Reader v11 (installer)
RM2+Apps\#3_archive	
RM2+Apps\#3_archive\ 7z1604.exe	7-Zip v16.04
RM2+Apps\#3_archive\peazip_portable-6.1.1.WINDOWS\peazip.exe ...	Peazip v6.1.1 (portable)
RM2+Apps\#4_multimedia	
RM2+Apps\#4_multimedia\PotPlayerSetup.exe	Daum PotPlayer v1.6.63840.0
RM2+Apps\#4_multimedia\vlc-2.2.4-win32.exe	VLC v2.2.4
RM2+Apps\#5_cloud-service	
RM2+Apps\#5_cloud-service\Evernote_6.4.2.3773.exe	Evernote v6.4.2.3773
RM2+Apps\#5_cloud-service\gsync_enterprise.msi	Google Drive (created at 2016-10-12)
RM2+Apps\#6_p2p	
RM2+Apps\#6_p2p\qbittorrent_3.3.7_setup.exe	Qbittorrent v3.3.7
RM2+Apps\#7_anti-forensics	
RM2+Apps\#7_anti-forensics\ccsetup523.exe	CCleaner v5.23
RM2+Apps\#7_anti-forensics\Eraser 5.8.8 Portable\Eraser.exe ...	Eraser v5.8.8 (portable)
RM2+Samples	
RM2+Samples\dir-1\executable1.exe	HxD Hex Editor v1.7.7.0
RM2+Samples\dir-1\executable2.exe	Sysinternals Process Explorer v16.5.0.0
RM2+Samples\dir-1\p1.torrent RM2+Samples\dir-1\p2.torrent	Torrent files
RM2+Samples\dir-1\dir-1-1\audio1.mp3 RM2+Samples\dir-1\dir-1-1\audio2.mp3 RM2+Samples\dir-1\dir-1-1\audio3.wav RM2+Samples\dir-1\dir-1-1\audio4.wav	Audio files
RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video1.mp4 RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video2.mp4 RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video3.avi RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video4.avi RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video5.mov RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video6.MOV RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video7.wmv RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video8.wmv RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video9.3gp RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video10.3gp	Video files
RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image1.png RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image2.png RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image3.tiff RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image4.tiff RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image5.gif RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image6.gif RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image7.jpg RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image8.jpg RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image9.bmp RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image10.bmp	Image files
RM2+Samples\dir-1\dir-1-2\document1.pdf RM2+Samples\dir-1\dir-1-2\document2.pdf RM2+Samples\dir-1\dir-1-2\document3.pptx RM2+Samples\dir-1\dir-1-2\document4.pptx RM2+Samples\dir-1\dir-1-2\document5.docx RM2+Samples\dir-1\dir-1-2\document6.docx RM2+Samples\dir-1\dir-1-2\document7.xlsx RM2+Samples\dir-1\dir-1-2\document8.xlsx	Document files

RM2+Samples\dir-1\dir-1-3\text1.txt RM2+Samples\dir-1\dir-1-3\text2.txt RM2+Samples\dir-1\dir-1-3\text3.html RM2+Samples\dir-1\dir-1-3\text4.html RM2+Samples\dir-1\dir-1-3\text5.xml RM2+Samples\dir-1\dir-1-3\text6.xml	Text files
RM2+Samples\dir-1\dir-1-4\archive1.7z RM2+Samples\dir-1\dir-1-4\archive2.bz2 RM2+Samples\dir-1\dir-1-4\archive3.gz RM2+Samples\dir-1\dir-1-4\archive4.tar RM2+Samples\dir-1\dir-1-4\archive5.rar RM2+Samples\dir-1\dir-1-4\archive6.zip	Archive files

Table 15. File list stored in 'RM3' that includes sample files

Directory tree and files	Note
RM3+Samples	
RM3+Samples\dir-1\executable1.exe	HxD Hex Editor v1.7.7.0
RM3+Samples\dir-1\executable2.exe	Sysinternals Process Explorer v16.5.0.0
RM3+Samples\dir-1\p1.torrent RM3+Samples\dir-1\p2.torrent	Torrent files
RM3+Samples\dir-1\dir-1-1\audio1.mp3 RM3+Samples\dir-1\dir-1-1\audio2.mp3 RM3+Samples\dir-1\dir-1-1\audio3.wav RM3+Samples\dir-1\dir-1-1\audio4.wav	Audio files
RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video1.mp4 RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video2.mp4 RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video3.avi RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video4.avi RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video5.mov RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video6.MOV RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video7.wmv RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video8.wmv RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video9.3gp RM3+Samples\dir-1\dir-1-1\dir-1-1-1\video10.3gp	Video files
RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image1.png RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image2.png RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image3.tiff RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image4.tiff RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image5.gif RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image6.gif RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image7.jpg RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image8.jpg RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image9.bmp RM3+Samples\dir-1\dir-1-1\dir-1-1-2\image10.bmp	Image files
RM3+Samples\dir-1\dir-1-2\document1.pdf RM3+Samples\dir-1\dir-1-2\document2.pdf RM3+Samples\dir-1\dir-1-2\document3.pptx RM3+Samples\dir-1\dir-1-2\document4.pptx RM3+Samples\dir-1\dir-1-2\document5.docx RM3+Samples\dir-1\dir-1-2\document6.docx RM3+Samples\dir-1\dir-1-2\document7.xlsx RM3+Samples\dir-1\dir-1-2\document8.xlsx	Document files
RM3+Samples\dir-1\dir-1-3\text1.txt RM3+Samples\dir-1\dir-1-3\text2.txt RM3+Samples\dir-1\dir-1-3\text3.html RM3+Samples\dir-1\dir-1-3\text4.html RM3+Samples\dir-1\dir-1-3\text5.xml RM3+Samples\dir-1\dir-1-3\text6.xml	Text files
RM3+Samples\dir-1\dir-1-4\archive1.7z RM3+Samples\dir-1\dir-1-4\archive2.bz2 RM3+Samples\dir-1\dir-1-4\archive3.gz RM3+Samples\dir-1\dir-1-4\archive4.tar RM3+Samples\dir-1\dir-1-4\archive5.rar RM3+Samples\dir-1\dir-1-4\archive6.zip	Archive files

3.3. DEFINITION OF USER ACTIONS RELATED TO WINDOWS REGISTRY

This step defines possible user actions associated with Windows registry to develop a common scenario at the next step. As listed in **Table 16**, a variety of actions are used for representing user behaviors that include not only basic classes such as starting/restarting a system and sending keyboard events, but also special classes for Windows such as enabling File History and creating a volume shadow copy. It is important to note that we primarily consider typical features of Windows OS rather than applications, so the action list needs to be updated if corpus creators need to consider other user actions.

Table 16. User actions related to Windows Registry

Action Class	Action Name	Description
Common	Start	- Start a Windows system
	Shutdown	- Shut down the current system
	Restart	- Restart the current system
	Close a window	- Close the current window
	Maximize a window	- Maximize the current window
	Set clipboard	- Copy data to Clipboard
	Check Notification Center	- Check messages of Notification Center (Windows 10)
	Create a virtual desktop	- Create a new virtual desktop (Windows 10)
	Close a virtual desktop	- Close the current virtual desktop (Windows 10)
Input Device	Send keyboard events	- Send keyboard events to the current system
	Send mouse events	- Send mouse events to the current system
Configuration	Set date and time	- Set date and time of the current system
	Change timezone	- Change(set) time zone information
	Disable/Enable NICs	- Disable/Enable network adapter(s)
	Configure NIC	- Configure network adapter(s) (→ set IP and DNS)
	Configure audit policy	- Configure audit policies
	Configure Eventlog setting	- Configure Eventlog settings
	Disable Windows update	- Disable the Windows update feature
	Disable/Enable UAC	- Disable/Enable the User Access Control (Windows Vista or higher)
	Enable File History	- Enable the File History feature (Windows 8 or higher)
	Configure File History	- Configure the File History feature (Windows 8 or higher)
	Disable auto logon option	- Turn off the automatic logon option using 'netplwiz.exe'
	Disable Edge save prompt	- Disable the save prompt of Edge browser
Account	Add a local account	- Add a new local account
	Add an email account	- Add a Microsoft e-mail account (Windows 8 or higher)
	Delete an account	- Delete an existing account
	Change account setting	- Change the password and full name of an existing account
	Logon an account	- Logon using keyboard events with an account ID and password
	Logoff	- Logoff from the current session
Registry	Set a registry value	- Set a registry value
	Delete registry data	- Delete registry data including keys, values and data
Process	Launch a program	- Launch a program with arguments (option, shortcut, and file path)
	Launch a Windows Store app	- Launch a Windows Store app using Windows Search

	Terminate a process	- Terminate a running process
	Control a web browser	- Control a web browser with assuming that the active process is a web browser program such as IE, Edge and Chrome - The controllable actions include visiting web-sites, searching keywords, downloading files, bookmarking web-sites and so on
Application	Install a program	- Launch an installer file with arguments
	Install a Windows Store app	- Install an app from Windows Store (Windows 8.1 or higher)
	Uninstall a program	- Uninstall a program
	Uninstall a Windows Store app	- Uninstall a Windows Store app
File system	Open a shell	- Open the default shell (Windows Explorer)
	Change directory	- Change(traverse) directories in the default shell
	Copy files	- Copy files (including directories)
Portable Device	Attach a USB device	- Attach a USB device with a serial number
	Detach a USB device	- Detach a connected USB device
Search	Search a keyword	- Search a keyword using Windows Search feature
Share	Share a directory	- Share a directory (after creating a directory if it does not exist)
Network Drive	Connect to a network drive	- Connect to a network drive
	Map a network drive	- Map a network drive as a local drive
Remote Desktop	Connect to a remote desktop	- Connect to a remote desktop using ‘mstsc.exe’
	Disconnect from a remote desktop	- Disconnect from a connected remote desktop
System Backup	Create a restore point	- Create a restore point (Windows XP or lower) or volume shadow copy (Windows Vista and higher)

3.4. DETAILED SCENARIO DESCRIPTIONS

Using user actions defined in the previous step, this step develops a common scenario (that consists of a sequence of user actions) depicting user behaviors to create forensically meaningful artifacts into virtual machines prepared for this project. For doing that, we constitute nine action stages such as pre-requirement, OS configuration, account, external device, application, special feature, and anti-forensics. Each action stage is composed of multiple user actions including detailed descriptions of each action. As a result, **Table 17** lists the common sequence of user actions for developing reference Windows systems.

Table 17. A sequence of user actions for developing reference Windows systems

Action	Windows Version	Description	Note
ACTION STAGE 0 – Pre-Requirement			
Start	*	Start a VM	
Logon	*	Logon the default account 'IEUser'	
Install programs	XP-	Install .NET 2.0, PowerShell 2.0, and Windows Resource Kits	
Restart	XP-	Restart the current system	
Logon	XP-	Logon the default account 'IEUser'	
Disable UAC	Vista+	Disable the UAC (User Access Control)	
Restart	Vista+	Restart the current system	
Logon	Vista+	Logon the default account 'IEUser'	
Disable Windows Update	*	Disable the Windows update feature	Disable Windows itself and Windows Store update
Install a program	Vista	Install KB2556308 for 'tzutil.exe'	
Restart	*	Restart the current system	
Logon	*	Logon the default account 'IEUser'	
Disable auto logon option	*	Disable the auto logon option	
Add a local account	*	Add a local account → Account Name (CFTT) Password (cftt@nist)	'Administrators' group
Logoff	*	Logoff from the current session 'IEUser'	
Logon	*	'CFTT' account with a valid password	Logon count: 1
ACTION STAGE 1 – OS Configuration			
Change timezone	*	Change the current timezone (UTC-8) to (UTC-05) Eastern Time	
Configure NIC	*	Configure IP address to the default network adapter	IP: 10.11.11.77 Mask: 255.255.255.0 Gateway: 10.11.11.1
Configure NIC	*	Configure DNS servers to the default network adapter	DNS: 8.8.8.8, 8.8.4.4
Restart	*	Restart the current system	
Logon	*	'CFTT' account with a valid password	Logon count: 2
Configure audio policy	*	Update the audit policy (secpol.msc)	[ON] Audit account logon event [ON] Audit system events

Action	Windows Version	Description	Note
Configure Eventlog setting	*	Update the Eventlog configuration	Set maximum log size of 'Security' log file to 81920 KB
Enable File History	8+	Turn on the File History feature with a local drive	
Configure File History	8+	Configure the File History feature	Save copies of files: 10 minutes
Check Notification Center	10+	Check messages of Notification Center	WIN + 'a'
ACTION STAGE 2 – Account			
Add a local account	*	Add a local account → Account Name (Forensics) Password (forensics@nist)	'Administrators' group
Add a local account	*	Add a local account → Account Name (Temporary) Password (12321)	'Administrators' group
Change account settings	*	Change account name from 'Temporary' to 'test' and remove the password of 'Temporary' account	
Logoff	*	Logoff from the current session 'CFTT'	
Logon	*	'Forensics' account with a valid password	Logon count: 1
Logoff	*	Logoff from the current session 'Forensics'	
Logon	*	'CFTT' account with a valid password	Logon count: 3
Add a local account	*	Add a local account → Account Name (CFReDS) Password (cfreds@nist)	'Administrators' group
Logoff	*	Logoff from the current session 'CFTT'	
Logon	*	'Forensics' account with an invalid password	Last login failure time is updated
Logon	*	'CFTT' account with an invalid password	Last login failure time is updated
Logon	*	'CFReDS' account with a valid password	Logon count: 1
Add an email account	8+	Add an e-mail account → Account Name (cftt.user1@outlook.com)	'Administrators' group
Add an email account	8+	Add an e-mail account → Account Name (cftt.user2@outlook.com)	'Administrators' group
Logoff	8+	Logoff from the current session 'CFReDS'	
Logon	8+	'cftt.user1@outlook.com' account with a valid password (In the case of Windows 10, set up a PIN '1234321')	Logon count: 1
Logoff	8+	Logoff from the current session 'cftt.user1@outlook.com'	
Logon	8+	'CFReDS' account with a valid password	[Vista-] Logon count: 1 [8+] Logon count: 2
ACTION STAGE 3 – External Device : Traversing directories and files stored in an external device			
Attach a USB device	*	'RM1' USB flash drive with USB interface → MBR & NTFS (SanDisk Cruzer Fit 4GB, SN: 4C530012550531106501)	RM1 includes sample files
Open a shell	*	Open the default shell (Windows Explorer) with the drive letter of 'RM1'	
Change directory	*	Traverse directories as follows: [Drive Letter] → [RM1+Samples] → [dir-1] → [dir-1-1] → [dir-1] → [dir-1-2] → [dir-1] → [dir-1-3]	
Launch programs and terminate them	*	Open files in [dir-1-3] (→ Launch notepad.exe with files) - \RM1+Samples\dir-1\dir-1-3\text1.txt - \RM1+Samples\dir-1\dir-1-3\text2.txt	
Change directory	*	Traverse directories as follows: [dir-1-3] → [dir-1] → [RM1+Samples] → [Drive Letter]	
Copy files	*	Copy [RM1+Samples] directory to Desktop (%UserProfile%\Desktop)	RM1 → PC

Action	Windows Version	Description	Note
Open a shell	*	Open the default shell (Windows Explorer) with the following path: (%UserProfile%\Desktop\RM1+Samples)	
Change directory	*	Traverse directories as follows: [RM1+Samples] → [dir-1] → [dir-1-1] → [dir-1-1-1]	
Close a window	*	Close the current window (Windows Explorer)	
Detach a USB device	*	'RM1' USB flash drive with USB interface	
Logoff	*	Logoff from the current session 'CFReDS'	
Logon	*	'CFTT' account with an invalid password	Last login failure time is updated
Logon	*	'CFTT' account with a valid password	Logon count: 4
ACTION STAGE 4 – Application Part I : Installing and/or launching applications			
Attach a USB device	*	'RM2' USB flash drive with USB interface → MBR & FAT (SanDisk Cruzer Fit 4GB, SN: 4C530012450531101593)	RM2 includes applications (including executables and installers) and sample files
Open a shell	*	Open the default shell (Windows Explorer) with the drive letter of 'RM2'	
Change directory	*	Traverse directories as follows: [Drive Letter] → [RM2+Apps] → [#1_web-browser]	
Install a program	7+	Install a program: Google Chrome (\RM2+Apps\#1_web-browser\ChromeStandaloneSetup.exe)	Option: /Silent /Install
Launch a program	*	Launch a web browser: Internet Explorer (C:\Program Files\Internet Explorer\iexplore.exe)	Launch IE browser for initializing the program environment
Terminate a process	*	Terminate a web browser: Internet Explorer	
Launch a program	*	Launch a web browser: Internet Explorer (C:\Program Files\Internet Explorer\iexplore.exe)	Launch IE browser again and use it
Control a web browser	*	Control a web browser as follows: - Create a new tab - Visit a web site (www.cftt.nist.gov) - Bookmark the current site - Create a new tab - Visit a web site (www.cfreds.nist.gov) - Bookmark the current site - Close the active tab - Create a new tab - Download a file (http://www.cfreds.nist.gov/data_leakage_case/images/rm%233/cfreds_2015_data_leakage_rm%233_type2.7z) - Close the active tab - Create a new tab - Visit a web site (www.google.com) - Search keywords (NIST CFTT, NIST CFReDS) -----> IE 8 or higher - Close the active tab - Create a new tab - Visit a web site (www.bing.com) - Search keywords (NIST CFTT, NIST CFReDS) -----> IE 7 or higher - Close the active tab - Create a new tab -----> IE 10 or higher - Login a web site (live.com) -----> IE 10 or higher - Close the active tab - Create a new tab - Visit a web site (toolcatalog.nist.gov) - Bookmark the current site	[Account for live.com] : ID (cftt.user1@outlook.com)
Terminate a process	*	Terminate a web browser: Internet Explorer	
Launch a program	10+	Launch a web browser: Edge (C:\Windows\SystemApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdge.exe)	Launch Edge browser for initializing the program environment
Terminate a process	10+	Terminate a web browser: Edge	
Disable Edge save prompt	10+	Disable the save prompt of the Edge browser	

Action	Windows Version	Description	Note
Launch a program	10+	Launch a web browser: Edge (C:\Windows\SystemApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdge.exe)	Launch Edge browser again and use it
Control a web browser	10+	Control a web browser as follows: - Create a new tab - Visit a web site (www.cftt.nist.gov) - Bookmark the current site - Create a new tab - Visit a web site (www.cfreds.nist.gov) - Bookmark the current site - Close the active tab - Create a new tab - Download a file (http://www.cfreds.nist.gov/data_leakage_case/images/rm%233/cfreds_2015_data_leakage_rm%233_type2.7z) - Close the active tab - Create a new tab - Visit a web site (www.google.com) - Search keywords (NIST CFTT, NIST CFReDS) - Close the active tab - Create a new tab - Visit a web site (www.bing.com) - Search keywords (NIST CFTT, NIST CFReDS) - Close the active tab - Create a new tab - Visit a web site (toolcatalog.nist.gov) - Bookmark the current site	
Terminate a process	10+	Terminate a web browser: Edge	
Launch a program	7+	Launch a web browser: Chrome x86 (C:\Program Files\Google\Chrome\Application\chrome.exe) x64 (C:\Program Files (x86)\Google\Chrome\Application\chrome.exe)	Launch Chrome browser for initializing the program environment
Terminate a process	7+	Terminate a web browser: Chrome	
Launch a program	7+	Launch a web browser: Chrome x86 (C:\Program Files\Google\Chrome\Application\chrome.exe) x64 (C:\Program Files (x86)\Google\Chrome\Application\chrome.exe)	Launch Chrome browser again and use it
Control a web browser	7+	Control a web browser as follows: - Create a new tab - Visit a web site (www.cftt.nist.gov) - Bookmark the current site - Create a new tab - Visit a web site (www.cfreds.nist.gov) - Bookmark the current site - Close the active tab - Create a new tab - Download a file (http://www.cfreds.nist.gov/data_leakage_case/images/rm%233/cfreds_2015_data_leakage_rm%233_type2.7z) - Close the active tab - Create a new tab - Visit a web site (www.google.com) - Search keywords (NIST CFTT, NIST CFReDS) - Close the active tab - Create a new tab - Visit a web site (www.bing.com) - Search keywords (NIST CFTT, NIST CFReDS) - Close the active tab - Create a new tab - Login a web site (live.com) - Close the active tab - Create a new tab - Visit a web site (toolcatalog.nist.gov) - Bookmark the current site	[Account for live.com] : ID (cftt.user1@outlook.com)
Terminate a process	7+	Terminate a web browser: Chrome	
Change directory	*	Traverse directories as follows: [#1_web-browser] → [RM2+Apps] → [#2_document]	
Install a program	7+	Install a program: MS Office 2016 x86 (\RM2+Apps\#2_document\Setup.x86.en-us_ProfessionalRetail_NKGG6-WBPCC-HXWMY-6DQGJ-CPQVG_act_1_.exe) x64 (\RM2+Apps\#2_document\Setup.x64.en-us_ProfessionalRetail_NKGG6-WBPCC-HXWMY-6DQGJ-CPQVG_act_1_.exe)	

Action	Windows Version	Description	Note
Launch a program	7+	Launch a program: MS Word 2016 (C:\Program Files\Microsoft Office\root\Office16\WINWORD.EXE)	
Terminate a process	7+	Terminate a process: MS Word 2016	
Launch a program	7+	Launch a program: MS Excel 2016 (C:\Program Files\Microsoft Office\root\Office16\EXCEL.EXE)	
Terminate a process	7+	Terminate a process: MS Excel 2016	
Launch a program	7+	Launch a program: MS PowerPoint 2016 (C:\Program Files\Microsoft Office\root\Office16\POWERPNT.EXE)	
Terminate a process	7+	Terminate a process: MS PowerPoint 2016	
Install a program	*	Install a program: Adobe Reader v11 (\RM2+Apps\#2_document\AdbeRdr11000_mui_Std\AcroRead.msi)	Option: /passive /norestart disable_arm_service_install="1"
Launch a program	*	Launch a program: Adobe Reader v11 x86 (C:\Program Files\Adobe\Reader 11.0\Reader\AcroRd32.exe) x64 (C:\Program Files (x86)\Adobe\Reader 11.0\Reader\AcroRd32.exe)	
Terminate a process	*	Terminate a process: Adobe Reader v11	
Set a registry value	*	Disable Adobe Reader v11 Updater using the following value: HKLM\SOFTWARE\Policies\Adobe\Acrobat Reader\11.0\FeatureLockDown Value name (bUpdater) DWORD data (00000000)	
Install a program	*	Install a program: Notepad++ (\RM2+Apps\#2_document\npp.7.1.Installer.exe)	Option: /S
Launch a program	*	Launch a program: Notepad++ x86 (C:\Program Files\Notepad++\notepad++.exe) x64 (C:\Program Files (x86)\Notepad++\notepad++.exe)	
Terminate a process	*	Terminate a process: Notepad++	
Change directory	*	Traverse directories as follows: [#2_document] → [RM2+Apps] → [#3_archive]	
Install a program	*	Install a program: 7-Zip (\RM2+Apps\#3_archive\7z1604.msi)	Option: /passive /norestart
Launch a program	*	Launch a program: 7-Zip x86 (C:\Program Files\7-Zip\7zFM.exe) x64 (C:\Program Files (x86)\7-Zip\7zFM.exe)	
Terminate a process	*	Terminate a process: 7-Zip	
Launch a program	*	Launch a program: PeaZip (portable version) (\RM2+Apps\#3_archive\peazip_portable-6.1.1.WINDOWS\peazip.exe)	Launch an executable file from 'RM#2' USB flash drive
Terminate a process	*	Terminate a process: PeaZip	
Change directory	*	Traverse directories as follows: [#3_archive] → [RM2+Apps] → [#4_multimedia]	
Launch a program	*	Launch a program: Windows Media Player (C:\Program Files\Windows Media Player\wmplayer.exe)	
Terminate a process	*	Terminate a process: Windows Media Player	
Install a program	*	Install a program: VLC media player (\RM2+Apps\#4_multimedia\vlc-2.2.4-win32.exe)	Option: /S
Launch a program	*	Launch a program: VLC media player x86 (C:\Program Files\VideoLAN\VLC\vlc.exe) x64 (C:\Program Files (x86)\VideoLAN\VLC\vlc.exe)	
Terminate a process	*	Terminate a process: VLC media player	
Install a program	*	Install a program: Potplayer (\RM2+Apps\#4_multimedia\PotPlayerSetup.exe)	Option: /S
Launch a program	*	Launch a program: Potplayer x86 (C:\Program Files\DAUM\PotPlayer\PotPlayerMini.exe) x64 (C:\Program Files (x86)\DAUM\PotPlayer\PotPlayerMini.exe)	
Terminate a process	*	Terminate a process: Potplayer	
Change directory	*	Traverse directories as follows: [#4_multimedia] → [RM2+Apps] → [#5_cloud-service]	

Action	Windows Version	Description	Note
Install a program	*	Install a program: Google Drive Sync (\RM2+Apps\#5_cloud-service\gsync_enterprise.msi)	Option: /passive /norestart
Launch a program	*	Launch a program: Google Drive Sync x86 (C:\Program Files\Google\Drive\googledrivesync.exe) x64 (C:\Program Files (x86)\Google\Drive\googledrivesync.exe)	
Terminate a process	*	Terminate a process: Google Drive Sync	
Install a program	*	Install a program: Evernote (\RM2+Apps\#5_cloud-service\Evernote_6.4.2.3773.exe)	Option: /passive /norestart
Launch a program	*	Launch a program: Evernote x86 (C:\Program Files\Evernote\Evernote\Evernote.exe) x64 (C:\Program Files (x86)\Evernote\Evernote\Evernote.exe)	
Terminate a process	*	Terminate a process: Evernote	
Change directory	*	Traverse directories as follows: [#5_cloud-service] → [RM2+Apps] → [#6_p2p]	
Install a program	*	Install a program: qBittorrent (\RM2+Apps\#6_p2p\qbittorrent_3.3.7_setup.exe)	Option: /S
Launch a program	*	Launch a program: qBittorrent x86 (C:\Program Files\qBittorrent\qbittorrent.exe) x64 (C:\Program Files (x86)\qBittorrent\qbittorrent.exe)	
Terminate a process	*	Terminate a process: qBittorrent	
Change directory	*	Traverse directories as follows: [#6_p2p] → [RM2+Apps] → [#7_anti-forensics]	
Install a program	*	Install a program: CCleaner (\RM2+Apps\#7_anti-forensics\ccsetup523.exe)	Option: /S
Launch a program	*	Launch a program: CCleaner (C:\Program Files\CCleaner\CCleaner.exe)	
Terminate a process	*	Terminate a process: CCleaner	
Launch a program	*	Launch a program: Eraser (portable version) (\RM2+Apps\#7_anti-forensics\Eraser 5.8.8 Portable\Eraser.exe)	Launch an executable file from 'RM#2' USB flash drive
Terminate a process	*	Terminate a process: Eraser	
Close a window	*	Close the current window (Windows Explorer)	
Detach a USB device	*	'RM2' USB flash drive with USB interface	
Launch a Windows Store app	8+	Launch a Windows Store app: Weather	Using Windows Search
Terminate a process	8+	Terminate a process: Weather	
Launch a program	10+	Launch a `Windows system program`: Photos (shell:AppsFolder\Microsoft.Windows.Photos_8wekyb3d8bbwe!App)	Using a registered shortcut
Terminate a process	10+	Terminate a process: Photos	
Launch a Program	10+	Launch a `Windows system program`: Calculator (shell:AppsFolder\Microsoft.WindowsCalculator_8wekyb3d8bbwe!App)	Using a registered shortcut
Terminate a process	10+	Terminate a process: Calculator	
Check Notification Center	10+	Check messages of Notification Center	WIN + 'a'
Logoff	8.1+	Logoff from the current session 'CFTT'	
Logon	8.1+	'cftt.user1@outlook.com' account with a valid password or PIN - Windows 8.1 → Enter a valid password - Windows 10 → Enter a valid PIN (1234321)	Logon count: 2
Install a Windows Store app	8.1+	Install a Windows Store app: ZIP Opener	(1) Launch 'Store' app using Windows Search (2) Search the name of application (3) Select the app (4) Click 'Install' button

Action	Windows Version	Description	Note
Launch a Windows Store app	8.1+	Launch a Windows Store app: ZIP Opener	Using Windows Search
Terminate a process	8.1+	Terminate a process: ZIP Opener	
Install a Windows Store app	8.1+	Install a Windows Store app: Dropbox	(1) Launch 'Store' app using Windows Search (2) Search the name of application (3) Select the app (4) Click 'Install' button
Launch a Windows Store app	8.1+	Launch a Windows Store app: Dropbox	Using Windows Search
Terminate a process	8.1+	Terminate a process: Dropbox	
Install a Windows Store app	8.1+	Install a Windows Store app: Facebook	(1) Launch 'Store' app using Windows Search feature (2) Search the name of application (3) Select the app (4) Click 'Install' button
Launch a Windows Store app	8.1+	Launch a Windows Store app: Facebook	Using Windows Search
Terminate a process	8.1+	Terminate a process: Facebook	
Install a Windows Store app	8.1+	Install a Windows Store app: TeamViewer	(1) Launch 'Store' app using Windows Search (2) Search the name of application (3) Select the app (4) Click 'Install' button
Logoff	8.1+	Logoff from the current session 'cftt.user1@outlook.com'	
Logon	8.1+	'CFTT' account with a valid password	[8-] Logon count: 4 [8.1+] Logon count: 5
Create a restore point	*	Create a Restore Point (XP or lower) or a Volume Shadow Copy (Vista or higher)	[Backup description] : 1st manual restore point [Backup Type] : APPLICATION_INSTALL
ACTION STAGE 5 – Application Part II : Launching(opening) files with specific applications			
Attach a USB device	*	'RM2' USB flash drive with USB interface → MBR & FAT (SanDisk Cruzer Fit 4GB, SN: 4C530012450531101593)	RM2 includes applications (including executables and installers) and sample files
Open a shell	*	Open the default shell (Windows Explorer) with the drive letter of 'RM2'	
Change directory	*	Traverse directories as follows: [Drive Letter] → [RM2+Samples] → [dir-1] → [dir-1-1] → [dir-1] → [dir-1-2] → [dir-1] → [dir-1-3] → [dir-1] → [dir-1-4] → [dir-1]	
Launch a program	*	Launch a program: HxD (\RM2+Samples\dir-1\executable1.exe)	
Terminate a process	*	Terminate a process: HxD	
Launch a program	*	Launch a program: Process Explorer (\RM2+Samples\dir-1\executable2.exe)	
Terminate a process	*	Terminate a process: Process Explorer	
Change directory	*	Traverse directories as follows: [dir-1] → [RM2+Samples] → [Drive Letter]	
Copy files	*	Copy [RM2+Samples] directory to Desktop (%UserProfile%\Desktop)	RM2 → PC
Close a window	*	Close the current window (Windows Explorer)	
Detach a USB device	*	'RM2' USB flash drive with USB interface	
Open a shell	*	Open the default shell (Windows Explorer) with the following path: (%UserProfile%\Desktop\RM2+Samples)	Open [RM2+Samples] directory in Desktop
Change directory	*	Traverse directories as follows: [RM2+samples] → [dir-1]	

Action	Windows Version	Description	Note
Launch a program and then terminate the process	*	Launch a program (qBittorrent) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\p1.torrent - %UserProfile%\Desktop\RM2+Samples\dir-1\p2.torrent	
Change directory	*	Traverse directories as follows: [dir-1] → [dir-1-1] → [dir-1-1-1]	
Launch a program and then terminate the process	*	Launch a program (Windows Media Player) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video1.mp4 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video2.mp4 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video3.avi - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video4.avi - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video5.mov - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video6.mov - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video7.wmv - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video8.wmv - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video9.3gp - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video10.3gp	
Launch a program and then terminate the process	*	Launch a program (VLC media player) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video1.mp4 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video2.mp4 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video3.avi - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video4.avi - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video5.mov - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video6.mov - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video7.wmv - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video8.wmv - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video9.3gp - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video10.3gp	
Launch a program and then terminate the process	*	Launch a program (PotPlayer) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video1.mp4 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video2.mp4 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video3.avi - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video4.avi - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video5.mov - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video6.mov - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video7.wmv - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video8.wmv - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video9.3gp - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-1\video10.3gp	
Change directory	*	Traverse directories as follows: [dir-1-1-1] → [dir-1-1] → [dir-1-1-2]	
Launch a program and then terminate the process	*	Launch a program (MS Paint) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image1.png - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image2.png - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image3.tiff - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image4.tiff - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image5.gif - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image6.gif - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image7.jpg - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image8.jpg - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image9.bmp - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image10.bmp	C:\Windows\System32\mspaint.exe
Launch a program and then terminate the process	*	Launch a program (Windows Photo) with the specific file, and then terminate the process	[Vista] Windows Photo Gallery

Action	Windows Version	Description	Note
		Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image1.png - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image2.png - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image3.tiff - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image4.tiff - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image5.gif - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image6.gif - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image7.jpg - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image8.jpg - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image9.bmp - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\dir-1-1-2\image10.bmp	[8+] Windows Photo Viewer
Change directory	*	Traverse directories as follows: [dir-1-1-2] → [dir-1-1]	
Launch a program and then terminate the process	*	Launch a program (Windows Media Player) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio1.mp3 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio2.mp3 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio3.wav - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio4.wav	
Launch a program and then terminate the process	*	Launch a program (VLC media player) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio1.mp3 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio2.mp3 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio3.wav - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio4.wav	
Launch a program and then terminate the process	*	Launch a program (PotPlayer) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio1.mp3 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio2.mp3 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio3.wav - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-1\audio4.wav	
Change directory	*	Traverse directories as follows: [dir-1-1] → [dir-1] → [dir-1-2]	
Launch a program and then terminate the process	*	Launch a program (Adobe Reader) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document1.pdf - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document2.pdf	
Launch a program and then terminate the process	7+	Launch a program (MS Office 2016) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document3.pptx - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document4.pptx - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document5.docx - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document6.docx - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document7.xlsx - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-2\document8.xlsx	
Change directory	*	Traverse directories as follows: [dir-1-2] → [dir-1] → [dir-1-3]	
Launch a program and then terminate the process	*	Launch a program (notepad.exe) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text1.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text2.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text3.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text4.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text5.xml - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text6.xml	C:\Windows\System32\notepad.exe

Action	Windows Version	Description	Note
Launch a program and then terminate the process	*	Launch a program (Notepad++) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text1.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text2.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text3.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text4.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text5.xml - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text6.xml	
Launch a program and then terminate the process	*	Launch a program (MS Internet Explorer) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text1.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text2.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text3.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text4.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text5.xml - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text6.xml	
Launch a program and then terminate the process	*	Launch a program (Google Chrome) with the specific file, and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text1.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text2.txt - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text3.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text4.html - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text5.xml - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-3\text6.xml	
Change directory	*	Traverse directories as follows: [dir-1-3] → [dir-1] → [dir-1-4]	
Launch a program and then terminate the process	*	Launch a program (7-Zip) with the specific file, extract files here (Press 'F5' and 'Enter'), and then terminate the process Repeat for the following sample files: - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-4\archive1.7z - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-4\archive2.bz2 - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-4\archive3.gz - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-4\archive4.tar - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-4\archive5.rar - %UserProfile%\Desktop\RM2+Samples\dir-1\dir-1-4\archive6.zip	
Close a window	*	Close the current window (Windows Explorer)	
Attach a USB device	Vista+	'RM3' USB flash drive with USB interface → GPT & NTFS (SanDisk Cruzer Fit 4GB, SN: 4C530012230531102000)	RM3 includes sample files
Open a shell	Vista+	Open the default shell (Windows Explorer) with the drive letter of 'RM3'	
Change directory	Vista+	Traverse directories as follows: [Drive Letter] → [RM3+Samples] → [dir-1] → [dir-1-1] → [dir-1] → [dir-1-2] → [dir-1] → [dir-1-3] → [dir-1] → [dir-1-4] → [dir-1]	
Launch a program	Vista+	Launch a program: HxD (\RM3+Samples\dir-1\executable1.exe)	
Terminate a process	Vista+	Terminate a process: HxD	
Launch a program	Vista+	Launch a program: Process Explorer (\RM3+Samples\dir-1\executable2.exe)	
Terminate a process	Vista+	Terminate a process: Process Explorer	
Change directory	Vista+	Traverse directories as follows: [dir-1] → [RM3+Samples] → [Drive Letter]	
Copy files	Vista+	Copy [RM3+Samples] directory to Desktop (%UserProfile%\Desktop)	RM3 → PC
Close a window	Vista+	Close the current window (Windows Explorer)	

Action	Windows Version	Description	Note
Detach a USB device	Vista+	'RM3' USB flash drive with USB interface	
ACTION STAGE 6 – Special Feature Part I : Searching keywords and sharing directories in Windows			
Search keywords	XP-	Search keywords using Windows Search feature [XP] - Launch Windows Search for files or folders (Press 'F3')	[keywords] - hello (English) - ¡Hola! (Spanish) - Здравствуйте! (Russian)
Search keywords	Vista+	Search keywords using Windows Search feature [Windows Vista & 7] - Launch Windows Search for files or folders (Press 'F3') [Windows 8] - Launch Windows Search for files or folders (Press 'F3') - Launch Windows Search for apps (Press 'WIN + q') - Launch Windows Search for settings (Press 'WIN + w') [Windows 8.1] - Launch Windows Search for files or folders (Press 'F3') - Launch Windows Search for settings (Press 'WIN + w') - Launch Windows Search for everywhere (Press 'WIN + s') [Windows 10] - Launch Windows Search (Press 'WIN') : A prefix 'folders: ' is used for searching folders : A prefix 'documents: ' is used for searching documents : A prefix 'apps: ' is used for searching apps : A prefix 'settings: ' is used for searching settings * In Windows 10, 'Cortana' process needs to be restarted for doing each keyword search to avoid stopping Windows Search feature	[keywords] - hello (English) - ¡Hola! (Spanish) - Здравствуйте! (Russian) - 안녕하세요 (Korean) - 你好 (Chinese) - 今日は (Japanese) - नमस्ते (Hindi)
Shared a directory	*	Share a directory after creating the directory if it does not exist	[English Path] C:\welcome
Shared a directory	*	Share a directory after creating the directory if it does not exist	[Spanish Path] C:\¡Hola!
Shared a directory	Vista+	Share a directory after creating the directory if it does not exist	[Korean Path] C:\환영합니다
Check Notification Center	10+	Check messages of Notification Center	WIN + 'a'
Create a virtual desktop	10+	Create a new virtual desktop	
Launch a program and then terminate the process	10+	Launch a program: Notepad.exe - Send a keyboard event: WIN + 'r' (Windows Run) - Send a keyboard event: "notepad" - Send a keyboard event: ENTER Write something to the current window - Send a keyboard event: "This is the 1st virtual desktop.\n\n\nLet's save this file." Save the text to a file (%UserProfile%\Documents\1st_virtual_desktop.txt) - Send a keyboard event: CTRL + 's' - Send a keyboard event: "1st_virtual_desktop" - Send a keyboard event: ENTER Terminate the process (= Close the current *Notepad.exe* window)	
Create a virtual desktop	10+	Create a new virtual desktop	
Launch a program and then terminate the process	10+	Launch a program: Notepad.exe - Send a keyboard event: WIN + 'r' (Windows Run) - Send a keyboard event: "notepad" - Send a keyboard event: ENTER Write something to the current window - Send a keyboard event: "This is the 2nd virtual desktop.\n\n\nLet's save this file."	

Action	Windows Version	Description	Note
		Save the text to a file (%UserProfile%\Documents\2nd_virtual_desktop.txt) - Send a keyboard event: CTRL + 's' - Send a keyboard event: "2nd_virtual_desktop" - Send a keyboard event: ENTER Terminate the process (= Close the current *Notepad.exe* window)	
Create a virtual desktop	10+	Create a new virtual desktop	
Launch a program and then terminate the process	10+	Launch a program: Notepad.exe - Send a keyboard event: WIN + 'r' (Windows Run) - Send a keyboard event: "notepad" - Send a keyboard event: ENTER Write something to the current window - Send a keyboard event: "This is the 3rd virtual desktop.\n\n\nLet's save this file." Save the text to a file (%UserProfile%\Documents\3rd_virtual_desktop.txt) - Send a keyboard event: CTRL + 's' - Send a keyboard event: "3rd_virtual_desktop" - Send a keyboard event: ENTER Terminate the process (= Close the current *Notepad.exe* window)	
Close a virtual desktop	10+	Close the current virtual desktop (→ Repeat 3 times for all virtual desktops)	
ACTION STAGE 7 – Special Feature Part II : Connecting a network drive and a remote desktop in Windows			
Connect to a network drive	*	Connect to a network drive using Windows Run - Send a keyboard event: WIN + 'r' - Send a keyboard event: "\\10.11.11.127\NETWORK_DIR" - Send a keyboard event: ENTER - Send a keyboard event: "cfreds-server1" - Send a keyboard event: TAB - Send a keyboard event: "cs1nist" - Send a keyboard event: ENTER	[URL] "\\10.11.11.127\NETWORK_DIR" [ID] cfreds-server1 [PW] cs1nist
Change directory	*	Traverse directories as follows: [Network Drive's URL] → [ND+Samples] → [dir-1] → [dir-1-1] → [dir-1] → [dir-1-2] → [dir-1] → [dir-1-3] → [dir-1] → [dir-1-4]	
Launch a program and then terminate the process	*	Launch a program (7-Zip) with the specific file, and then terminate the process Repeat for the following sample files: - \\10.11.11.127\NETWORK_DIR\ND+Samples\dir-1\dir-1-4\archive1.7z - \\10.11.11.127\NETWORK_DIR\ND+Samples\dir-1\dir-1-4\archive2.bz2 - \\10.11.11.127\NETWORK_DIR\ND+Samples\dir-1\dir-1-4\archive3.gz - \\10.11.11.127\NETWORK_DIR\ND+Samples\dir-1\dir-1-4\archive4.tar - \\10.11.11.127\NETWORK_DIR\ND+Samples\dir-1\dir-1-4\archive5.rar - \\10.11.11.127\NETWORK_DIR\ND+Samples\dir-1\dir-1-4\archive6.zip	
Close a window	*	Close the current window (Windows Explorer)	
Restart	*	Restart the current system	
Logon	*	'CFTT' account with a valid password	[8-] Logon count: 5 [8.1+] Logon count: 6
Map a network drive	*	Map a network drive as a local drive - Send a keyboard event: WIN + 'r' - Send a keyboard event: "Rundll32.exe Shell32.dll,SHHelpShortcuts_RunDLL Connect" - Send a keyboard event: ENTER - Send a keyboard event: SHIFT + TAB - Send a keyboard event: "w" - Send a keyboard event: TAB - Send a keyboard event: "\\10.11.11.127\NETWORK_DIR" - Send a keyboard event: ENTER - Send a keyboard event: "cfreds-server1" - Send a keyboard event: TAB - Send a keyboard event: "cs1nist" - Send a keyboard event: ENTER	[Drive Letter] W [URL] "\\10.11.11.127\NETWORK_DIR" [ID] cfreds-server1 [PW] cs1nist
Open a shell	*	Open the default shell (Windows Explorer) with the drive letter 'W'	

Action	Windows Version	Description	Note
Change directory	*	Traverse directories as follows: [Drive Letter] → [ND+Samples] → [dir-1] → [dir-1-1] → [dir-1] → [dir-1-2]	
Launch a program and then terminate the process	*	Launch a program (Adobe Reader) with the specific file, and then terminate the process Repeat for the following sample files: - w:\ND+Samples\dir-1\dir-1-2\document1.pdf - w:\ND+Samples\dir-1\dir-1-2\document2.pdf	
Launch a program and then terminate the process	7+	Launch a program (MS Office 2016) with the specific file, and then terminate the process Repeat for the following sample files: - w:\ND+Samples\dir-1\dir-1-2\document3.pptx - w:\ND+Samples\dir-1\dir-1-2\document4.pptx - w:\ND+Samples\dir-1\dir-1-2\document5.docx - w:\ND+Samples\dir-1\dir-1-2\document6.docx - w:\ND+Samples\dir-1\dir-1-2\document7.xlsx - w:\ND+Samples\dir-1\dir-1-2\document8.xlsx	
Close a window	*	Close the current window (Windows Explorer)	
Connect to a remote desktop	*	Connect to a remote desktop using 'mstsc.exe' - Send a keyboard event: WIN + 'r' - Send a keyboard event: "mstsc" - Send a keyboard event: ENTER Enter URL, ID and password	[URL] 10.11.11.127 [ID] cfreds-server1 [PW] cs1nist
Disconnect from a remote desktop	*	Disconnect from a connected remote desktop (= Terminate a process: mstsc.exe)	
ACTION STAGE 8 – Anti-Forensics : Deleting registry data and uninstalling applications			
Logoff	8.1+	Logoff from the current session 'CFTT'	
Logon	8.1+	'cftt.user1@outlook.com' account with a valid password or PIN - Windows 8.1 → Enter a valid password - Windows 10 → Enter a valid PIN (1234321)	Logon count: 3
Uninstall a Windows Store app	8.1+	Uninstall a Windows Store app: Dropbox	
Uninstall a Windows Store app	8.1+	Uninstall a Windows Store app: Facebook	
Logoff	8.1+	Logoff from the current session 'cftt.user1@outlook.com'	
Logon	8.1+	'CFTT' account with a valid password	[8-] Logon count: 5 [8.1+] Logon count: 7
Disable NIC	*	Disable the default network adapter	
Restart	*	Restart the current system	
Logon	*	'CFTT' account with a valid password	[8-] Logon count: 6 [8.1+] Logon count: 8
Set date and time	*	Set the date and time of the current system: +24h	
Create a restore point	*	Create a Restore Point (XP or lower) or a Volume Shadow Copy (Vista or higher)	[Backup description] : 2nd manual restore point [Backup Type] : MODIFY_SETTINGS
Delete an account	*	Delete an existing account: Temporary	
Uninstall a program	*	Uninstall a program: qBittorrent	Option: /S
Uninstall a program	*	Uninstall a program: Evernote	Option: /qb
Set date and time	*	Set the date and time of the current system: +24h	

Action	Windows Version	Description	Note
Create a restore point	*	Create a Restore Point (XP or lower) or a Volume Shadow Copy (Vista or higher)	[Backup description] : 3rd manual restore point [Backup Type] : APPLICATION_UNINSTALL
Launch a program	*	Launch a program: CCleaner (C:\Program Files\CCleaner\CCleaner.exe)	Option: /AUTO → Delete artifacts with the default setting of CCleaner
Terminate a process	*	Terminate a process: CCleaner	
Set a registry value	*	Set a registry value as the following: HKCU\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit Value name (LastKey) String data (Computer\HKEY_CURRENT_USER\Software\Piriform)	For setting the last accessed key of 'regedit.exe'
Launch a program	*	Launch a program: regedit.exe - Send a keyboard event: WIN + 'r' - Send a keyboard event: "regedit" - Send a keyboard event: ENTER	Using Windows Run
Delete registry data	*	Delete registry data manually through 'regedit.exe' - Send a keyboard event: DEL - Send a keyboard event: ENTER	[Target registry key] HKCU\Software\Piriform
Close a window	*	Close the current window (Regedit.exe)	
Check Notification Center	10+	Check messages of Notification Center	WIN + 'a'
Set date and time	*	Set the date and time of the current system: -48h	
Shutdown	*	Shutdown the current system	

3.5. VIRTUAL MACHINE POPULATION AND DATA EXTRACTION PROCESSES

In this step, we first implement a scenario established in the previous step as executable codes based on the *pyvmpop*. Afterward, the implemented scenario will be executed to populate base virtual machines and extract Windows registry data from the populated machines.

More specifically, **Table 18** shows actual Python codes excerpted from the implemented scenario, which is a Python class designed for this project. As shown in the initialization method of the class, this scenario has a pre-assigned target OS list that includes six virtual machines registered at VirtualBox. The `start()` method is the entry point function of the class, so there is a for-loop for repeating the population and extraction process with all the target virtual machines. To populate each virtual machine, an instance of 'VmPop' class is created and configured by using the `basic_config()` method of the instance, and then the `scenario()` method tries to call the `action_stage_#()` methods after connecting to the target machine through the current VmPop instance. As an example of implemented action stages, the `action_stage_1()` method shows sample codes for various user actions including changing the time zone, configuring the IP/DNS address, restarting the system and logging on to the system. After completing all population processes, the next few lines are in charge of exporting virtual storages of the target machine to VHD image files by using methods from HIS, and then extracting Windows registry data from the exported image files by calling methods from DES. Finally, the current VmPop instance is terminated by the `close()` method, and the for-loop repeats until all virtual machines are processed.

Table 18. Python code snippet excerpted from 'VmPopScenarioCFReDS2017WinReg' class

```
…(skip)…
from pyvmpop.vmpop import VmPop
from pyvmpop.common_defines import *
from pyvmpop.utility.pt_utils import PtUtils
from pyvmpop.logging.actlog_manager import ActionItem

…(skip)…

class VmPopScenarioCFReDS2017WinReg:

    def __init__(self):
        """The constructor for defining the common variables
        """
        self.os_list = list()  # (vm_name, VmPopOSType)
        self.os_list.append(("Win10RS1_14393_IE11+Edge_(CFReDS)", VmPopOSType.Windows10_64))
        self.os_list.append(("Win10_10586_IE11+Edge_(CFReDS)", VmPopOSType.Windows10_64))
        self.os_list.append(("Win81_IE11_(CFReDS)", VmPopOSType.Windows81))
        self.os_list.append(("Win_8_IE10_(CFReDS)", VmPopOSType.Windows8))
        self.os_list.append(("Win_7_IE09_(CFReDS)", VmPopOSType.Windows7))
        self.os_list.append(("Vista_IE07_(CFReDS)", VmPopOSType.WindowsVista))

        # The VMs from Microsoft has the default account 'IEUser'
        self.default_id = "IEUser"
        self.default_pw = "Passw0rd!"

        self.shared_dir = "..\\pyvmpop_shared"
        self.hv_type = VmPopHypervisor.VBOX
        self.hv_start_mode = VmPopStartMode.CLONE_LINKED

        self.rm1 = "4C5300125505311065O1"  # MBR & NTFS
        self.rm2 = "4C5300124505311015S3"  # MBR & FAT
        self.rm3 = "4C5300122305311020O0"  # GPT & NTFS
        return

    def start(self):
        """Start population processes
        """
        for vm_name, os_type in self.os_list:
            vmpop = VmPop()
            if vmpop.basic_config(hv_type=self.hv_type, os_type=os_type, start_mode=self.hv_start_mode,
                                  shared_dir=self.shared_dir, log_dir=log_dir) is False:
                vmpop.close()
```

```
                continue

            # Populate a VM with defined actions
            if self.scenario(vmpop, vm_name, self.default_id, self.default_pw) is False:
                vmpop.close()
                continue

            # Export virtual storages of the target VM to VHD image files
            images = list()

            dl = vmpop.hypervisor.get_disk_list()
            if isinstance(dl, list):
                for d in dl:
                    output_path = log_dir
                    output_path += "\\{}_{}_{}.{}".format(d.get('controller').split(" ", 1)[0],
                                                          d.get('controller_port'), d.get('device_slot'),
                                                          VmPopImageFormat.VHD.name)
                    output_path = os.path.abspath(output_path)
                    ret = vmpop.hypervisor.export_disk(d.get('id'), output_path, VmPopImageFormat.VHD)
                    if ret is True:
                        images.append(output_path)

            # Extract forensically interesting data from image files
            for image in images:
                if vmpop.extractor.open_image(image) is False:
                    continue
                vmpop.extractor.extract(data_class=[VmPopDataClass.WINDOWS_REGISTRY])

            # Close this VmPop instance
            vmpop.close()

    def scenario(self, vmpop, vm_name, user_id, password):
        """Execute all action stages implemented this VMPOP Scenario
            - Action Stage (AS) 0: Pre-requirements for AS 1 to 8
            - Action Stage (AS) 1 to 8: Reference actions
        """
        try:
            if vmpop.connect_to_vm(vm_name=vm_name, user_id=user_id, password=password) is False:
                return False

            vmpop.hypervisor.start_video_capturing("{}.webm".format(vm_name))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 0", note="PRE-REQUIREMENT"))
            self.action_stage_0(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 0"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 1", note="OS CONFIGURATION"))
            self.action_stage_1(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 1"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 2", note="ACCOUNT"))
            self.action_stage_2(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 2"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 3", note="BASIC ACTIONS with EXTERNAL DEVICE"))
            self.action_stage_3(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 3"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 4", note="APPLICATION Part I"))
            self.action_stage_4(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 4"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 5", note="APPLICATION Part II"))
            self.action_stage_5(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 5"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 6", note="SPECIAL FEATURES Part I"))
            self.action_stage_6(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 6"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 7", note="SPECIAL FEATURES Part II"))
            self.action_stage_7(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 7"))

            vmpop.actlog_mgr.add(ActionItem(desc="[BEGIN] ACTION STAGE 8", note="ANTI-FORENSICS"))
            self.action_stage_8(vmpop)
            vmpop.actlog_mgr.add(ActionItem(desc="[ END ] ACTION STAGE 8"))
```

```
        except:
            return False

        return True

    …(skip)…

    def action_stage_1(self, vmpop):
        """OS CONFIGURATION: Timezone, NIC, EventLog, etc

        Args:
            vmpop (VmPop)
        """
        '''Start with 'CFTT' account'''
        # [PS] change the timezone
        vmpop.automation.change_timezone("Eastern Standard Time", VmPopActionMethod.WIN_PS)

        # [PS] configure IP address to the network adapter "Local Area Connection"
        #   - Name    : if empty (""), the default adapter is selected automatically
        ip = "10.11.11.77"
        mk = "255.255.255.0"
        gw = "10.11.11.1"
        vmpop.automation.configure_nic_ip(name="", mode=VmPopNICMode.STATIC, address=ip, mask=mk, gateway=gw)

        # [PS] configure DNS servers to the network adapter
        dns = ["8.8.8.8", "8.8.4.4"]
        vmpop.automation.configure_nic_dns(name="", mode=VmPopNICMode.STATIC, address=dns)

        # Restart the system & Restore the user session
        vmpop.automation.restart(mode=VmPopFunctionMode.HV)

        # Select "CFTT" account
        if vmpop.vm_os_type.code < VmPopOSType.WindowsVista.code:
            vmpop.hypervisor.send_event_keyboard(['DOWN'], note="Select 'CFTT' account")
        elif VmPopOSType.WindowsVista.code <= vmpop.vm_os_type.code <= VmPopOSType.WindowsVista_64.code:
            vmpop.hypervisor.send_event_keyboard(['E_DEL'], ['CTRL', 'ALT'])
            vmpop.hypervisor.send_event_keyboard(['ENTER'], note="Select 'CFTT' account")
        elif VmPopOSType.Windows7.code <= vmpop.vm_os_type.code <= VmPopOSType.Windows7_64.code:
            vmpop.hypervisor.send_event_keyboard(['ENTER'], note="Select 'CFTT' account")
        elif VmPopOSType.Windows8.code <= vmpop.vm_os_type.code:
            vmpop.hypervisor.send_event_keyboard(['ENTER'], delay_s=2.0, note="Select 'CFTT' account")

        # Logon "CFTT" account with a valid password
        vmpop.automation.logon_account("CFTT", "cftt@nist")

        # [KM] update audit policy (secpol.msc)
        #   - 'ON' audit account logon event
        #   - 'ON' audit system events
        vmpop.automation.configure_audit_policy_using_km()

        # [PS] update Eventlog configuration (eventvwr.msc)
        #   - set maximum log size of 'Security' log file to 80MB (81920KB)
        log_name = 'Security'
        max_size = '80MB'
        vmpop.automation.configure_eventlog(log_name, max_size)

        # == Windows 8 or higher  ==
        if VmPopOSType.Windows8.code <= vmpop.vm_os_type.code:
            # Turn on 'File History' feature with a shared directory
            vmpop.automation.enable_file_history()
            vmpop.automation.configure_file_history()  # Setting 'DPFrequency' to 10 min

        # == Windows 10 ==
        # Check messages in 'Notification Center'
        if VmPopOSType.Windows10.code <= vmpop.vm_os_type.code:
            vmpop.automation.check_notification_center()

        return

…(skip)…
```

3.6. GENERATED REFERENCE DATA INFORMATION

This section summarizes system-generated Windows registry data as a result of all the steps described previous sub-sections. **Table 19-24** list details on the registry data and associated log files. As listed in the tables, reference data include not only registry hive files from an active system partition, but also from all restore points (volume shadow copies in this case).

Table 19. File list of system-generated Windows registry data from Windows Vista

Directory tree and files	Description
[2016-11-02_20.49.06]_Vista_IE07_(CFReDS)\	Root directory
(2016-11-02_20.49.06)_Actions.csv	An action log file
(E_0001)_(A_0024)_(2016-11-02_20.54.10)~(2016-11-02_20.54.32)_Events.csv ... (E_0347)_(A_1807)_(2016-11-04_22.56.14)~(2016-11-02_22.56.20)_Events.csv	Event log files (total 347) created by Procmon.exe
last_progress_log.txt	A progress log file (for debugging)
Vista_IE07_(CFReDS).webm ... Vista_IE07_(CFReDS)-2016-11-03T02-52-03-562866200Z.webm	Recorded video files (total 7)
WINDOWS_REGISTRY\p1\Boot\BCD WINDOWS_REGISTRY\p1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1\Windows\System32\config\SAM WINDOWS_REGISTRY\p1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1\Windows\System32\config\RegBack\COMPONENTS WINDOWS_REGISTRY\p1\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p1\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p1\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p1\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p1\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Boot & System Partition] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) System hives (backup) : COMPNENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss1\Boot\BCD WINDOWS_REGISTRY\p1_vss1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\RegBack\COMPONENTS WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p1_vss1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 1] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) System hives (backup) : COMPNENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss2\Boot\BCD WINDOWS_REGISTRY\p1_vss2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\DEFAULT	[Volume Shadow Copy 2] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives

WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\RegBack\COMPONENTS WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p1_vss2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	: BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) System hives (backup) : COMPNENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss3\Boot\BCD WINDOWS_REGISTRY\p1_vss3\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\RegBack\COMPONENTS WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p1_vss3\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 3] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) System hives (backup) : COMPNENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss4\Boot\BCD WINDOWS_REGISTRY\p1_vss4\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss4\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss4\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss4\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss4\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss4\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss4\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss4\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss4\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\RegBack\COMPONENTS WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p1_vss4\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p1_vss4\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 4] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) System hives (backup) : COMPNENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (5) ETC : SCHEMA.DAT

Table 20. File list of system-generated Windows registry data from Windows 7

Directory tree and files	Description
[2016-11-03_09.09.37]_Win_7_IE09_(CFReDS)\	Root directory
(2016-11-03_09.09.37)_Actions.csv	An action log file
(E_0001)_(A_0022)_(2016-11-03_09.14.01)~(2016-11-03_09.14.22)_Events.csv ... (E_0409)_(A_2107)_(2016-11-05_11.49.27)~(2016-11-03_11.49.33)_Events.csv	Event log files (total 409) created by Procmon.exe
last_progress_log.txt	A progress log file (for debugging)
Win_7_IE09_(CFReDS).webm ... Win_7_IE09_(CFReDS)-2016-11-03T15-44-36-396131200Z.webm	Recorded video files (total 7)
WINDOWS_REGISTRY\p1\Boot\BCD WINDOWS_REGISTRY\p2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2\Windows\System32\config\COMPONENTS	[Boot & System Partition] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives

WINDOWS_REGISTRY\p2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2\Windows\System32\config\SAM WINDOWS_REGISTRY\p2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT WINDOWS_REGISTRY\p2\System Volume Information\Syscache.hve	: BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) ETC : Syscache.hve, SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT WINDOWS_REGISTRY\p2_vss1\System Volume Information\Syscache.hve	[Volume Shadow Copy 1] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (3) ETC : Syscache.hve, SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT WINDOWS_REGISTRY\p2_vss2\System Volume Information\Syscache.hve	[Volume Shadow Copy 2] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (3) ETC : Syscache.hve, SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss3\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss3\Windows\System32\SMI\Store\Machine\SCHEMA.DAT WINDOWS_REGISTRY\p2_vss3\System Volume Information\Syscache.hve	[Volume Shadow Copy 3] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (3) ETC : Syscache.hve, SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss4\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss4\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss4\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss4\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss4\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss4\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss4\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss4\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss4\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss4\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss4\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss4\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss4\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss4\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss4\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss4\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss4\Windows\System32\SMI\Store\Machine\SCHEMA.DAT WINDOWS_REGISTRY\p2_vss4\System Volume Information\Syscache.hve	[Volume Shadow Copy 4] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BCD-Template, COMPONENTS, DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (3) ETC : Syscache.hve, SCHEMA.DAT

Table 21. File list of system-generated Windows registry data from Windows 8

Directory tree and files	Description
[2016-11-06_20.44.11]_Win_8_IE10_(CFReDS)\	Root directory
(2016-11-06_20.44.11)_Actions.csv	An action log file

(E_0001)_(A_0030)_(2016-11-06_20.52.06)~(2016-11-06_20.52.34)_Events.csv ... (E_0416)_(A_2345)_(2016-11-09_00.00.02)~(2016-11-07_00.00.08)_Events.csv	Event log files (total 416) created by Procmon.exe
last_progress_log.txt	A progress log file (for debugging)
Win_8_IE10_(CFReDS).webm ... Win_8_IE10_(CFReDS)-2016-11-07T04-55-33-540922800Z.webm	Recorded video files (total 7)
WINDOWS_REGISTRY\p1\Boot\BCD WINDOWS_REGISTRY\p2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Windows\AppCompat\Programs\Amcache.hve WINDOWS_REGISTRY\p2\Windows\System32\config\BBI WINDOWS_REGISTRY\p2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2\Windows\System32\config\SAM WINDOWS_REGISTRY\p2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Boot & System Partition] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (5) Application compatibility : Amcache.hve (6) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Windows\AppCompat\Programs\Amcache.hve WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\BBI WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2_vss1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 1] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (3) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Windows\AppCompat\Programs\Amcache.hve WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\BBI WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SECURITY	[Volume Shadow Copy 2] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (3) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT

WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2_vss2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	
WINDOWS_REGISTRY\p2_vss3\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Windows\AppCompat\Programs\Amcache.hve WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\BBI WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2_vss3\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 3] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (3) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT

Table 22. File list of system-generated Windows registry data from Windows 8.1

Directory tree and files	Description
[2016-11-11_19.46.22]_Win81_IE11_(CFReDS)\	Root directory
(2016-11-11_19.46.22)_Actions.csv	An action log file
(E_0001)_(A_0027)_(2016-11-11_19.50.46)~(2016-11-11_19.51.19)_Events.csv ... (E_0433)_(A_2505)_(2016-11-13_23.23.05)~(2016-11-11_23.23.11)_Events.csv	Event log files (total 433) created by Procmon.exe
last_progress_log.txt	A progress log file (for debugging)
Win81_IE11_(CFReDS).webm ... Win81_IE11_(CFReDS)-2016-11-12T04-17-55-756925800Z.webm	Recorded video files (total 7)
WINDOWS_REGISTRY\p1\Boot\BCD WINDOWS_REGISTRY\p2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2\Windows\AppCompat\Programs\Amcache.hve WINDOWS_REGISTRY\p2\Windows\System32\config\BBI WINDOWS_REGISTRY\p2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2\Windows\System32\config\SAM WINDOWS_REGISTRY\p2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Boot & System Partition] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (5) Application compatibility : Amcache.hve (6) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss1\Windows\AppCompat\Programs\Amcache.hve	[Volume Shadow Copy 1] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM

WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\BBI WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2_vss1\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2_vss1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	(3) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss2\Windows\AppCompat\Programs\Amcache.hve WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\BBI WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2_vss2\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2_vss2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 2] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (3) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p2_vss3\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\cfttu_000\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\cfttu_000\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p2_vss3\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p2_vss3\Windows\AppCompat\Programs\Amcache.hve WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\BBI WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\ELAM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SAM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\DEFAULT WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SAM WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SECURITY WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SOFTWARE WINDOWS_REGISTRY\p2_vss3\Windows\System32\config\RegBack\SYSTEM WINDOWS_REGISTRY\p2_vss3\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 3] (1) User hives : NTUSER.DAT & UsrClass.dat (2) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (3) System hives (backup) : DEFAULT, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT

Table 23. File list of system-generated Windows registry data from Windows 10 (10586)

Directory tree and files	Description
[2016-11-12_16.52.41]_Win10_10586_IE11+Edge_(CFReDS)\	Root directory
(2016-11-12_16.52.41)_Actions.csv	An action log file
(E_0001)_(A_0027)_(2016-11-12_16.57.47)~(2016-11-12_16.58.10)_Events.csv ... (E_0467)_(A_2763)_(2016-11-14_20.59.50)~(2016-11-14_20.59.57)_Events.csv	Event log files (total 467) created by Procmon.exe
last_progress_log.txt	A progress log file (for debugging)
Win10_10586_IE11+Edge_(CFReDS).webm ... Win10_10586_IE11+Edge_(CFReDS)-2016-11-13T01-54-30-009196600Z.webm	Recorded video files (total 7)

WINDOWS_REGISTRY\p1\Boot\BCD WINDOWS_REGISTRY\p1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\cfttu\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1\Windows\System32\config\BBI WINDOWS_REGISTRY\p1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1\Windows\System32\config\SAM WINDOWS_REGISTRY\p1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Boot & System Partition] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss1\Boot\BCD WINDOWS_REGISTRY\p1_vss1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\cfttu\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\BBI WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss1\Windows\System32\SMI\Store\Machine	[Volume Shadow Copy 1] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss2\Boot\BCD WINDOWS_REGISTRY\p1_vss2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\cfttu\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\BBI WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 2] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss3\Boot\BCD WINDOWS_REGISTRY\p1_vss3\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\cfttu\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat	[Volume Shadow Copy 3] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives

WINDOWS_REGISTRY\p1_vss3\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\BBI WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss3\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	: BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT

Table 24. File list of system-generated Windows registry data from Windows 10RS1 (14393)

Directory tree and files	Description
[2016-11-13_17.35.22]_Win10RS1_14393_IE11+Edge_(CFReDS)\	Root directory
(2016-11-03_09.09.37)_Actions.csv	An action log file
(E_0001)_(A_0027)_(2016-11-13_17.40.48)~(2016-11-13_17.41.09)_Events.csv ... (E_0467)_(A_2769)_(2016-11-15_21.18.11)~(2016-11-13_21.18.17)_Events.csv	Event log files (total 467) created by Procmon.exe
last_progress_log.txt	A progress log file (for debugging)
Win10RS1_14393_IE11+Edge_(CFReDS).webm ... Win10RS1_14393_IE11+Edge_(CFReDS)-2016-11-14T02-13-37-244303800Z.webm	Recorded video files (total 7)
WINDOWS_REGISTRY\p1\Boot\BCD WINDOWS_REGISTRY\p1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\cfttu\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1\Windows\System32\config\BBI WINDOWS_REGISTRY\p1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1\Windows\System32\config\SAM WINDOWS_REGISTRY\p1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Boot & System Partition] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss1\Boot\BCD WINDOWS_REGISTRY\p1_vss1\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\cfttu\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1_vss1\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss1\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\BBI WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss1\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss1\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 1] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss2\Boot\BCD WINDOWS_REGISTRY\p1_vss2\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\cfttu\NTUSER.DAT	[Volume Shadow Copy 2] (1) Boot Configuration Data : BCD

WINDOWS_REGISTRY\p1_vss2\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1_vss2\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss2\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\BBI WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss2\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss2\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	(2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT
WINDOWS_REGISTRY\p1_vss3\Boot\BCD WINDOWS_REGISTRY\p1_vss3\Users\CFReDS\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\CFReDS\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\CFTT\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\CFTT\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\cfttu\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\cfttu\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\Default\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\Forensics\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\Forensics\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\IEUser\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\IEUser\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Users\sshd_server\NTUSER.DAT WINDOWS_REGISTRY\p1_vss3\Users\sshd_server\AppData\Local\Microsoft\Windows\UsrClass.dat WINDOWS_REGISTRY\p1_vss3\Windows\appcompat\Programs\Amcache.hve WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\BBI WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\BCD-Template WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\COMPONENTS WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\DEFAULT WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\DRIVERS WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\ELAM WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SAM WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SECURITY WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SOFTWARE WINDOWS_REGISTRY\p1_vss3\Windows\System32\config\SYSTEM WINDOWS_REGISTRY\p1_vss3\Windows\System32\SMI\Store\Machine\SCHEMA.DAT	[Volume Shadow Copy 3] (1) Boot Configuration Data : BCD (2) User hives : NTUSER.DAT & UsrClass.dat (3) System hives : BBI, BCD-Template, COMPONENTS, DEFAULT, DRIVERS, ELAM, SAM, SECURITY, SOFTWARE, SYSTEM (4) Application compatibility : Amcache.hve (5) ETC : SCHEMA.DAT

4. HISTORY

Rev	Issue Date	Section	History
0.50	2017-09-27	All	- Draft 1
1.00	2017-12-13	All	- The first release
1.10	2018-05-17	2.1.1.	- Update the Figure 1 with a new type 'Naming convention' - Add comments on the 'Naming convention' type
		2.2.8.	- Add a new type 'Naming convention'
		2.7.	- Add a row, [nr]-08 (Naming convention), into Table 6