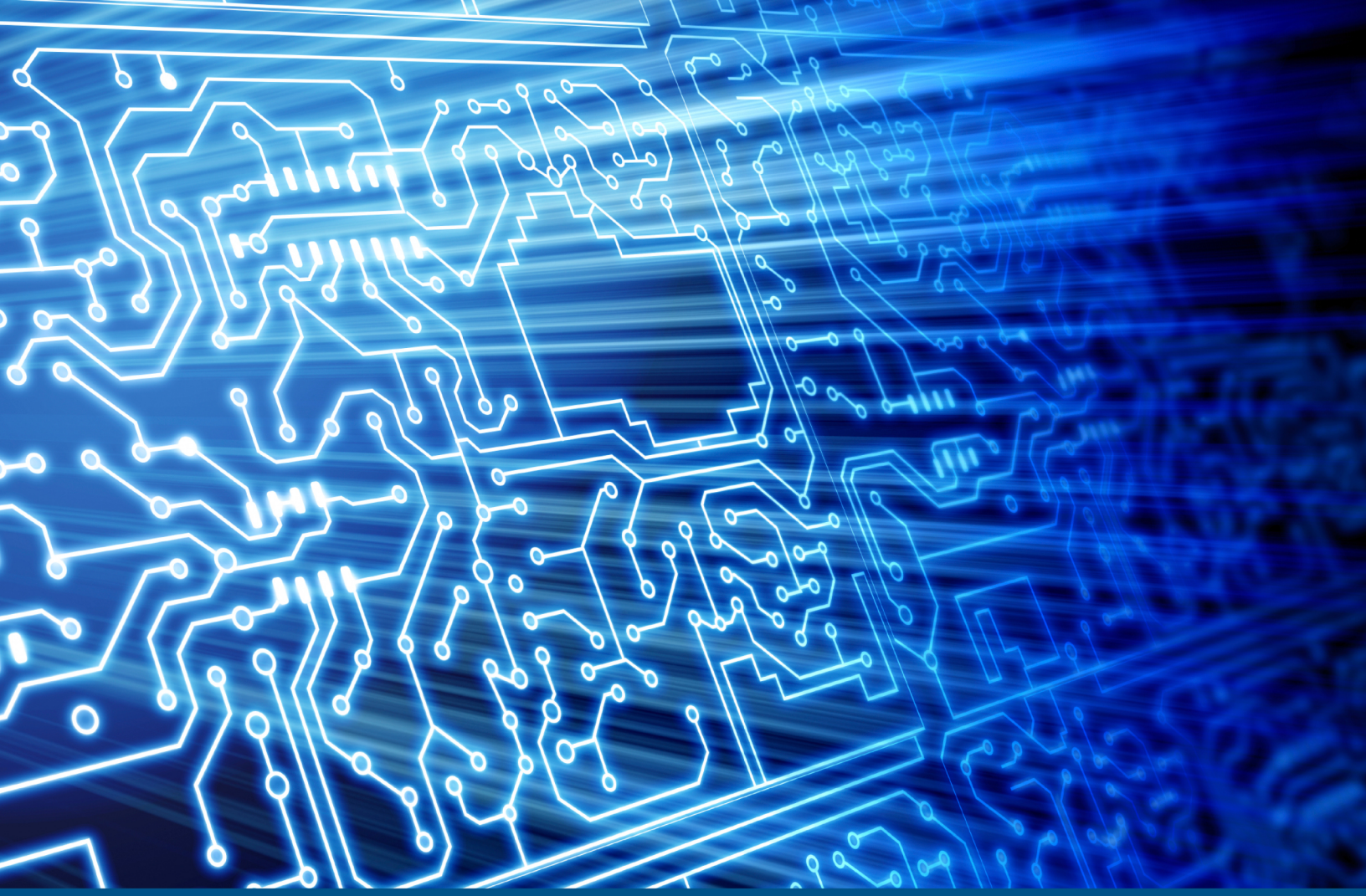


EnCase Forensic 8.07.00.93 (x64)

Test Results for Windows Registry Forensic Tool

April 07, 2019

Homeland Security

Science and Technology

This report was prepared for the Department of Homeland Security (DHS) Science and Technology Directorate (S&T) by the Office of Law Enforcement Standards of the National Institute of Standards and Technology.

For additional information about ongoing DHS S&T cybersecurity projects, please visit https://www.dhs.gov/science-and-technology/cyber-security-division.

April 2019

Test Results for Windows Registry Forensic Tool: EnCase Forensic 8.07.00.93 (x64)

Table of Contents

Introduction

The Computer Forensics Tool Testing (CFTT) program is a joint project of the Department of Homeland Security (DHS), and the National Institute of Standards and Technology Special Program Office (SPO) and Information Technology Laboratory (ITL). CFTT is supported by other organizations, including the Federal Bureau of Investigation, the U.S. Department of Defense Cyber Crime Center, U.S. Internal Revenue Service Criminal Investigation Division Electronic Crimes Program, the National Institute of Justice (NIJ), and the U.S. Department of Homeland Security's Bureau of Immigration and Customs Enforcement, U.S. Customs and Border Protection and U.S. Secret Service. The objective of the CFTT program is to provide measurable assurance to practitioners, researchers, and other applicable users that the tools used in computer forensics investigations provide accurate results. Accomplishing this requires the development of specifications and test methods for computer forensic tools and subsequent testing of specific tools against those specifications.

Test results provide the information necessary for developers to improve tools, users to make informed choices, and the legal community and others to understand the tools' capabilities. The CFTT approach to testing computer forensics tools is based on well recognized methodologies for conformance and quality testing. Interested parties in the computer forensics community can review and comment on the specifications and test methods posted on the CFTT Web site (www.cftt.nist.gov).

This document reports the results from testing EnCase Forensic 8.07.00.93 against a registry dataset that consists of various Windows NT registry hive files. The dataset is available at the CFReDS web site, www.cfreds.nist.gov.

Test results from other tools can be found on the DHS S&T-sponsored digital forensics web page, www.dhs.gov/science-and-technology/nist-cftt-reports.

How to Read This Report

This report is divided into four sections. Section 1 identifies and provides a summary of any significant anomalies observed in the test runs. This section is sufficient for most readers to assess the suitability of the tool for the intended use. Section 2.1 and 2.2 list a testing environment and test data (a registry dataset) prepared for measuring the success of each test. Section 2.3 identifies the test cases that were selected. The test cases are selected, in general, based on features offered by the tool. Section 3 provides an overview of the test case results reported by the tool.

[1] NIST does not endorse nor recommend products or trade names identified in this paper. All products used in this paper are mentioned for use in research and testing by NIST.

Test Results for Windows Registry Forensic Tool

Tool Tested:	EnCase Forensic
Software Version:	8.07.00.93 (x64)
Supplier:	OpenText
Address:	275 Frank Tompa Drive Waterloo, ON N2L 0A1 Canada
Tel:	1-800-499-6544
WWW:	https://www.opentext.com https://www.guidancesoftware.com/encase-forensic

1. Results Summary

Below is a comprehensive summary on how EnCase Forensic 8.07.00.93 performed when processing hive files of a Windows registry dataset. Except for the following anomalies, the tool processed and extracted all supported data completely and accurately for all registry hive files tested.

Results on Core Features:

- The tool was terminated without any notification when it processed a tree structure with a large number of levels (about 1 million) in an experimental hive file.
- Long value names (16,383 bytes and more) were not reported.
- The tool did not report UTF-16LE characters properly.
- The tool did not identify unusual ASCII characters (between 0x04 and 0x0D) of key and value names.
- The 'Tree' and 'Table' panes of the tool operated differently when showing ASCII and UTF-16LE characters.

Results on Recovering Deleted Registry Objects:

- The tool reported deleted keys but failed to recover deleted values from several reference hive files.
- Partial deleted keys were recovered from several reference hive files.
- Deleted values were not recovered from several reference hive files.

Results on Extracting Windows Forensic Artifacts:

- Application
  - Partial application related data was reported. [Windows: ALL]
    - The tool did not identify names of several applications.
    - 32-bit (x86) application related data in 64-bit Windows systems was not reported. [Windows: 10, 10RS1]

- Auto Run
  - Partial auto run related data was reported. [Windows: ALL]
    - The tool listed auto run related data in the SOFTWARE hive, but those data in NTUSER.DAT hives was not reported.

- External Device
  - Partial external device related data was reported. [Windows: ALL]
    - The tool identified all USB storage devices, but it did not report several device related metadata such as 'Last Connected Date'.

- Recently Opened File and Directory
  - Partial opened file and directory related data was reported. [Windows: Vista, 7, 8, 8.1]
    - The tool identified recent file and directory entries stored in a user account (IEUser), but it did not report those data from other user accounts such as 'CFTT'.
  - Opened file and directory related data was not reported. [Windows: 10, 10RS1]

- Others
  - The tool identified additional artifacts on the network drive connection, but it returned some invalid paring results. [Windows: 8, 8.1, 10, 10RS1]

For more test results, see section 3.

2. Test Environment and Selected Cases

This section describes the test execution environment, test dataset and test cases.

2.1. Execution Environment

EnCase Forensic 8.07.00.93 was installed on Windows 7 Enterprise v6.1.7601 (x64, English).

2.2. Test Dataset

The tool was measured by analyzing interpreted and extracted data from various registry hive files developed as a reference dataset. Table 1, Table 2 and Table 3 list data codes that are linked to registry files for testing core features and an optional feature relating to recovering deleted registry objects. In addition, well-known registry hive files from reference Windows systems with ground truth data were used to test an optional feature on extracting Windows registry forensic artifacts. In that regard, Table 4 defines several artifact groups considered for populating the reference Windows systems (Vista, 7, 8, 8.1, 10 and 10RS1) to limit the scope of tool testing.

It should be noted that because the tool supports to extract registry artifacts from full disk images, disk image files exported from the reference Windows systems were used for this tool testing. More specifically, the last volume shadow copy (VSC) of each disk image having a Windows system was processed by the tool, because the last VSC includes the greatest number of artifacts before performing anti-forensic activities to remove usage history. For more information, the dataset and related documents can be obtained from: www.cfreds.nist.gov.

Table 1. Dataset for Testing Core Features

Category	Code	Description	Generation method
Normal Registry Hive File	NR-01-1	Possible data types	[Windows] API (.REG)
	NR-01-2	Possible data types	[Linux] *hivex* library
	NR-02-1	Simple tree structure	[Windows] API (.REG)
	NR-02-2	Simple tree structure	[Linux] *hivex* library
	NR-03-1	Tree structure with the maximum levels (512)	[Windows] API (.REG)
	NR-03-2	Tree structure with the maximum levels (512)	[Linux] *hivex* library
	NR-03-3	Tree structure with abnormal levels (1 million)	[Linux] *hivex* library
	NR-04-1	Maximum key name length (255 and 256)	[Windows] API (.REG)
	NR-04-2	Maximum key name length (255 and 256)	[Linux] *hivex* library
	NR-04-3	Maximum key name length beyond limitation	[Linux] *hivex* library
	NR-05-1	Maximum value name length (16,383)	[Windows] API (.REG)
	NR-05-2	Maximum value name length (16,383)	[Linux] *hivex* library
	NR-05-3	Maximum value name length beyond limitation	[Linux] *hivex* library
	NR-06-1	Big data (16,344 or more)	[Windows] API (.REG)

Category	Code	Description	Generation method
	NR-06-2	Big data (16,344 or more)	[Linux] *hivex* library
	NR-07-1	Non-ASCII characters	[Windows] API (.REG)
	NR-07-2	Non-ASCII characters	[Linux] *hivex* library
	NR-08	Naming convention	[Windows] API (Python)
Corrupted Registry Hive File	CR-01	A hive bin with Root key	Python script
	CR-02	A hive bin	Python script
	CR-03	Last half	Python script
	CR-04	Multiple fragments with hbin header	Python script
	CR-05	Base block	Python script

Table 2. Dataset for Testing Core Features (continue)

Category	Code	Description	Manipulation point
Manipulated Registry Hive File	MR-01-1	Hide a root key (invalid checksum)	'root cell offset' in the base block
	MR-01-2	Hide a root key (valid checksum)	'root cell offset' in the base block
	MR-02-1	Hide key names	'key name size' in the key (nk) cell
	MR-02-2	Hide key names	'key cell size' in the key (nk) cell
	MR-03-1	Hide subkeys of a key	'number of subkeys' in the key (nk) cell
	MR-03-2	Hide subkeys of a key	'subkey-list cell size' in the key (nk) cell
	MR-03-3	Hide subkeys of a key	'number of subkeys' in the subkey-list cell
	MR-03-4	Hide subkeys of a key	'subkey offset' items in the subkey-list cell
	MR-04-1	Hide values of a key	'number of values' in the key (nk) cell
	MR-04-2	Hide values of a key	'value-list cell size' in the value-list cell
	MR-04-3	Hide values of a key	'value offset' items in the value-list cell
	MR-05-1	Hide value names	'value name size' in the value (vk) cell
	MR-05-2	Hide value names	'value cell size' in the value (vk) cell
	MR-06-1	Hide data of a value	'data size' in the value (vk) cell
	MR-06-2	Hide data of a value	'data cell size' in the data cell
	MR-06-3	Hide data of a value	'data offset' in the value (vk) cell
	MR-06-4	Hide data of a value	'data type' in the value (vk) cell
	MR-07	Hide big data of a value	'data size' in the value (vk) cell
	MR-08	Infinite key loop	'subkey offset' in the subkey-list cell
	MR-09	Invalid integer data size	'data size' in the value (vk) cell
	MR-10	Invalid binary data size	'data size' in the value (vk) cell
	MR-11	Invalid string data size	'data size' in the value (vk) cell
	MR-12	Version mismatch (big data processing)	'minor version value' in the base block
	MR-13	Ambiguous key name	'encoding flag' in the key (nk) cell
	MR-14	Ambiguous value name	'encoding flag' in the value (vk) cell
	MR-15	Ambiguous encodings	text encoded by various encoding standards

Table 3. Dataset for Testing an Optional Feature: Recovering Deleted Registry Objects

Category	Code	Description	Generation method
Normal Registry Hive File with Deleted Registry Data	NRD-01-1	Delete keys with values, but without subkeys	[Windows] API (.REG)
	NRD-01-2	Delete keys with values, but without subkeys	[Linux] *hivex* library
	NRD-02-1	Delete a key with values and subkeys	[Windows] API (.REG)
	NRD-02-2	Delete a key with values and subkeys	[Linux] *hivex* library
	NRD-03-1	Delete a key without values and subkeys	[Windows] API (.REG)
	NRD-03-2	Delete a key without values and subkeys	[Linux] *hivex* library
	NRD-04	Delete a value with normal data	[Windows] API (.REG)
	NRD-05	Delete a value with big data	[Windows] API (.REG)
	NRD-06	Delete multiple values in a key	[Windows] API (.REG)

Table 4. Artifacts considered for Testing an Optional Feature: Extracting Forensic Artifacts

Windows	Artifact group	Details (**D**: description, **C**: check points, **R**: related paths) * The paths (R) show a few examples although there may exist other paths. Note that 'Wow6432Node' key should be also considered on 64-bit Windows systems.	
Vista+ The '+' symbol signifies later versions.	Account	D	Accounts
		C	Name, type, login count, timestamps (login, pw reset, failed), etc.
		R	SAM\SAM\Domains\Account\Users\ SAM\SAM\Domains\Builtin\Aliases\ SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList\ SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\
	Application	D	Installed programs
		C	Name, vendor, version, installed path, timestamp, etc.
		R	SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths\ SOFTWARE\Microsoft\Windows\CurrentVersion\Installer\UserData\?SID?\Products\ SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\ SOFTWARE\Classes\Installer\Products\ USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppModel\Repository\
	Application Experience & Compatibility (Shimcache)	D	Windows Application Compatibility related data
		C	File name, file size, timestamp, etc.
		R	SYSTEM\?ControlSet?\Control\Session Manager\AppCompatCache\
	Auto Run	D	Programs that start automatically when a user logs on
		C	Name, executable path, timestamp, etc.
		R	NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Run\ NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\RunOnce\ SOFTWARE\Microsoft\Windows\CurrentVersion\Run\ SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce\
	Dialog Usage	D	Dialog box related user actions
		C	Name, timestamps, etc.
		R	NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRU\ NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidlMRU\
	External Device	D	External devices (like USB storages) attached into the system
		C	Vendor, product, serial number, connected date, drive letter, etc.
		R	SYSTEM\MountedDevices\ SYSTEM\?ControlSet?\Control\DeviceClasses\ SYSTEM\?ControlSet?\Control\DeviceContainers\ SYSTEM\?ControlSet?\Enum\ SOFTWARE\Microsoft\WindowsNT\CurrentVersion\EMDMgmt\ SOFTWARE\Microsoft\Windows Portable Devices\Devices\ NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\

Windows	Artifact group	Details (**D**: description, **C**: check points, **R**: related paths) * The paths (R) show a few examples although there may exist other paths. Note that 'Wow6432Node' key should be also considered on 64-bit Windows systems.	
Vista+, continued	Network Connection	D	Configurations of interface cards and network connection history
		C	Name, IP, gateway, MAC, SSID, DNS, etc.
		R	SYSTEM\?ControlSet?\Services\Tcpip\Parameters\Interfaces\ SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkCards\ SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\ SOFTWARE\Microsoft\WZCSVC\Parameters\Interfaces\
	Network Drive	D	Network connection history to external systems
		C	Name, IP, account drive letter, type, timestamp, etc.
		R	NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\Map Network Drive MRU\ NTUSER.DAT\Network\
	OS Information	D	Installed OS (Windows) information
		C	Version, install date, computer name, owner, shutdown time, etc.
		R	SOFTWARE\Microsoft\Windows NT\CurrentVersion\ SYSTEM\?ControlSet?\Control\Windows\ SYSTEM\?ControlSet?\Control\ComputerName\
	Recently Opened File and Directory	D	Recently opened files and directories
		C	Name, timestamp, etc.
		R	NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\ NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Applets\?APP_NAME?\Recent File List\ NTUSER.DAT\Software\Microsoft\MediaPlayer\Player\RecentFileList\ NTUSER.DAT\Software\Microsoft\Office\?VERSION?\?APP_NAME?\User MRU\ NTUSER.DAT\Software\Adobe\Acrobat Reader\?VERSION?\AVGeneral\cRecentFiles\ NTUSER.DAT\Software\Adobe\Acrobat Reader\?VERSION?\AVGeneral\cRecentFolders\
	Remote Desktop	D	Network connection history to external systems
		C	IP, account ID, timestamp, etc.
		R	NTUSER.DAT\Software\Microsoft\Terminal Server Client\Default\ NTUSER.DAT\Software\Microsoft\Terminal Server Client\Servers\?IP?\
	Run Command History	D	Recently used commands from Windows Run
		C	Command, timestamp, etc.
		R	NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU\
	Service and Driver	D	Service and driver list
		C	Display name, description, type, start, image path, etc.
		R	SYSTEM\?ControlSet?\Services\?NAME?\
	Shared Directory	D	Shared directory list
		C	Name, directory path, type, timestamp, etc.
		R	SYSTEM\?ControlSet?\Services\LanmanServer\Shares\
	ShellBag	D	Directories or files accessed by each user account (Database to track user's window viewing preferences)
		C	Directory or file path, timestamp, etc.
		R	NTUSER.DAT\Software\Microsoft\Windows\Shell\Bags\ NTUSER.DAT\Software\Microsoft\Windows\Shell\BagMRU\ NTUSER.DAT\Software\Microsoft\Windows\ShellNoRoam\Bags\ NTUSER.DAT\Software\Microsoft\Windows\ShellNoRoam\BagMRU\ USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\Shell\Bags\ USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\ USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\ShellNoRoam\Bags\ USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\ShellNoRoam\BagMRU\

Windows	Artifact group	Details (**D**: description, **C**: check points, **R**: related paths) * The paths (R) show a few examples although there may exist other paths. Note that 'Wow6432Node' key should be also considered on 64-bit Windows systems.	
Vista+, continued	Timezone	D	Timezone information
		C	Timezone name, time offset, etc.
		R	SYSTEM\?ControlSet?\Control\TimeZoneInformation\
	UserAssist	D	Programs executed by each user account (executable and link files)
		C	Account, file name, run count, timestamp, etc.
		R	NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\
Win 7 and Win 8	Search	D	Search history using Windows Search feature
		C	Search keyword, timestamp, etc.
		R	Win 7: NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\WordWheelQuery\ Win 8: NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\SearchHistory\Microsoft.Windows.FileSearchApp\ (Vista, 8.1 and 10 does not save search keywords into the registry.)
Win 7+	Application Experience & Compatibility (Amcache)	D	Windows Application Compatibility related data
		C	App name, executable path, hash value, timestamp, etc.
		R	Amcache.hve\Root\File\?VOLUME_GUID?\ Amcache.hve\Root\Programs\?PROGRAM_ID?\

2.3. Test Case Selection

EnCase Forensic 8.07.00.93 as an integrated multi-functional forensic toolkit does support built-in features to process the Windows registry hive format. The tool provides a registry hive file handler through an internal feature known as 'View File Structure'. In addition, the 'Evidence Processor' feature of the tool also provides several options for extracting various forensic artifacts from well-known registry hive files. Therefore, the selected test cases are:

- Core features
  - Processing normal registry hive files
  - Processing corrupted registry hive files
  - Processing manipulated registry hive files
- Optional features
  - Recovering deleted registry objects
  - Extracting Windows registry forensic artifacts

The following tables give a brief description of available test cases in the dataset. Not all test cases are used for this tool testing.

Table 5. Test Cases for Testing Core Features

Test case ID	Case description
WRT-TC-NR-01-1	◦ Process a primary file containing values with various data types (total 12) ◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NR-01-2	
WRT-TC-NR-02-1	◦ Process a primary file containing a simple tree structure ◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NR-02-2	
WRT-TC-NR-03-1	◦ Process a primary file containing an experimental tree structure that is 512 or more levels deep
WRT-TC-NR-03-2	

Test case ID	Case description
WRT-TC-NR-03-3	◦ Note that 3 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NR-04-1	◦ Process a primary file containing keys with long names (255 or more bytes)
WRT-TC-NR-04-2	◦ Note that 3 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NR-04-3	
WRT-TC-NR-05-1	◦ Process a primary file containing values with long names (16,383 or more bytes)
WRT-TC-NR-05-2	
WRT-TC-NR-05-3	◦ Note that 3 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NR-06-1	◦ Process a primary file containing values with big data (> 16,344 bytes)
WRT-TC-NR-06-2	◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NR-07-1	◦ Process a primary file containing keys and values with non-ASCII characters
WRT-TC-NR-07-2	◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NR-08	◦ Process a primary file containing keys and values with unusual (but valid) names
WRT-TC-CR-01	◦ Process a corrupted primary file that contains a wiped hive bin (having root key)
WRT-TC-CR-02	◦ Process a corrupted primary file that contains a wiped hive bin (randomly selected)
WRT-TC-CR-03	◦ Process a corrupted primary file that contains wiped hive bins (last half)
WRT-TC-CR-04	◦ Process a corrupted primary file that contains wiped multiple blocks (randomly selected among blocks having the hbin header structure)
WRT-TC-CR-05	◦ Process a corrupted primary file that contains a wiped base block
WRT-TC-MR-01-1	◦ Process a manipulated primary file that contains hidden keys
WRT-TC-MR-01-2	◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-MR-02-1	◦ Process a manipulated primary file that contains hidden key names
WRT-TC-MR-02-2	◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-MR-03-1	◦ Process a manipulated primary file that contains hidden subkeys
WRT-TC-MR-03-2	◦ Note that 4 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-MR-03-3	
WRT-TC-MR-03-4	
WRT-TC-MR-04-1	◦ Process a manipulated primary file that contains hidden values
WRT-TC-MR-04-2	◦ Note that 3 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-MR-04-3	
WRT-TC-MR-05-1	◦ Process a manipulated primary file that contains hidden value names
WRT-TC-MR-05-2	◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-MR-06-1	◦ Process a manipulated primary file that contains hidden data
WRT-TC-MR-06-2	◦ Note that 4 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-MR-06-3	
WRT-TC-MR-06-4	
WRT-TC-MR-07	◦ Process a manipulated primary file that contains hidden big data
WRT-TC-MR-08	◦ Process a manipulated primary file that contains an infinite key loop
WRT-TC-MR-09	◦ Process a manipulated primary file that contains an invalid integer data size
WRT-TC-MR-10	◦ Process a manipulated primary file that contains an invalid binary data size
WRT-TC-MR-11	◦ Process a manipulated primary file that contains an invalid string data size
WRT-TC-MR-12	◦ Process a manipulated primary file that contains a mismatched version indicator (focusing on big data processing)
WRT-TC-MR-13	◦ Process a manipulated primary file that contains a mismatched key name encoding flag

Test case ID	Case description
WRT-TC-MR-14	◦ Process a manipulated primary file that contains a mismatched value name encoding flag
WRT-TC-MR-15	◦ Process a manipulated primary file that contains key names, value names and data encoded by unsupported encoding standards

Table 6. Test Cases for Testing Optional Features: Recovering Deleted Registry

Test case ID	Case description
WRT-TC-NRD-01-1	◦ Process a primary file that contains deleted keys with values but without subkeys ◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NRD-01-2	
WRT-TC-NRD-02-1	◦ Process a primary file that contains a deleted key with values and subkeys ◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NRD-02-2	
WRT-TC-NRD-03-1	◦ Process a primary file that contains a deleted key without values and subkeys ◦ Note that 2 subcases were defined based on the dataset (see Section 2.2)
WRT-TC-NRD-03-2	
WRT-TC-NRD-04	◦ Process a primary file that contains a deleted value with data
WRT-TC-NRD-05	◦ Process a primary file that contains a deleted value with big data
WRT-TC-NRD-06	◦ Process a primary file that contains deleted multiple values in a key

Table 7. Test Cases for Testing Optional Features: Extracting Forensic Artifacts

Test case ID	Case description
WRT-TC-FA-01	◦ Process primary files containing Account related data
WRT-TC-FA-02	◦ Process primary files containing Application related data
WRT-TC-FA-03	◦ Process primary files containing Application Compatibility (Amcache) data
WRT-TC-FA-04	◦ Process primary files containing Application Compatibility (Shimcache) data
WRT-TC-FA-05	◦ Process primary files containing Auto Run related data
WRT-TC-FA-06	◦ Process primary files containing Dialog Usage related data
WRT-TC-FA-07	◦ Process primary files containing External Device related data
WRT-TC-FA-08	◦ Process primary files containing Network Connection related data
WRT-TC-FA-09	◦ Process primary files containing Network Drive related data
WRT-TC-FA-10	◦ Process primary files containing OS Information related data
WRT-TC-FA-11	◦ Process primary files containing Recently Opened File and Directory related data
WRT-TC-FA-12	◦ Process primary files containing Remote Desktop related data
WRT-TC-FA-13	◦ Process primary files containing Run Command History related data
WRT-TC-FA-14	◦ Process primary files containing Search related data
WRT-TC-FA-15	◦ Process primary files containing Service and Driver related data
WRT-TC-FA-16	◦ Process primary files containing Shared Directory related data
WRT-TC-FA-17	◦ Process primary files containing ShellBag related data
WRT-TC-FA-18	◦ Process primary files containing Timezone related data
WRT-TC-FA-19	◦ Process primary files containing UserAssist related data

NOTES:

- Some test cases are for specific tool features.
- The 'WRT-TC' prefix (that means Windows Registry Tool – Test Case) will be omitted for simplicity in the remainder of this report.

3. Test Results

This section provides the test results reported by EnCase Forensic 8.07.00.93. The results are as follows:

AS Expected	The Windows registry forensic tool returned expected test results – the tool processed and reported data from registry hive files successfully.
Partial	The Windows registry forensic tool returned some of data from registry hive files.
Observed	The Windows registry forensic tool returned some of data from 'corrupted' or 'manipulated' registry hive files without any error or crash. This type of test results is not subject to selection between 'As Expected' and 'Partial'. Instead, each result describes detailed behaviors when processing abnormal (Or experimental) hive files. *Note: The observed results may be useful to understand and improve each tool's registry processing algorithms, which include format validation, data interpretation, anomaly detection and exception handling.*
Not As Expected	The Windows registry forensic tool failed to return expected test results – the tool did not process or report supported data from registry hive files properly.
Not Applicable (NA)	The Windows registry forensic tool does not provide support for a particular test case.

3.1. Results on Core Features

The reference registry hive files were processed with EnCase Forensic 8.07.00.93.

All test cases were successful except for the following.

- The tool was terminated without any notification when it processed a tree structure with a large number of levels (about 1 million) of an experimental hive file NR-03-3.
- Long value names (16,383 bytes and more) were not reported. (NR-05-1, NR-05-2 and NR-05-3)
- The tool did not report UTF-16LE characters properly. (NR-07-1 and NR-07-2)
- The tool did not identify unusual ASCII characters (between 0x04 and 0x0D) of key and value names. (NR-08)
- The 'Tree' and 'Table' panes of the tool operated differently when showing ASCII and UTF-16LE characters. (NR-08)

NOTES:

- Detailed observation results when the tool processed the 'corrupted' and 'manipulated' hive files are available in Table 8.

See Table 8 below for more details.

Table 8. Test Results on Core Features

EnCase Forensic 8.07.00.93		
Test case	**Test result**	**Note**
NR-01-1	*As Expected*	-
NR-01-2	*As Expected*	-
NR-02-1	*As Expected*	-
NR-02-2	*As Expected*	-
NR-03-1	*As Expected*	-
NR-03-2	*As Expected*	-
NR-03-3	***Not As Expected***	The tool was terminated without any notification. Note that this reference file is not a normal but experimental hive file.
NR-04-1	*As Expected*	-
NR-04-2	*As Expected*	-
NR-04-3	*As Expected*	-
NR-05-1	***Not As Expected***	Long value names were shown as '<_??__>'.
NR-05-2	***Not As Expected***	Long value names were shown as '<_??__>'.
NR-05-3	***Not As Expected***	Long value names were shown as '<_??__>'.
NR-06-1	*As Expected*	-
NR-06-2	*As Expected*	-

EnCase Forensic 8.07.00.93		
Test case	**Test result**	**Note**
NR-07-1	***Partial***	The tool failed to report UTF-16LE characters of key names in the 'Tree' pane. Note that UTF-16LE characters in key and value names were reported properly in the 'Table' pane.
NR-07-2	***Partial***	
NR-08	***Partial***	Note that the following findings were the results compared with outputs from Windows (RegEdit.exe). The 'Tree' and 'Table' panes operated differently when showing ASCII characters between 0x01 and 0x1B. In addition, the tool did not process ASCII characters between 0x04 and 0x0D. The tool probably ignored them unlike operations of RegEdit.exe.

EnCase Forensic 8.07.00.93		
Test case	**Test result**	**Note**
NR-08	***Partial***	The tool processed Extended ASCII characters of key and value names using Windows-1252 (CP-1252). The ASCII 0x80 was printed as '€'. Note that RegEdit.exe showed those characters using ISO/IEC 8859-1, so characters between 0x80 and 0x9F were ignored. This finding is not abnormal but shows a difference from RegEdit.exe. The 'Tree' and 'Table' panes operated differently when reporting UTF-16LE characters. For example, a selected key has the name: 'UTF-16LE_☆_0x03F8_(þ)'.
CR-01	*Observed*	The tool didn't return any result.
CR-02	*Observed*	The tool processed (probably ignored) corrupted registry objects without error, and it reported recoverable (accessible) keys and values. However, there were no notifications about abnormal data.
CR-03	*Observed*	
CR-04	*Observed*	
CR-05	*NA*	The tool does not support this kind of corruption.

EnCase Forensic 8.07.00.93		
Test case	**Test result**	**Note**
MR-01-1	*Observed*	The tool identified a key (the 1st subkey of the original root key) due to the edited 'root cell offset' in a base block. Although many allocated cells still exist, the tool didn't identify them. In addition, the tool does not validate a checksum value in the base block.
MR-01-2	*Observed*	The tool identified a key (the 1st subkey of the original root key) due to the edited 'root cell offset' in a base block. Although many allocated cells still exist, the tool didn't identify them.
MR-02-1	*Observed*	The tool identified a partial key name due to the edited 'key name size' in a key (nk) cell.
MR-02-2	*Observed*	The tool identified a valid key name (`0x01_TYPE1_DATA-TYPES`), even with the edited 'key cell size' in a key (nk) cell. That is, the tool does not consider 'cell size' as a key factor for parsing.
MR-03-1	*Observed*	The tool identified all valid keys (7 of 7), even with the adjusted 'number of subkeys' in a key (nk) cell. That is, the tool does not consider 'number of subkeys' in the key cell as a key factor for parsing.
MR-03-2	*Observed*	The tool identified all valid keys (7 of 7), even with the edited 'subkey-list cell size' in a subkey-list cell. That is, the tool does not consider 'cell size' as a key factor for parsing.
MR-03-3	*Observed*	The tool identified partial keys (3 of 7) due to the adjusted 'number of subkeys' in a subkey-list cell.
MR-03-4	*Observed*	The tool identified partial keys (3 of 7) due to zeroized 'subkey offset' items in a subkey-list cell.
MR-04-1	*Observed*	The tool identified partial values (8 of 12) due to the adjusted 'number of values' in a key (nk) cell.
MR-04-2	*Observed*	The tool identified all valid values (12 of 12), even with the edited 'value-list cell size' in a value-list cell. That is, the tool does not consider 'cell size' as a key factor for parsing.
MR-04-3	*Observed*	The tool identified partial values (8 of 12) due to zeroized 'value offset' items in a value-list cell.
MR-05-1	*Observed*	The tool identified a partial value name (`VALUE 0x0`) due to the edited 'value name size' (a half of the original size) in a value (vk) cell.
MR-05-2	*Observed*	The tool identified a full value name (`VALUE 0x00 (NONE)`), even with the edited 'value cell size' in a value (vk) cell. That is, the tool does not consider 'cell size' as a key factor for parsing.
MR-06-1	*Observed*	The tool didn't identify a data stream due to the NULL 'data size' in a value (vk) cell.
MR-06-2	*Observed*	The tool identified a valid data stream, even with the edited 'data cell size' in a data cell. That is, the tool does not consider 'cell size' as a key factor for parsing.
MR-06-3	*Observed*	The tool identified a wrong data stream due to the NULL 'data offset' in a value (vk) cell. Having the NULL offset is abnormal, but there was no notification from the tool.
MR-06-4	*Observed*	The tool identified a BINARY value as a string (SZ) due to the edited 'data type' in a value (vk) cell.

EnCase Forensic 8.07.00.93		
Test case	**Test result**	**Note**
MR-07	*Observed*	The tool didn't identify a big-data stream due to the NULL 'data size' in a value (vk) cell.
MR-08	*Observed*	The tool processed a key loop without error. That is, it allowed users to access keys within a closed loop until they reached a limited level (about 136 levels in this test case). Infinite loops are abnormal, but there was no notification from the tool.
MR-09	*Observed*	The tool processed invalid sizes of data without error. The 'data size' item beyond a cell's boundary is abnormal, but there was no notification from the tool.
MR-10	*Observed*	
MR-11	*Observed*	
MR-12	*Observed*	When processing a file patched from v1.3 to v1.5 format, the tool didn't return any results. In the case of a file patched from v1.5 to v1.3, partial data were identified due to ignoring big-data cells (> 16,344) of v1.5. That is, the tool considers the version indicator in the base block as a key factor for parsing.
MR-13	*Observed*	The tool returned printable ASCII values due to the edited 'encoding flag' in a key (nk) cell.
MR-14	*Observed*	The tool returned printable ASCII values due to the edited 'encoding flag' in a value (vk) cell.
MR-15	*Observed*	The tool tried to decode key/value names and data as UTF-16LE, so names encoded by other encoding standards could not be identified properly.

3.2. Results on Recovering Deleted Registry Objects

The reference registry hive files were processed with EnCase Forensic 8.07.00.93.

All test cases were successful except for the following.

- The tool reported deleted keys but failed to recover deleted (but complete) values from NRD-01-2, NRD-02-1 and NRD-02-2.
- Partial deleted (but complete) keys were recovered from NRD-02-1 and NRD-02-2.
- Deleted (but complete) values were not recovered from NRD-04, NRD-05 and NRD-06.

NOTES:

- All reference hive files were opened through the 'View File Structure' function along with the 'Find deleted content' option.

- In all test cases, the tool unnecessarily reported 'non-deleted' keys as 'deleted' keys. For example, in NRD-01-1, '0x07_TYPE1_NON-ASCII' is a non-deleted key, but the tool reported a duplicated key with the 'deleted' property.

See Table 9 below for more details.

Table 9. Test Results on Recovering Deleted Registry Objects

EnCase Forensic 8.07.00.93		
Test case	**Test result**	**Note**
NRD-01-1	*As Expected*	-
NRD-01-2	***Partial***	The tool returned deleted keys (`0x01_TYPE2_DATA-TYPES`, `0x06_TYPE2_BIG-DATA`), but it failed to recover deleted values. Note that it was not possible to link deleted values and their keys in this test case. It is because when deleting a key using *hivex* library, the 'value-list offset' field in its key cell structure was initialized.

EnCase Forensic 8.07.00.93		
Test case	**Test result**	**Note**
NRD-02-1	***Partial***	The tool failed to recover a deleted key (`0x07_TYPE1_NON-ASCII`) and its sub-entries.
NRD-02-2	***Partial***	The tool returned deleted keys (`Node_1, Node_2`), but it failed to identify their parent key (`0x02_TYPE2_TREE`) and recover deleted values. In addition, the tool did not identify a deleted key (`0x07_TYPE2_NON-ASCII`) and its sub-entries. Note that it was not possible to link deleted values and their keys in this test case. It is because when deleting a key using *hivex* library, the 'value-list offset' field in its key cell structure was initialized.
NRD-03-1	*As Expected*	-
NRD-03-2	*As Expected*	-
NRD-04	***Not As Expected***	The tool failed to recover deleted values.
NRD-05	***Not As Expected***	The tool failed to recover deleted values.
NRD-06	***Not As Expected***	The tool failed to recover deleted values.

3.3. Results on Extracting Windows Forensic Artifacts

The reference registry hive files were processed with EnCase Forensic 8.07.00.93[2].

DETAILS:

- This optional feature was not measured strictly by a detailed criterion. In the light of the fact that forensic tools may produce different forms of results although they process identical data to extract Windows registry forensic artifacts, this test was performed through comparative analysis with ground truth data and registry-related knowledge by CFTT staff.

- To perform this test, disk images were added as 'Evidence' of the tool, and then they were processed by the 'Evidence Processor' of EnCase.

- The following 'Evidence Processor' options were enabled for this tool testing:
  - Modules – System Info Parser
    - Check all items of 'Artifact Collection Options'
    - Check all items of 'Advanced Registry Commands'
  - Modules – Windows Artifact Parser
    - Check the 'ShellBags' option

- The registry-related results produced through the 'Evidence Processor' were identified in the 'Artifacts' view of the tool. This work also considered those results organized by the 'Case Analyzer', which is a default EnScript utilized for reporting outputs of the 'Evidence Processor'. The following table shows specific positions (paths) of outputs connected with each test case.

Test case	Related result lists in the 'Artifacts' view	Related report categories in the 'Case Analyzer'
FA-01	◦ System Info Parser – Accounts – User Accounts	◦ Users – Registry
FA-02	◦ System Info Parser – Software – Installed Applications ◦ System Info Parser – Software – Installed MS Applications ◦ System Info Parser – Software – Uninstalled Applications	◦ Installed Apps ◦ Installed MS Apps ◦ Uninstalled Apps
FA-03	-	-
FA-04	-	-
FA-05	◦ System Info Parser – Custom Keys – Auto Start	-
FA-06	-	-
FA-07	◦ System Info Parser – USB Devices – Devices ◦ System Info Parser – Hardware – Devices ◦ System Info Parser – Custom Keys – Hardware	◦ USB Devices ◦ Hardware Devices
FA-08	◦ System Info Parser – Network – Interfaces	◦ IP Gateway Pairs ◦ Network Interfaces – Registry
FA-09	-	-
FA-10	◦ System Info Parser – Operating System – System Artifacts	◦ System Info
FA-11	◦ System Info Parser – MRU Artifacts – Explorer Recent Documents ◦ System Info Parser – MRU Artifacts – WordPad	◦ Recent Files ◦ Explorer Typed Folders
FA-12	-	-

[2] The option for displaying times was configured as 'Eastern Time'.

Test case	Related result lists in the 'Artifacts' view	Related report categories in the 'Case Analyzer'
FA-13	-	-
FA-14	-	-
FA-15	◦ System Info Parser – Operating System – Service Artifacts	◦ OS Services
FA-16	◦ System Info Parser – Shared Devices – Devices	◦ Mapped Shared
FA-17	◦ Windows Artifact Parser – ShellBags Parser	-
FA-18	◦ System Info Parser – Operating System – Time Zone	◦ Time Zone
FA-19	-	-

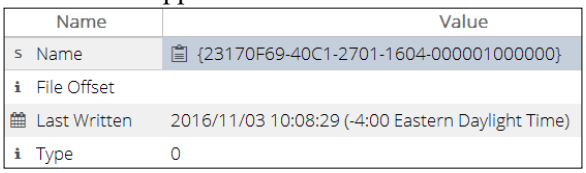
- Although there may exist a variety of freely (or commercially) available EnScripts[3] for expanding the tool's capability, this work only considered default functions provided by the tool.

[3] For instance, the vendor operates 'EnCase App Central' page (https://www.guidancesoftware.com/app) to promote distribution and sharing of EnScripts.

All test cases were successful except for the following.

- Account
  - All registered accounts were reported as expected, but the following findings need to be taken into consideration. [Windows: ALL[4]]
    - The tool did not report several well-known account related entries such as 'Last password reset date', 'Last failed login date' and 'Login count'.
    - The tool did not report several Microsoft Live (Internet) account related entries such as 'Internet user name', 'Given name' and 'Surname'. [Windows: 8, 8.1, 10, 10RS1]

- Application
  - Partial application related data was reported. [Windows: ALL]
    - The tool identified legacy desktop applications and associated timestamps from `Uninstall` and `App Paths` keys as listed in Table 4. However, it did not identify names of several applications appropriately. That is, it just displayed key names instead of getting the 'DisplayName' value of each `Uninstall` key as shown in the below example. Note that results organized by the 'Case Analyzer' properly included well-known values of each `Uninstall` key such as 'DisplayName', 'Publisher', 'InstallLocation' and 'InstallSource', but the results did not include any associated timestamps.

Description	Related screenshots (Windows 7 as an example)
• The tool did not identify names of several applications. • The tool reported GUIDs instead of application names.	• An example item of the 'System Info Parser – Software – Uninstalled Applications' list in the 'Artifacts' view:  • The following shows manual verification results: ◦ `SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{23170F69-40C1-2701-1604-000001000000}\` → 'DisplayName' value: 7-Zip 16.04

    - 32-bit (x86) application related data in 64-bit Windows systems was not reported. [Windows: 10, 10RS1]
  - The tool did not support to extract Windows Store app[5] related data.

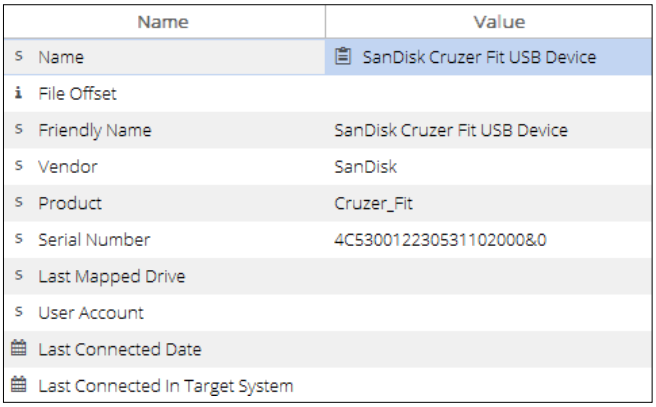
- Auto Run
  - Partial auto run related data was reported. [Windows: ALL]
    - The tool listed auto run related data in the SOFTWARE hive, but those data in NTUSER.DAT hives were not reported.
    - It should be noted here that the tool did not support interpreting (or normalizing) auto run related data. Instead, the tool simply provided bookmarks to pre-defined

[4] 'Windows: ALL' includes Vista, 7, 8, 8.1, 10 and 10RS1. (refer to Section 2.2)

[5] Artifacts on installed app packages can be identified from 'StateRepository-Machine.srd' SQLite file (in Windows 10) and 'PackageRepository.edb' ESE file (until Windows 8.1), as well as USRCLASS hives as shown in Table 4.

registry key paths. The bookmark list could be accessed through the 'System Info Parser – Custom Keys – Auto Start' in the 'Artifacts' view.

- External Device
  - Partial external device related data was reported. [Windows: ALL]
    - The tool identified all USB storage devices, but it did not report several device related metadata such as 'Last Connected Date' as shown in the below example.

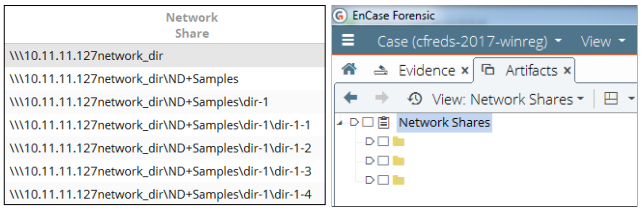
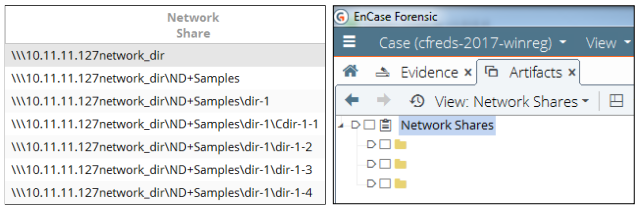
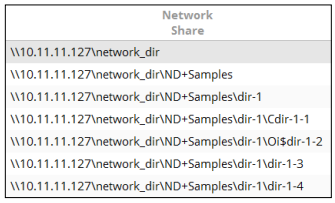
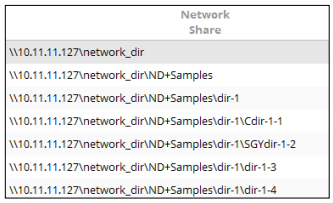
Description	Related screenshots (Windows 7 as an example)
• The tool did not report any timestamps relating to external devices.	• An example item of the 'System Info Parser – USB Devices – Devices' list in the 'Artifacts' view:  ◦ The same result was identified in the 'USB Devices' category in the 'Case Analyzer'.

- Network Connection
  - A valid network interface card (IP: 10.11.11.77) was reported as expected, but the following findings need to be taken into consideration. [Windows: ALL]
    - The tool ignored several well-known entries such as DNS addresses.
    - The tool utilized columns named 'IP' and 'Gateway' for the 'IP Gateway Pairs' category in the 'Case Analyzer'. However, data stored in those columns were 'DhcpIPAddress' and 'DhcpDefaultGateway' values, respectively. That is, the tool ignored 'IPAddress' and 'DefaultGateway' values for static IP addresses.
    - The tool reported invalid gateway addresses at the 'Network Interfaces – Registry' category in the 'Case Analyzer'.

- Recently Opened File and Directory
  - Partial opened file and directory related data was reported. [Windows: Vista, 7, 8, 8.1]
    - The tool identified recent file and directory entries stored in a user account (IEUser), but it did not report those data from other user accounts such as 'CFTT'.
  - Opened file and directory related data was not reported. [Windows: 10, 10RS1]

- Service and Driver
  - Service and driver related data was reported as expected, but the following findings need to be taken into consideration. [Windows: ALL]
    - The tool did not provide the meaning of each integer value stored in 'Type' and 'Start' values. That is, for example, it reported a 'Start' value as 2, instead of providing its meaning, 'Automatic'.

- Others
  - The tool also reported artifacts on the network drive connection, but it returnedsome invalid paring results. [Windows: 8, 8.1, 10, 10RS1]
    - The following shows the detailed findings. Note that the results were identified through the 'System Info Parser – Network Shares – Network Shares' list in the 'Artifacts' view, as well as the 'UNC Folders Visited' list of the 'Case Analyzer'.

Description	Related screenshots
• Directory paths were reported inaccurately. • The tool did not report the name (IP address) of network shares in Windows 8 and 8.1.	• Listed network shares [Windows 8]:  ◦ Missing backslash: → `\\\10.11.11.127\network_dir\...` ◦ Invalid network share names • Listed network shares [Windows 8.1]:  ◦ Missing backslash: → `\\\10.11.11.127\network_dir\...` ◦ Invalid directory name: → 'Cdir-1-1' should be corrected as 'dir-1-1' ◦ Invalid network share names • Listed network shares [Windows 10]:  ◦ Invalid directory name: → 'Cdir-1-1' should be corrected as 'dir-1-1' → 'Oi$dir-1-2' should be corrected as 'dir-1-2' • Listed network shares [Windows 10RS1]:  ◦ Invalid directory name: → 'Cdir-1-1' should be corrected as 'dir-1-1' → 'SGYdir-1-2' should be corrected as 'dir-1-2'

NOTES:

- The following test cases were not supported by the tool:
  - Application (Windows Store app related data)
  - Application Compatibility (Amcache)
  - Application Compatibility (Shimcache)
  - Dialog Usage
  - Network Drive
  - Remote Desktop
  - Run Command History
  - Search
  - UserAssist

See Table 10 below for more details.

Table 10. Test Results on Extracting Windows Registry Forensic Artifacts
(The hyphen (-) symbol means that reference hive files do not include any associated entries)

EnCase Forensic 8.07.00.93						
Test cases - registry artifacts	**Test result**					
	Vista	**7**	**8**	**8.1**	**10**	**10 RS1**
FA-01 Account	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*
FA-02 Application	***Partial***	***Partial***	***Partial***	***Partial***	***Partial***	***Partial***
FA-03 App Compatibility (Amcache)	-	-	*NA*	*NA*	*NA*	*NA*
FA-04 App Compatibility (Shimcache)	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
FA-05 Auto Run	***Partial***	***Partial***	***Partial***	***Partial***	***Partial***	***Partial***
FA-06 Dialog Usage	-	-	-	-	NA	NA
FA-07 External Device	***Partial***	***Partial***	***Partial***	***Partial***	***Partial***	***Partial***
FA-08 Network Connection	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*
FA-09 Network Drive	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
FA-10 OS Information	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*
FA-11 Recently Opened F&D	***Partial***	***Partial***	***Partial***	***Partial***	***Not As Expected***	***Not As Expected***
FA-12 Remote Desktop	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
FA-13 Run Command History	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
FA-14 Search	-	*NA*	*NA*	-	-	-
FA-15 Service and Driver	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*
FA-16 Shared Directory	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*
FA-17 ShellBag	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*
FA-18 Timezone	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*	*As Expected*
FA-19 UserAssist	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*