May 2022

Test Results for String Search Tool:
IPED Version 3.18.13

Federated Testing Suite for String Searching

Contents

Introduction

The Computer Forensics Tool Testing (CFTT) program is a joint project of the Department of Homeland Security (DHS) Science and Technology Directorate (S&T), the National Institute of Justice, and the National Institute of Standards and Technology (NIST) Special Programs Office and Information Technology Laboratory. 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, and the DHS 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 forensics 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 (https://www.cftt.nist.gov/).

This document reports the results from testing the string search function of IPED Version 3.18.13 using the CFTT Federated Testing Test Suite for String Searching, Version 5.

Federated Testing is an expansion of the CFTT program to provide forensic investigators and labs with test materials for tool testing and to support shared test reports. The goal of Federated Testing is to help forensic investigators to test the tools that they use in their labs and to enable sharing of tool test results. CFTT's Federated Testing Forensic Tool Testing Environment and included test suites can be downloaded from http://www.cftt.nist.gov/federated-testing.html and used to test forensic tools. The results can be optionally shared with CFTT, reviewed by CFTT staff, and then shared with the community.

Test results from other tools can be found on DHS's computer forensics web page, https://www.dhs.gov/science-and-technology/nist-cftt-reports.

How to Read This Report

This report is organized into the following sections:

1. Tested Tool Description. The tool name, version, vendor information, and support environment version (e.g., operating system version) are listed.
2. Testing Organization. The name and contact information of the organization that performed the tests are listed.
3. Results Summary. This section identifies any significant anomalies observed in the test runs. This section provides a narrative of key findings identifying where the tool meets expectations and provides a summary of any ways the tool did not meet expectations. The section also provides any observations of interest about the tool or about testing the tool including any observed limitations or organization-imposed restrictions on tool use.
4. Test Environment. Description of hardware and software used in tool testing in sufficient detail to satisfy the testing organization's policy and requirements.
5. Test Result Details by Case. Automatically generated test results that identify anomalies.
6. Appendix: Additional Details. Additional administrative details for each test case such as, who ran the test, when the test was run, computer used, etc.

Federated Testing Test Results for String Search Tool: IPED Version 3.18.13

Tool Description

Tool Name: IPED - Indexador e Processador de Evidências Digitais

Tool Version: 3.18.13

Tool Developer: Brazilian Federal Police, Luís Filipe da Cruz Nassif, Wladimir Luiz Caldas Leite and others from the Community - IPED GitHub page

IPED is an open source software that can be used to process and analyze digital evidence, often seized at crime scenes by law enforcement or in a corporate investigation by private examiners.

Testing Organization

Organization conducting test: Brazilian Federal Police
Report date (MM/DD/YYYY): 05/10/2022
Authored by: PCF João Fernando
Reviewed by: PCF Luís Filipe da Cruz Nassif
Reviewed by date: 05/12/2022
Approved by: PCF Mateus de Castro Polastro
Approved by date: 05/13/2022

This test report was generated using CFTT's Federated Testing Forensic Tool Testing Environment, see Federated Testing Home Page.

Results Summary

IPED got better results with the Windows file systems (FAT, ExFAT and NTFS).

IPED-3.18.13 uses TSK version 4.6.5 with patches from Blackbag and Luís Nassif to decode allocated files from APFS file systems, not supported by that official version of TSK.
Unallocated space of APFS file system is not supported by IPED-3.18.13.
It recovered DELETED file references from the Unix file systems of the test images, but all recovered files were empty (zero sized).
Live searches were able to find some deleted results from the test images.

Test Environment & Selected Cases

This section describes test hardware, software, data sets, and test cases.

Test Hardware and Software

Testing was performed using HP Z8 Workstation with Windows 10 Pro for Workstations (21H2)

Testing was performed using CFTT's Federated Testing Test Suite Version 5.

Test Data Sets and Test Cases

This section discusses the test data sets and the test cases used in testing.

Test Data Sets

String search test data set package Version 1.1 was used. The package can be downloaded from either the CFTT web site (www.cftt.nist.gov then select String Search) or the CFReDS web site (www.cfreds.nist.gov). The package includes two dd files with known content. One of the dd test images contains target strings within FAT, ExFAT, and NTFS file systems (Windows). The other dd test image contains target strings within HFS+ journaled case insensitive (OSXJ), HFS+ journaled case sensitive (OSXC), ext4, and APFS (Apple file system) file systems (UNIX-like).

In general, each target string is encoded in ASCII and located in both an active file and a recoverable deleted file in each partition of the test image. The Windows dd image also has a block of unallocated storage that contains the target strings without a file system. Some of the target strings are also encoded in Unicode UTF-8, UTF-16BE and UTF-16LE with a byte-order-mark. Test case FT-SS-07 is organized to test language and Unicode specific situations such as Unicode UTF-16 without a byte-order-mark, Unicode text with and without combining characters (diacritic marks), and Unicode text with and without ligatures ("fi" as two characters and as one character). Test case FT-SS-09 is organized to test specific situations such as formatted strings, strings spanning file fragments, and strings located in inaccessible areas. Each instance of a target string also has a unique associated string ID located immediately after the target string. The string ID helps identify the specific string matched by the search tool.

Test Case Descriptions

The following table gives a brief description of available test cases in the data sets. Not all test cases are used for all data sets. To see what tests were run, see section 6.

Case	Case Description
FT-SS-01	Find an ASCII string
FT-SS-02	Search for a substring and search ignore case
FT-SS-03	Search for words (whole word match vs a substring match)
FT-SS-04	Search Logical AND
FT-SS-05	Search Logical OR
FT-SS-06	Search Logical NOT
FT-SS-07-CJK-char	Search Unicode Chinese/Japanese ideograms (Asian)
FT-SS-07-CJK-hangul	Search Unicode CJK Korean Hangul (Asian)
FT-SS-07-CJK-kana	Search Unicode CJK Japanese phonetic Kana (Asian)
FT-SS-07-Cyrillic	Search Unicode Cyrillic (Russian)
FT-SS-07-Latin	Search Unicode Latin characters with diacritical marks
FT-SS-07-NoBOM	Search Unicode 16 without a byte-order-mark
FT-SS-07-Norm	Search Unicode for normalized diacritic marks (NFC & NFD) and ligatures (NFKC & NFKD)
FT-SS-07-RTL	Search Unicode Right-To-Left (Arabic)
FT-SS-08-Email	Search tool-defined queries -- Email Address
FT-SS-08-Phone	Search tool-defined queries -- Telephone Number
FT-SS-08-SS	Search tool-defined queries -- Social Security Number
FT-SS-09-Doc	Search Formatted Document Text
FT-SS-09-Frag*	Search Fragmented File
FT-SS-09-Lost*	Search Inaccessible (lost) Areas
FT-SS-09-MFT*	Search File in NTFS Master File Table (MFT)
FT-SS-09-Meta	Search file name substring in Meta-data
FT-SS-10-Hex	Search Hexadecimal Character Match
FT-SS-10-Regex	Search Pattern Character Match

Some test cases are for specific features, e.g., logical conditions (**and**, **or**, **not**), built in searches (email, telephone numbers), etc. Three test cases (marked with "*"), FT-SS-09-Frag, FT-SS-09-Lost &

Test case FT-SS-09-Stem is not supported by IPED.

Test Result Details by Case (per Data Set)

A string search tool may implement more than one search algorithm (also known as a search engine) for searching text. The two most common search engines are *indexed search* and *live search*. An indexed search reads all the acquired data once before doing any searching and builds an index to all words found. Each query can be looked up quickly in the index. A Live search reads all the acquired data for each query.

This section presents test results by test image: Windows file systems, or UNIX-like file systems. For each test image, there is a result table for each search engine tested. Each table shows results by test case of the number of expected search hits, the number of actual search hits and the number of strings missed (i.e., expected hits minus actual hits) for allocated files, deleted files and unallocated space.

The following search engines were tested: Indexed.

Results for Data Set: Windows

This section provides results for the Windows data set.

Results for Indexed Search of Windows Data Set

The table columns contain the following information:

- **Case:** The test case identifier.
- **Expected String:** The strings that should be reported by the search.
- **Active Files:** A group of three columns (**Expected, Hits and Misses**) giving the number of hits and misses when searching for the expected string in an active file.
- **Deleted Files:** A group of three columns (**Expected, Hits and Misses**) giving the number of hits and misses when searching for the expected string in a deleted file.
- **Unallocated Space:** A group of three columns (**Expected, Hits and Misses**) giving the number of hits and misses when searching for the expected string in unallocated space.
- **Expected:** The number of instances of the expected string found in the group (i.e., Active files, Deleted files or Unallocated space).
- **Hits:** The number of times the expected string was found in the group.
- **Misses:** The number of times the expected string was missed (not found) in the group.

In the Expected String column for test case FT-SS-09-DOC each string is labeled to indicate features of the expected string. The labels include the file type (.doc, .docx or .html) and the encoding of the string (if a .doc file). If the string has embedded formatting it is labeled as *Formatted*, e.g., the string *crossbow* has the substring *cross* formatted as bold and underlined, i.e., **cross**bow.

Note: the first row of results for a test case is a summary for all the strings that should be found for that case.

Results for Indexed Search of Windows Data Set										
Case	Expected String	Active Files			Deleted Files			Unallocated Space		
		Expected	Hits	Misses	Expected	Hits	Misses	Expected	Hits	Misses
FT-SS-01		3	3	0	3	3	0	1	1	0
	DireWolf	3	3	0	3	3	0	1	1	0
FT-SS-02		15	15	0	15	15	0	5	5	0
	WOLF	3	3	0	3	3	0	1	1	0
	wolf	3	3	0	3	3	0	1	1	0
	Wolf	3	3	0	3	3	0	1	1	0
	DireWolf	3	3	0	3	3	0	1	1	0
	WereWolf	3	3	0	3	3	0	1	1	0
FT-SS-03		9	9	0	9	9	0	3	3	0
	WOLF	3	3	0	3	3	0	1	1	0
	wolf	3	3	0	3	3	0	1	1	0
	Wolf	3	3	0	3	3	0	1	1	0
FT-SS-04		3	3	0	3	3	0	0	0	0
	panda and fox	3	3	0	3	3	0	0	0	0
FT-SS-05		6	6	0	6	6	0	2	2	0
	DireWolf	3	3	0	3	3	0	1	1	0
	WereWolf	3	3	0	3	3	0	1	1	0
FT-SS-06		12	12	0	12	12	0	0	0	0
	fox and not tiger	12	12	0	12	12	0	0	0	0
FT-SS-07-CJK-char		18	18	0	18	18	0	6	0	6
	中国	9	9	0	9	9	0	3	0	3
	東京	9	9	0	9	9	0	3	0	3
FT-SS-07-CJK-hangul		9	9	0	9	9	0	3	0	3
	서울	9	9	0	9	9	0	3	0	3
FT-SS-07-CJK-kana		18	18	0	18	18	0	6	0	6
	スバル	9	9	0	9	9	0	3	0	3
	みつびし	9	9	0	9	9	0	3	0	3
FT-SS-07-Cyrillic		9	9	0	9	9	0	3	0	3
	Сибирь	9	9	0	9	9	0	3	0	3

Results for Indexed Search of Windows Data Set										
Case	Expected String	Active Files			Deleted Files			Unallocated Space		
		Expected	Hits	Misses	Expected	Hits	Misses	Expected	Hits	Misses
FT-SS-07-Latin		18	18	0	18	18	0	6	6	0
	garçon	9	9	0	9	9	0	3	3	0
	Schönheit	9	9	0	9	9	0	3	3	0
FT-SS-07-NoBOM		39	39	0	39	39	0	13	4	9
	Россия	9	9	0	9	9	0	3	0	3
	فلافل	9	9	0	9	9	0	3	0	3
	中國	9	9	0	9	9	0	3	0	3
	QuarterHorse	12	12	0	12	12	0	4	4	0
FT-SS-07-Norm		75	75	0	75	75	0	25	20	5
	mañana (NFD)	9	9	0	9	9	0	3	2	1
	infinity (No Ligature)	12	12	0	12	12	0	4	4	0
	Mäuse (NFD)	9	9	0	9	9	0	3	3	0
	infinity (Ligature)	9	9	0	9	9	0	3	0	3
	Mäuse (NFC)	9	9	0	9	9	0	3	3	0
	libertà (NFC)	9	9	0	9	9	0	3	3	0
	libertà (NFD)	9	9	0	9	9	0	3	2	1
	mañana (NFC)	9	9	0	9	9	0	3	3	0
FT-SS-07-RTL		9	9	0	9	9	0	3	0	3
	الكسكس	9	9	0	9	9	0	3	0	3
FT-SS-08-Email		21	21	0	21	21	0	7	7	0
	iron.man@marvel.com	12	12	0	12	12	0	4	4	0
	potus@capitol.gov	3	3	0	3	3	0	1	1	0
	berlin@deutschland.net	3	3	0	3	3	0	1	1	0
	kgb@moscow.red.square.ru	3	3	0	3	3	0	1	1	0
FT-SS-08-Phone		21	21	0	21	21	0	7	7	0
	301.555-9009	12	12	0	12	12	0	4	4	0
	800-555-1122	3	3	0	3	3	0	1	1	0
	(901)555-1111	3	3	0	3	3	0	1	1	0
	202.555.3270	3	3	0	3	3	0	1	1	0
FT-SS-08-SS		9	9	0	9	9	0	3	3	0
	123-45-6789	3	3	0	3	3	0	1	1	0
	999-55-1321	3	3	0	3	3	0	1	1	0
	987-65-4321	3	3	0	3	3	0	1	1	0

Results for Indexed Search of Windows Data Set										
Case	Expected String	Active Files			Deleted Files			Unallocated Space		
		Expected	Hits	Misses	Expected	Hits	Misses	Expected	Hits	Misses
FT-SS-09-Doc		16	16	0	0	0	0	16	16	0
	longbow .html	2	2	0	0	0	0	2	2	0
	shotgun Formatted .doc UTF-16	2	2	0	0	0	0	2	2	0
	revolver .doc UTF-16	2	2	0	0	0	0	2	2	0
	peroxide .docx	2	2	0	0	0	0	2	2	0
	nitroglycerin Formatted .docx	2	2	0	0	0	0	2	2	0
	rifle .doc UTF-8	2	2	0	0	0	0	2	2	0
	crossbow Formatted .html	2	2	0	0	0	0	2	2	0
	flintlock Formatted .doc UTF-8	2	2	0	0	0	0	2	2	0
FT-SS-09-Frag		2	2	0	0	0	0	0	0	0
	Washington	1	1	0	0	0	0	0	0	0
	California	1	1	0	0	0	0	0	0	0
FT-SS-09-Lost		0	0	0	0	0	0	4	4	0
	SecretKey	0	0	0	0	0	0	2	2	0
	disconnected	0	0	0	0	0	0	2	2	0
FT-SS-09-MFT		4	4	0	4	4	0	0	0	0
	bear	4	4	0	4	4	0	0	0	0
FT-SS-09-Meta		6	6	0	6	6	0	2	2	0
	cañón	3	3	0	3	3	0	1	1	0
	thunderbird	3	3	0	3	3	0	1	1	0
FT-SS-10-Hex		3	3	0	3	3	0	1	1	0
	panda	3	3	0	3	3	0	1	1	0
FT-SS-10-Regex		6	6	0	6	6	0	2	2	0
	DireWolf	3	3	0	3	3	0	1	1	0
	WereWolf	3	3	0	3	3	0	1	1	0

Meta-Data results for Indexed Search of Windows Data Set

The following table presents search results for strings located in file system meta-data.
The **Case** column identifies the test case, the **String** column identifies the search string,
the **Partition** column identifies the partition (file system) where the string is located and
the **Seen** column records if the search tool reported at least one instance of the string (yes or no)
in meta-data.

Meta-Data Results for Indexed Search of Windows Data Set			
Case	**String**	**Partition**	**Seen**
FT-SS-09-Meta			
	thunderbird	ntfs	Yes
	cañón	fat32	Yes
	cañón	exfat	Yes
	cañón	ntfs	Yes

Comments on Indexed Search of Windows Data Set

IPED configuration is based on profiles. Each profile consists of a folder with several text files.
To match the configuration for each test, some profiles were created.

General considerations on IPED configurations:

- IPED uses Apache Lucene library to index file content and metadata.
- IPED automatically detects the text encoding, so it is not possible to select ASCII or Unicode.
- Finding Whole Words is the default IPED behavior. To match substrings, wildcards are used.
- Case sensitive is configured changing the option convertCharsToLowerCase found inside AdvancedConfig.txt on the profile folder.
- Logical operators on IPED are uppercased.
- IPED carves files from the Unallocated Space and indexes each filesystem unallocated space.
    - So, some deleted strings are also found within the unallocated space, counting them twice.
- IPED was not designed to count the number of hits when searching for a string. It counts the number of fragments with approximately 100 characters that contains hits.
    - Sometimes one hit may be broken into two text fragments in the Hits results Panel and counted twice. Therefore, the user must manually count the number of hits using the tool's interface.
- Known issue: Misses for non-Latin languages in unallocated space
  - https://github.com/sepinf-inc/IPED/issues/441

The following table presents any comments recorded during testing for a test case.

Case	Comments on Indexed Search of Windows Data Set
FT-SS-01	All strings found. Search string used: DireWolf

Case	Comments on Indexed Search of Windows Data Set
FT-SS-02	All strings found. Search string used: *wolf
FT-SS-03	All strings found. Search string used: Wolf
FT-SS-04	All strings found. Search string used: panda AND fox
FT-SS-05	All strings found. Search string used: Were* OR DireW*
FT-SS-06	All strings found. Search string used: fox AND NOT tiger
FT-SS-07-CJK-char	It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings. Search strings used: 中国 東京
FT-SS-07-CJK-hangul	Korean Hangul glyphs were shown as squares (Windows 10). Install the Windows 10 Korean language pack to fix this issue. It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings. Search string used: 서울
FT-SS-07-CJK-kana	It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings. Search strings used: スバル みつびし
FT-SS-07-Cyrillic	It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings. Search string used: Сибирь
FT-SS-07-Latin	All strings found. Search strings used: garçon Schönheit
FT-SS-07-NoBOM	It is not possible to search for Russia, Asian or Arabic characters on Unallocated Space using IPED's search bar. To find all strings within unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings. Search strings used: QuarterHorse Россия 中國 فلافل

Case	Comments on Indexed Search of Windows Data Set
FT-SS-07-Norm	Search string used: copy/paste from file ft-ss-07-Norm-strings.txt IPED found several false positives while searching for string mañana NFD. Putting the string on double quotes gives better results with no false positives, but it misses the string 3177 on Unallocated Space. Without the double quotes the string 3177 is wrongly found as man, so the box will not be checked. IPED couldn't find string 3241 libertà, but there were no false positives. IPED struggles with UTF-8 NFD strings when they are on Unallocated Space. IPED couldn't find any infinity NFD strings on Unallocated Space using the search bar. All strings can be found using IPED's Hex Viewer, selecting the correct encoding and searching them. It's not the common use of the tool, so the boxes related to them will not be checked.
FT-SS-07-RTL	It is not possible to search for Arabic characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings. Search string used: الكسكس
FT-SS-08-Email	IPED found the expected strings but broke some of them into two text fragments in the Hits results Panel. It's the expected behavior. See **"General considerations on IPED configurations"** for more information on the way IPED counts hits. To find the exact number of hits, the user must count them manually.
FT-SS-08-Phone	IPED telephone numbers search feature is configured as a regex expression before processing and indexing the files. The regex expression must be put on the file RegexConfig.txt inside your case profile folder. IPED default regex for telephone numbers follow the Brazilian format. To correctly find US numbers, a new regular expression was created and inserted on the RegexConfig.txt before indexing the evidence. This is the regular expression created: [^0-9]((\([0-9]{3}\))|([0-9]{3}(\.|-)))[0-9]{3}(\.|-)[0-9]{4}[^0-9]
FT-SS-08-SS	IPED SSN search feature is configured as a regex expression before processing and indexing the files. The regex expression must be put on the file RegexConfig.txt inside your case profile folder. IPED default settings don’t include SSN search. For this test case, a simple regular expression was created: ([0-9]{3})\-([0-9]{2})\-([0-9]{4})

Case	Comments on Indexed Search of Windows Data Set
FT-SS-09-Doc	IPED carves files from the Unallocated Space and from many allocated file types. So there are two hits for every carved doc file, one inside the carved doc file and the other one when you select the unallocated space. This behavior is expected, but you can disable the file carving feature. It does not happen with the strings inside HTML/DOCX files.
FT-SS-09-Frag	All strings were found.
FT-SS-09-Lost	All strings were found.
FT-SS-09-MFT	All files from the MFT were recovered.
FT-SS-09-Meta	For the string cañón: FAT32 - All the strings inside the files. Two Substrings that are part of filenames EXTFAT - Two results inside the meta-data of a folder named extfat. Two Substrings that are part of filenames NTFS - 5 results inside $LogFile and 2 inside $MFT. Two Substrings that are part of filenames For the string thunderbird: NTFS: 8 results inside $LogFile and 2 inside $MFT. 4 Substrings that are part of filenames
FT-SS-10-Hex	IPED search bar doesn't allow hexadecimal searches. To do this kind of search, first select the dd image, then change to hex view and then search for the hex string. All strings were found.
FT-SS-10-Regex	Regular expression search is done putting the expression between forward slashes. All strings were found.

Results for Data Set: UNIX

This section provides results for the UNIX data set.

Results for Indexed Search of UNIX Data Set

The table columns contain the following information:

- **Case:** The test case identifier.
- **Expected String:** The strings that should be reported by the search.
- **Active Files:** A group of three columns (**Expected, Hits and Misses**) giving the number of hits and misses when searching for the expected string in an active file.
- **Deleted Files:** A group of three columns (**Expected, Hits and Misses**) giving the number of hits and misses when searching for the expected string in a deleted file.
- **Unallocated Space:** A group of three columns (**Expected, Hits and Misses**) giving the number of hits and misses when searching for the expected string in unallocated space.
- **Expected:** The number of instances of the expected string found in the group (i.e., Active files, Deleted files or Unallocated space).

- **Hits:** The number of times the expected string was found in the group.
- **Misses:** The number of times the expected string was missed (not found) in the group.

In the Expected String column for test case FT-SS-09-DOC each string is labeled to indicate features of the expected string. The labels include the file type (.doc, .docx or .html) and the encoding of the string (if a .doc file). If the string has embedded formatting it is labeled as *Formatted*, e.g., the string *crossbow* has the substring *cross* formatted as bold and underlined, i.e., **cross**bow.

Note: the first row of results for a test case is a summary for all the strings that should be found for that case.

Results for Indexed Search of UNIX Data Set										
Case	Expected String	Active Files			Deleted Files			Unallocated Space		
		Expected	Hits	Misses	Expected	Hits	Misses	Expected	Hits	Misses
FT-SS-01		4	4	0	4	4	0	0	0	0
	DireWolf	4	4	0	4	4	0	0	0	0
FT-SS-02		20	20	0	20	20	0	0	0	0
	WOLF	4	4	0	4	4	0	0	0	0
	wolf	4	4	0	4	4	0	0	0	0
	Wolf	4	4	0	4	4	0	0	0	0
	DireWolf	4	4	0	4	4	0	0	0	0
	WereWolf	4	4	0	4	4	0	0	0	0
FT-SS-03		12	12	0	12	12	0	0	0	0
	WOLF	4	4	0	4	4	0	0	0	0
	wolf	4	4	0	4	4	0	0	0	0
	Wolf	4	4	0	4	4	0	0	0	0
FT-SS-04		4	4	0	4	4	0	0	0	0
	panda and fox	4	4	0	4	4	0	0	0	0
FT-SS-05		8	8	0	8	8	0	0	0	0
	DireWolf	4	4	0	4	4	0	0	0	0
	WereWolf	4	4	0	4	4	0	0	0	0
FT-SS-06		16	16	0	16	16	0	0	0	0
	fox and not tiger	16	16	0	16	16	0	0	0	0
FT-SS-07-CJK-char		24	24	0	24	0	24	0	0	0
	中国	12	12	0	12	0	12	0	0	0
	東京	12	12	0	12	0	12	0	0	0

Results for Indexed Search of UNIX Data Set										
Case	Expected String	Active Files			Deleted Files			Unallocated Space		
		Expected	Hits	Misses	Expected	Hits	Misses	Expected	Hits	Misses
FT-SS-07-CJK-hangul		12	12	0	12	0	12	0	0	0
	서울	12	12	0	12	0	12	0	0	0
FT-SS-07-CJK-kana		24	24	0	24	0	24	0	0	0
	スバル	12	12	0	12	0	12	0	0	0
	みつびし	12	12	0	12	0	12	0	0	0
FT-SS-07-Cyrillic		12	12	0	12	0	12	0	0	0
	Сибирь	12	12	0	12	0	12	0	0	0
FT-SS-07-Latin		24	24	0	24	24	0	0	0	0
	garçon	12	12	0	12	12	0	0	0	0
	Schönheit	12	12	0	12	12	0	0	0	0
FT-SS-07-NoBOM		52	52	0	52	16	36	0	0	0
	Россия	12	12	0	12	0	12	0	0	0
	فلافل	12	12	0	12	0	12	0	0	0
	中國	12	12	0	12	0	12	0	0	0
	QuarterHorse	16	16	0	16	16	0	0	0	0
FT-SS-07-Norm		100	100	0	100	80	20	0	0	0
	mañana (NFD)	12	12	0	12	8	4	0	0	0
	infinity (No Ligature)	16	16	0	16	16	0	0	0	0
	Mäuse (NFD)	12	12	0	12	12	0	0	0	0
	infinity (Ligature)	12	12	0	12	0	12	0	0	0
	Mäuse (NFC)	12	12	0	12	12	0	0	0	0
	libertà (NFC)	12	12	0	12	12	0	0	0	0
	libertà (NFD)	12	12	0	12	8	4	0	0	0
	mañana (NFC)	12	12	0	12	12	0	0	0	0
FT-SS-07-RTL		12	12	0	12	0	12	0	0	0
	الكسكس	12	12	0	12	0	12	0	0	0

Results for Indexed Search of UNIX Data Set										
Case	Expected String	Active Files			Deleted Files			Unallocated Space		
		Expected	Hits	Misses	Expected	Hits	Misses	Expected	Hits	Misses
FT-SS-08-Email		28	28	0	28	28	0	0	0	0
	iron.man@marvel.com	16	16	0	16	16	0	0	0	0
	potus@capitol.gov	4	4	0	4	4	0	0	0	0
	berlin@deutschland.net	4	4	0	4	4	0	0	0	0
	kgb@moscow.red.square.ru	4	4	0	4	4	0	0	0	0
FT-SS-08-Phone		28	28	0	28	28	0	0	0	0
	301.555-9009	16	16	0	16	16	0	0	0	0
	800-555-1122	4	4	0	4	4	0	0	0	0
	(901)555-1111	4	4	0	4	4	0	0	0	0
	202.555.3270	4	4	0	4	4	0	0	0	0
FT-SS-08-SS		12	12	0	12	12	0	0	0	0
	123-45-6789	4	4	0	4	4	0	0	0	0
	999-55-1321	4	4	0	4	4	0	0	0	0
	987-65-4321	4	4	0	4	4	0	0	0	0
FT-SS-09-Doc		16	16	0	0	0	0	0	0	0
	longbow .html	2	2	0	0	0	0	0	0	0
	shotgun Formatted .doc UTF-16	2	2	0	0	0	0	0	0	0
	revolver .doc UTF-16	2	2	0	0	0	0	0	0	0
	peroxide .docx	2	2	0	0	0	0	0	0	0
	nitroglycerin Formatted .docx	2	2	0	0	0	0	0	0	0
	rifle .doc UTF-8	2	2	0	0	0	0	0	0	0
	crossbow Formatted .html	2	2	0	0	0	0	0	0	0
	flintlock Formatted .doc UTF-8	2	2	0	0	0	0	0	0	0

Results for Indexed Search of UNIX Data Set										
Case	Expected String	Active Files			Deleted Files			Unallocated Space		
		Expected	Hits	Misses	Expected	Hits	Misses	Expected	Hits	Misses
FT-SS-09-Meta		8	8	0	8	8	0	0	0	0
	cañón	4	4	0	4	4	0	0	0	0
	thunderbird	4	4	0	4	4	0	0	0	0
FT-SS-10-Hex		4	4	0	4	4	0	0	0	0
	panda	4	4	0	4	4	0	0	0	0
FT-SS-10-Regex		8	8	0	8	8	0	0	0	0
	DireWolf	4	4	0	4	4	0	0	0	0
	WereWolf	4	4	0	4	4	0	0	0	0

Meta-Data results for Indexed Search of UNIX Data Set

The following table presents search results for strings located in file system meta-data.
The **Case** column identifies the test case, the **String** column identifies the search string,
the **Partition** column identifies the partition (file system) where the string is located and
the **Seen** column records if the search tool reported at least one instance of the string (yes or no)
in meta-data.

Meta-Data Results for Indexed Search of UNIX Data Set			
Case	String	Partition	Seen
FT-SS-07-CJK-char			
	中国	osxj	No
	中国	osxc	No
	中国	apfs	No
	東京	osxj	No
	東京	osxc	No
	東京	apfs	No
FT-SS-07-Cyrillic			
	Сибирь	osxj	No
	Сибирь	osxc	No
	Сибирь	apfs	No

Meta-Data Results for Indexed Search of UNIX Data Set			
Case	**String**	**Partition**	**Seen**
FT-SS-07-NoBOM			
	فلافل	osxj	No
	فلافل	osxc	No
	فلافل	apfs	No
	Россия	osxj	No
	Россия	osxc	No
	Россия	apfs	No
	中國	osxj	No
	中國	osxc	No
	中國	apfs	No
FT-SS-07-RTL			
	الكسكس	osxj	No
	الكسكس	osxc	No
	الكسكس	apfs	No
FT-SS-09-Meta			
	thunderbird	osxj	Yes
	thunderbird	osxc	Yes
	thunderbird	apfs	Yes
	thunderbird	ext4	Yes
	cañón	ext4	Yes

Comments on Indexed Search of UNIX Data Set

IPED configuration is based on profiles. Each profile consists of a folder with several text files. To match the configuration for each test, some profiles were created.

General considerations on IPED configurations:

- IPED uses Apache Lucene library to index file content and metadata.
- IPED automatically detects the text encoding, so it is not possible to select ASCII or Unicode.
- Finding Whole Words is the default IPED behavior. To match substrings, wildcards are used.
- Case sensitive is configured changing the option convertCharsToLowerCase found inside AdvancedConfig.txt on the profile folder.
- Logical operators on IPED are uppercased.
- Known issue: Misses for non-Latin languages in unallocated space - https://github.com/sepinf-inc/IPED/issues/441

- **Unallocated space of APFS file system is not supported by IPED-3.18.13.**
    - **It recovered DELETED file references from the Unix file systems of the test images, but all recovered files were empty (zero sized).**
- IPED was not designed to count the number of hits when searching for a string. It counts the number of fragments with approximately 100 characters that contains hits.
    - Sometimes one hit may be broken into two text fragments in the Hits results Panel and counted twice. Therefore, the user must manually count the number of hits using the tool's interface.

The following table presents any comments recorded during testing for a test case.

Case	Comments on Indexed Search of UNIX Data Set
FT-SS-01	Search string used: DireWolf APFS: No deleted files or unallocated space could be recovered, but deleted strings were all found using live search. HFS (OSXJ): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. EXT4: deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. HFS (OSXC): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized.
FT-SS-02	Search string used: *wolf APFS: No deleted files or unallocated space could be recovered, but deleted strings were all found using live search. HFS (OSXJ): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. EXT4: deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. HFS (OSXC): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized.

Case	Comments on Indexed Search of UNIX Data Set
FT-SS-03	APFS: No deleted files or unallocated space could be recovered, but deleted strings were all found using live search. HFS (OSXJ): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. EXT4: deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. HFS (OSXC): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized.
FT-SS-04	Search string used: panda AND fox APFS: No deleted files or unallocated space could be recovered, but deleted strings were all found using live search. HFS (OSXJ): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. EXT4: deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. HFS (OSXC): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. Several false positives were found. APFS: 0840, 0841, 0842, 0843, 0844, 0845, 0846 and 0847. HFS (OSXJ): 0824, 0825, 0826, 0827, 0828, 0829, 0830 and 0831. EXT4: 0816, 0817, 0818, 0819, 0820, 0821, 0822 and 0823. HFS (OSXC): 0832, 0833, 0834, 0835, 0836, 0837, 0838 and 0839.
FT-SS-05	Search string used: Were* OR DireW* APFS: No deleted files or unallocated space could be recovered, but deleted strings were all found using live search. HFS (OSXJ): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. EXT4: deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. HFS (OSXC): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized.
FT-SS-06	Search string used: fox AND NOT tiger. All deleted strings were found on unallocated space, but because they were inside the same unallocated space as other strings, the NOT operator excluded all the results (expected behavior).

Case	Comments on Indexed Search of UNIX Data Set
FT-SS-07-CJK-char	Search strings used: 中国 東京 No meta-data results were found. It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings.
FT-SS-07-CJK-hangul	Search string used: 서울 Koren Hangul glyphs were shown as squares (Windows 10). Install the Korean language pack to fix this issue. It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings.
FT-SS-07-CJK-kana	Search string used: スバル みつびし It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings.
FT-SS-07-Cyrillic	Search string used: Сибирь No meta-data results were found. It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings.
FT-SS-07-Latin	Search strings used: garçon Schönheit APFS: No deleted files or unallocated space could be recovered, but deleted strings were all found using live search. HFS (OSXJ): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. EXT4: deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized. HFS (OSXC): deleted strings were found inside unallocated space. Deleted files were found inside the unallocated space, but could not be recovered correctly, all deleted files recovered were empty zero sized.
FT-SS-07-NoBOM	Search string used: QuarterHorse Россия 中國 فلافل No meta-data results were found. It is not possible to search for Russian/Arabic/Asian characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings.

Case	Comments on Indexed Search of UNIX Data Set
FT-SS-07-Norm	Search strings from file ft-ss-07-Norm-strings.txt Strings searched using double quotes. All deleted strings found were found on unallocated space, all deleted files recovered were empty zero sized. All results for NFC and NFD were shown separately. IPED returned false positives for string mañana NFD, the substring man was counted several times. Putting the string between double quotes, the results are better with no false positives, but it misses the strings 3189, 3197 and 3205 on Unallocated Space. Without the double quotes IPED the strings 3189 and 3205 are found as man and the string 3197 as ana. The strings infinity NFD were not found on unallocated space, but were found using the hex viewer and selecting the proper encoding. NFD mañana - missed some utf-8 deleted strings. NFD libertà - missed some utf-8 deleted strings. NFD infinity - missed all deleted strings
FT-SS-07-RTL	Search string used: الكسكس No meta-data results were found. It is not possible to search for these characters on Unallocated Space using IPED's search bar. To find all strings withing unallocated space, change IPED to Hexadecimal view, select the charset and search for the strings.
FT-SS-08-Email	All deleted strings found were found on unallocated space, all deleted files recovered were empty zero sized. APFS deleted strings were only found with live search. Some results were broken into two text fragments in the Hits results Panel. It's the expected behavior. See **"General considerations on IPED configurations"** for more information on the way IPED counts hits. To find the exact number of hits, the user must count them manually.
FT-SS-08-Phone	IPED telephone numbers search feature is configured as a regex expression before processing and indexing the files. The regex expression must be put on the file RegexConfig.txt inside your case profile folder. IPED default regex for telephone numbers follow the Brazilian format. To correctly find US numbers, a new regular expression was created and inserted on the RegexConfig.txt before indexing the evidence. This is the regular expression created: [^0-9]((\([0-9]{3}\))|([0-9]{3}(\.|-)))[0-9]{3}(\.|-)[0-9]{4}[^0-9]

Case	Comments on Indexed Search of UNIX Data Set
FT-SS-08-SS	IPED SSN search feature is configured as a regex expression before processing and indexing the files. The regex expression must be put on the file RegexConfig.txt inside your case profile folder. IPED default settings don't include SSN search. For this test case, a simple regular expression was created: ([0-9]{3})\-([0-9]{2})\-([0-9]{4}) All deleted strings found were found on unallocated space, all deleted files recovered were empty zero sized. APFS deleted strings were only found with live search.
FT-SS-09-Doc	All strings were found.
FT-SS-09-Meta	All strings found.
FT-SS-10-Hex	IPED search bar doesn't allow hexadecimal searches. To do this kind of search, first select the dd image, then change to hex view and then search for the hex string. All strings were found.
FT-SS-10-Regex	Regular expression search is done putting the expression between forward slashes. All deleted strings found were found on unallocated space, all deleted files recovered were empty zero sized. APFS deleted strings were only found with live search.

OS: Linux Version 4.13.0-37-generic
Done: 2022-05-04 13:48:31.189608
Federated Testing Version 5, released 3/12/2020

END of REPORT