



PTK Forensics

Dario Forte, Founder and Ceo DFLabs

The Sleuth Kit and Open Source Digital Forensics Conference

What PTK is about



- PTK forensics is a computer forensic framework based on command line tools in the SleuthKit to which many new software modules were added. Its mission is: Making open source forensic tools more usable and merging the opensource/free/and commercial tools in an effective way
- Thanks to this approach, users can investigate a system much easier, without spending a big budget
- Born as a free interface in order to improve the features already present in 'Autopsy Forensic Browser' (the former TSK interface), PTK Forensic is now much more. Thus, in addition to providing the features present in the 'Autopsy Forensic Browser' it now implements numerous new essential forensic features.
-
- PTK forensics is more than just a new graphic and highly professional interface based on Ajax and other advanced technologies; it offers many features such as analysis, search and management of complex digital investigation cases.

PTK Forensics is available in two versions:

PTK Forensics free basic edition

PTK Forensics full version.



- The first PTK Forensics version was released in May 2008 (and presented at the DoD Cybercrime Conference in Jan 08).
- Thanks to **Sourceforge.net** and the continuous interest in this project, we have already registered **15811 downloads** not to mention external mirrors.
- Access to the **official website** <http://ptk.dflabs.com> is increasing constantly (more than 50000 visits so far).
- We presented the project at the **DoD, DHS, Europol, interpol, NATO** and it had a huge success.

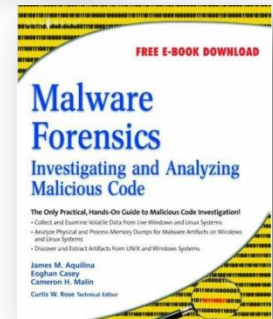
PTK Forensics: reference



- Given the success registered, the project was included in the **SIFT (Sans Investigative Forensic Toolkit)** at its second major release.



- Malware Forensics: Investigating and Analyzing Malicious Code**, By Cameron H. Malin, Eoghan Casey, James M. Aquilina



- SIFT Workstation 2.0: SANS Investigative Forensic Toolkit**, By Russ McRee

Where we Are: Free Vs Full



- Strategy: having a free version for “basic” purposes and full version with many enhancements and features.
 - Method: having 5 types of advanced features in the Full version rather than the free version.
 - The full version is continuously updated, so when a new one is added, the oldest one is added to the free version, excepting the new indexing engine.
- Results: the full version will finance the development of the free version
- Free version users will have their software up to date.
- The community will hopefully work on the free version, improving what has been shifted down from the full one.

PTK Forensics comparison



Features	PTK Forensics Free	PTK Forensics Full
Dedicated Technical Support	-	X
Evidence management	X	X
User management	X	X
Evidence integrity control	X	X
Indexing type	base	advanced
Indexing timeline	X	X
Indexing MD5 , SHA1	X	X
Filetype indexing	X	X
Indexing keyword search	X	X
Job advanced management	-	X
Filter management based on file type or timestamp	X	X
Recursive visualization	X	X
Hex value interpreter	-	X
Tabular timeline	X	X

PTK Forensics comparison

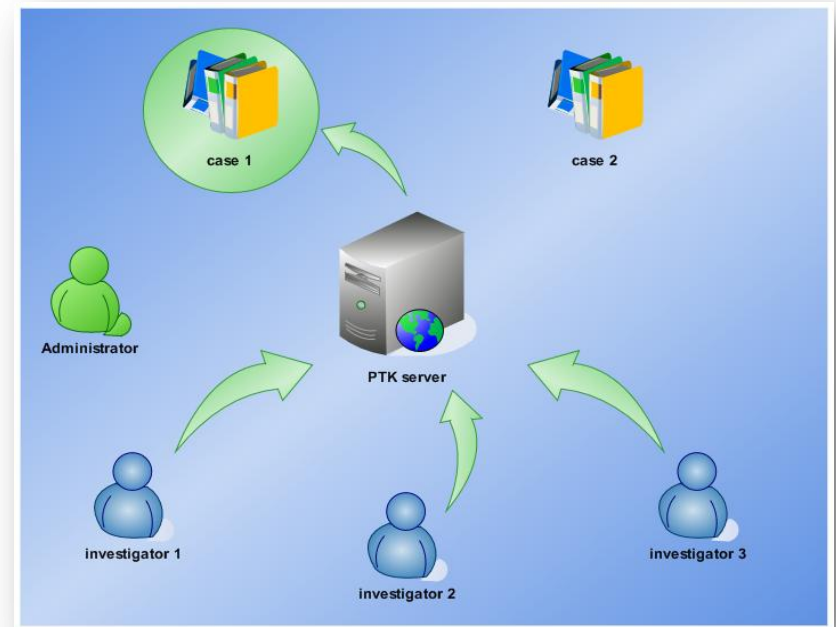


Features	PTK Forensics Free	PTK Forensics Full
Gallery section	X	X
Image details section	X	X
Data unit section	X	X
Bookmark management	X	X
Report section	X (with limitation)	X
Datacarving section (zero-storage)	-	X
Adding header and footer custom	-	X
Hashset section (know good/bad)	-	X
Plug-in section	-	X
Graphical timeline	X	X
Filter management based on macb time	X	X
Keyword search section	X	X
Regular expression search	X	X
Pre-indexing Folder optimization	-	X

Although it is FREE...



- PTK Forensics uses a **centralized database** for case management; thus, **more investigators can work simultaneously on the same case** from different computers.
- With the Indexing Engine the administrator can perform preliminary operations and this result can be used by every investigator associated with a case:
 - **Timeline generation**
 - **File categorization**
 - **MD5, SHA1**
 - **Keyword indexing**



Advanced Tabular Timeline (free version)



Why are **timeline analysis/file timestamps** so important?

- It can be used by an investigator to gain insight of what happened and the people involved: who was logged when event “x” happened?

- It can be used to identify anomalies: how come we had 1000 failed logins after working hours?

- It can be used to reconstruct the sequence of events

Select partition: All
Start date: 2002-01-02 08:00:00
End date:
Create timeline
Choose timeline type: table graphic

Show 100 rows

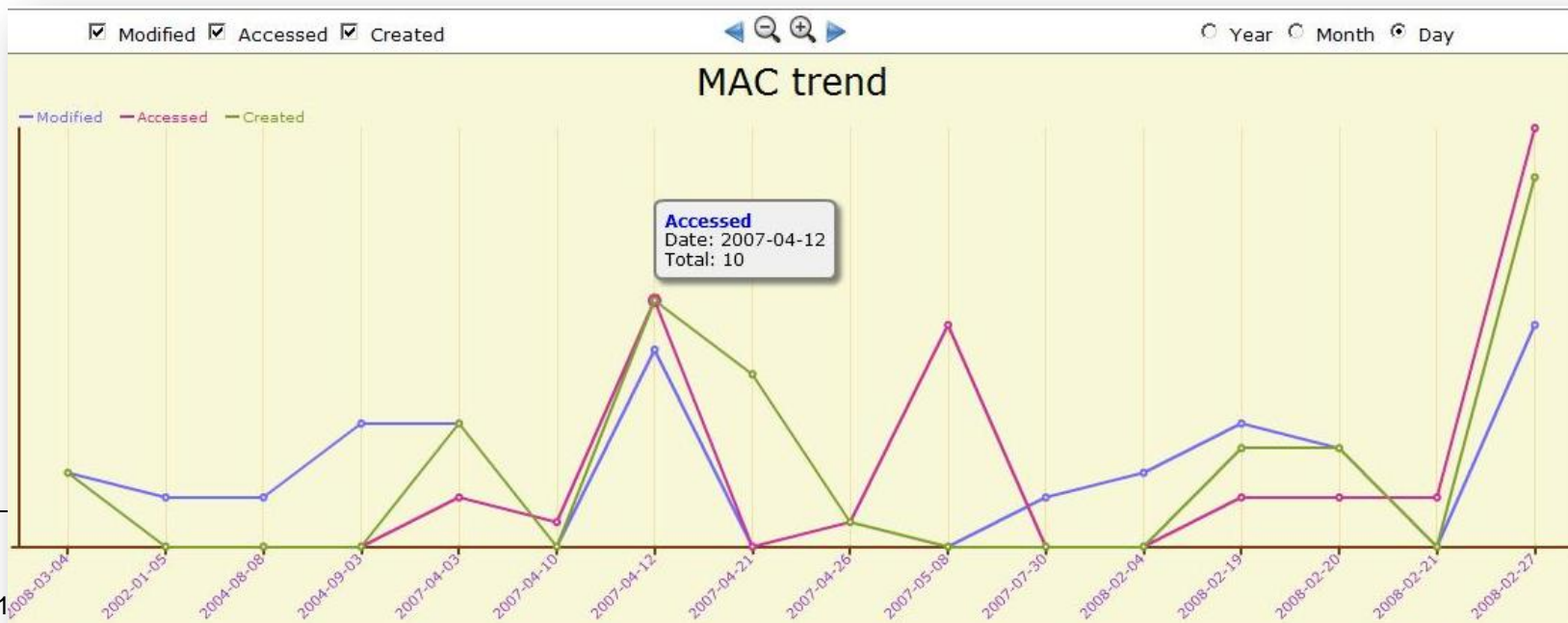
If selected: [bookmark all](#)

	Date-time	File name	Action	Size	Permissions
<input type="checkbox"/>	2002-01-05 15:37:00	fat16/_svcr70.dll (deleted)	m..	344064	-/-rwxrwxrwx
<input type="checkbox"/>	2002-01-05 15:40:00	fat16/_SVCP70.DLL (deleted)	m..	487424	-/-rwxrwxrwx
<input type="checkbox"/>	2004-08-08 22:17:06	fat16/fpdns.pl (deleted)	m..	29454	-/-rwxrwxrwx
<input type="checkbox"/>	2004-08-08 23:21:28	fat16/fpdns.1 (deleted)	m..	5988	-/-rwxrwxrwx
<input type="checkbox"/>	2004-09-03 19:34:46	fat16/_etopt.dll (deleted)	m..	9216	-/-rwxrwxrwx
<input type="checkbox"/>	2004-09-03 22:34:46	fat16/_dSsum.exe (deleted)	m..	17408	-/-rwxrwxrwx
<input type="checkbox"/>		fat16/_dSlib.dll (deleted)	m..	14848	-/-rwxrwxrwx
<input type="checkbox"/>		fat16/zlibU.dll (deleted)	m..	51200	-/-rwxrwxrwx
<input type="checkbox"/>	2007-04-03 13:12:00	fat16/_zip.exe (deleted)	m..	68096	-/-rwxrwxrwx
<input type="checkbox"/>	2007-04-10 00:00:00	fat16/PsTools (deleted)	.a.	0	d/drwxrwxrwx

Advanced Graphic Timeline (free version)



- This is a graphic that shows the trend of each type of action (among the three of the MAC time), distributed over the entire period or over the **selected time interval**.
- Provides a useful instrument in order to **visualize peaks of access/modification/creation to files**.
- Investigators can apply filters (i.e. show only modified event, accessed event or created event).



2009/10 New Features: Data Carving

(currently full version only, will be the first to scale to the free version by the end of Q3)



PTK Forensics uses the technique called “zero storage”;

The tool uses Scalpel, which originally export on file system every file identified during the data carving phase, using the `-p` switch so just a reference is stored, without allocating new disk space.

This modality enables users to **run the data carving process without having to allocate the physical space on the disk**; saving instead, for every recognized file, its own reference inside the disk (start sector and offset);

2009/10 New Features: Data Carving

(currently full version only)



Types of files that are carved:

Graphics Multimedia Documents Mails Archives Other Custom

Extension	Header	Footer
<input type="checkbox"/> art	\x4a\x47\x04\x0e	\xc7\xc7\xcb
<input type="checkbox"/> art	\x4a\x47\x03\x0e	\xd0\xcb\x00\x00
<input type="checkbox"/> bmp	BM??\x00\x00\x00	
<input type="checkbox"/> gif	\x47\x49\x46\x38\x39\x61	\x00\x00\x3b
<input type="checkbox"/> gif	\x47\x49\x46\x38\x37\x61	\x00\x3b
<input type="checkbox"/> jpg	\xff\xd8	\xff\xd9
<input checked="" type="checkbox"/> jpg	\xff\xd8\xff\xe1	\xff\xd9
<input checked="" type="checkbox"/> jpg	\xff\xd8\xff\xe0\x00\x10	\xff\xd9
<input type="checkbox"/> png	\x50\xe4e\x47?	\xff\xfc\xfd\xfe

Save

Customized file signature

Predefined file signature

Types of files that are carved:

Graphics Multimedia Documents Mails Archives Other Custom

Add file type info:

Extension:

Size:

Case sensitive:

Header*:

Footer*:

* hex format \x0-9A-Fa-f

Add

Extension	Header	Footer
<input type="checkbox"/> ptk	\x47\x49\x46\x38\x61	\xa6\x00

Save

2009/10 New Features: Data Carving example

(currently full version only)



At the end of the process, the investigator can choose to export only those files which are of major interest.

- It is automatically integrated with the PTK Forensics (File analysis).

The screenshot displays a forensic analysis tool interface. On the left is a file tree with folders like 'jpeg_image', 'ntfs', '\$Extend', 'alloc', 'archive', 'del1', 'del2', 'invalid', 'misc', 'RECYCLER', 'System Volume Information', '\$OrphanFiles', and 'Carved files'. The main area shows a table of analyzed files:

File analysis	Timeline	Keyword	Gallery	Hashset	Image details	Data unit	Bookmark	Reports	[X] Close
		<input type="checkbox"/>	<input type="checkbox"/>	00000013.jpg	1070564	69299	NO	Carving_case_jpeg_image.001	
		<input type="checkbox"/>	<input type="checkbox"/>	00000012.jpg	1068965	70898	NO	Carving_case_jpeg_image.001	
		<input type="checkbox"/>	<input type="checkbox"/>	00000011.jpg	1065620	74243	NO	Carving_case_jpeg_image.001	
		<input type="checkbox"/>	<input type="checkbox"/>	00000010.jpg	1060595	79268	NO	Carving_case_jpeg_image.001	
		<input type="checkbox"/>	<input type="checkbox"/>	00000009.jpg	872960	175630	NO	Carving_case_jpeg_image.001	
		<input type="checkbox"/>	<input type="checkbox"/>	00000008.jpg	545792	326859	NO	Carving_case_jpeg_image.001	
		<input type="checkbox"/>	<input type="checkbox"/>	00000007.jpg	271360	274260	NO	Carving_case_jpeg_image.001	
		<input type="checkbox"/>	<input type="checkbox"/>	00000006.jpg	6442825	110373	NO	Carving_case_jpeg_image.001	
		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	00000005.jpg	6166564	271181	NO	Carving_case_jpeg_image.001	

Below the table, the selected file '00000005.jpg' is shown in a preview window. It includes tabs for 'Ascii', 'Hex', 'Ascii strings', 'Preview', 'Export', and 'Bookmark'. The preview text reads 'JPEG image data, JFIF standard 1.01' and shows a blue-tinted image of a cloud with the text 'I Am Pictures HD'.

2009/10 New Features: Plug-in system (currently full version only)



Why is the plug-in system so important?

- Every investigator can **extend PTK Forensics' features** according to his needs
- The output of a single plug-in can be exported and saved as bookmark for further analyses;

Investigators Logging Hashset **Plug-in**

Plug-in

[Add new plug-in]

*tool:

description:

*base command:

Create new parameter ✕

*Name:

*Parameter:

Output type: shell output
 binary file


Comment:


add


2009/10 New Features: Plug-in system example (currently full version only)




Currently PTK forensics supports the plug-in for the **registry analysis (RegRipper)** and also the **windows artifact analysis;**

Tool: 

Description: 

Base command: 

 Add

























Output type	Name	Parameters	Comments	Operations
	USB	-p usb		
	Mount device	-p mountdev		
	Timezone	-p timezone		
	Network	-p network		
	Company Name	-p compname		
	Shares	-p shares		
	Shutdown	-p shutdown		

image001

- ntfs
 - \$Extend
 - Elvis Admin
 - Secret User
 - System Volume Information
 - WINDOWS
 - System32
 - config
 - \$OrphanFiles

Total listed files:20 | Total filtered files:0 | Total bookmarked files:0

<input type="checkbox"/>		Name	Modified	Accessed	Changed	Birth	Size	UID	GID	Meta	Hashset
<input type="checkbox"/>		AppEvent.Evt	2003-12-01 00:40:10 (CET)	2003-12-20 04:57:29 (CET)	2003-12-20 05:14:22 (CET)	2003-12-20 04:57:29 (CET)	65536	0	0	489-128-3	
<input type="checkbox"/>		default	2003-12-01 00:40:15 (CET)	2003-12-20 04:57:29 (CET)	2003-12-20 05:14:22 (CET)	2003-12-20 04:57:29 (CET)	262144	0	0	490-128-3	
<input type="checkbox"/>		default.LOG	2003-12-01 03:05:26 (CET)	2003-12-20 04:57:29 (CET)	2003-12-20 05:14:22 (CET)	2003-12-20 04:57:29 (CET)	1024	0	0	491-128-3	
<input type="checkbox"/>		default.sav	2003-08-29 11:42:26 (CEST)	2003-12-20 04:57:30 (CET)	2003-12-20 05:14:22 (CET)	2003-12-20 04:57:29 (CET)	94208	0	0	492-128-3	
<input type="checkbox"/>		SAM	2003-12-01 02:51:22 (CET)	2003-12-20 04:57:30 (CET)	2003-12-20 05:14:22 (CET)	2003-12-20 04:57:30 (CET)	262144	0	0	493-128-3	
<input type="checkbox"/>		software.LOG	2003-11-28 03:27:25 (CET)	2003-12-20 04:57:48 (CET)	2003-12-20 05:14:32 (CET)	2003-12-20 04:57:48 (CET)	1024	0	0	499-128-3	
<input type="checkbox"/>		software.sav	2003-08-29 11:42:26 (CEST)	2003-12-20 04:57:48 (CET)	2003-12-20 05:15:24 (CET)	2003-12-20 04:57:48 (CET)	626688	0	0	500-128-3	
<input type="checkbox"/>		SysEvent.Evt	2003-12-01 00:40:10 (CET)	2003-12-20 04:57:49 (CET)	2003-12-20 05:15:27 (CET)	2003-12-20 04:57:48 (CET)	327680	0	0	501-128-3	
<input checked="" type="checkbox"/>		system	2003-12-01 02:28:50 (CET)	2003-12-20 04:57:52 (CET)	2003-12-20 05:15:27 (CET)	2003-12-20 04:57:49 (CET)	3145728	0	0	502-128-3	

2009/10 New Features: Plug-in system example

(currently full version only)



The PTK Forensics development team tested and validated several plug-ins in order to increase PTK capabilities:

- **Volatility Framework (already included also in the free version);**
- **Pasco;**
- **Rifiuti;**
- **Galleta.**

```
EVILBOX,ROOT_HUB,4&28279229&0,1070141998,
, ,5&164792fb&0
EVILBOX,ROOT_HUB,4&3ae142ce&0,1070141998,
, ,5&4f7f8b2&0
EVILBOX,ROOT_HUB20,4&f98c74f&0,1070141998,
,
EVILBOX,Vid_0000&Pid_0000,5&4f7f8b2&0&2,106E
```

```
History File: /opt/lampp/htdocs/ptk/temp/test
Version: 5.2 TYPE URL MODIFIED TIME ACCESS
TIME FILENAME DIRECTORY HTTP HEADERS URL
Visited: Secret User@file:///C:
/Documents%20and%20Settings/Secret%20User
/Desktop/EFS%20FILE%201.txt 12/20/2003
05:27:48 12/20/2003 05:27:48 URL Visited:
Secret User@file:///C:
```

Users are required to decide which plug-in must be included

Application settings

Investigators | Logging | Hashset | **Plug-in**

Plug-in

[Add new plug-in]
Galleta
Pasco
RegRipper

Tool: Pasco

Description: pasco options filename d Undelete Activity Records t Field Delimiter TAB by default

Base command: pasco/pasco \$filename

Output type	Name	Parameters
	Field Delimiter	-t
	No parameters	

2009/10 New Features: Hashset

(currently full version only)



Why is the **hashset** so important?

- The hashset analysis can be used, for example, in cases of copyright infringement where the aim of the investigation is to identify possible projects that two companies may have in common (and one has been stolen)
- In order to hide files that are not relevant to the investigation (i.e. Operating system files or well-known program files)
- Users can create their own hash sets and or include external ones

Application settings

Investigators | Logging | **Hashset** | Plug-in

Hashset

[Import from file] | add row(s) manually

[Add new hashset]

Suspect Evidence

<input type="checkbox"/>	Category [new]	Filename	Hash MD5	Hash SHA1	Comments	Operation
<input type="checkbox"/>	Suspect Evidence	suspect project	2dfc817f62df8276f0947314af8ab931			

if selected set | | | |

save hashset

2009/10 New Features: Hashset

(currently full version only)



It is possible to create, for example, three different hash sets (such as INFECTED, SYSTEM, STOLEN) giving each of them a name, description, a particular colour, and manually inserting the entries with the hashes to search inside the evidence.

Category	Description	Active
Example		●
INFECTED		●
project		●
STOLEN		●
SYSTEM		●

2009/10 New Features: Hashset example

(currently full version only)



Total listed files:29 | Total filtered files:0 | Total bookmarked files:0

<input type="checkbox"/>	★	Name	Modified	Accessed	Changed	Birth	Size	UID	GID	Meta	Hashset
<input type="checkbox"/>	★	dots_off.jpg	2003-09-27 01:45:22 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	3683	0	0	227-128-4	INFECTED
<input type="checkbox"/>	★	50_off.jpg	2003-09-27 01:45:16 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	4388	0	0	217-128-4	INFECTED
<input type="checkbox"/>	★	Am Ex Logo.jpg	2003-09-27 01:42:18 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	1659	0	0	218-128-4	STOLEN STOLEN
<input type="checkbox"/>	★	Amex Holo.jpg	2003-09-27 01:42:24 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	6482	0	0	219-128-4	STOLEN STOLEN
<input type="checkbox"/>	★	Amex Login Template.jpg	2003-09-27 01:44:30 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	10133	0	0	220-128-4	STOLEN STOLEN
<input type="checkbox"/>	★	BEWARE !!.jpg	2003-09-27 01:47:54 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	4303	0	0	221-128-4	
<input type="checkbox"/>	★	Blue Template.bmp	2003-09-27 01:43:50 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	16438	0	0	222-128-4	
<input type="checkbox"/>	★	blurry but good.jpg	2003-09-27 01:34:06 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	2416	0	0	223-128-4	
<input type="checkbox"/>	★	CCG1.GIF	2003-09-27 00:54:20 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	32049	0	0	224-128-3	SYSTEM
<input type="checkbox"/>	★	CCG2.GIF	2003-09-27 00:54:14 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	4428	0	0	225-128-3	SYSTEM
<input type="checkbox"/>	★	CCG3.GIF	2003-09-27 00:54:16 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	4553	0	0	226-128-3	SYSTEM
<input type="checkbox"/>	★	dots_off.jpg	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0000-00-00 00:00:00 (UTC)	0	0	0	0	
<input type="checkbox"/>	★	dreamin.jpg	2003-09-27 01:31:22 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	3650	0	0	228-128-3	
<input type="checkbox"/>	★	fake ids.jpg	2003-09-27 01:32:14 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	3586	0	0	229-128-4	
<input type="checkbox"/>	★	harry.jpg	2003-09-27 01:31:50 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	4073	0	0	230-128-3	INFECTED
<input type="checkbox"/>	★	High Dollar Purchase.jpg	2003-09-27 01:47:20 (CEST)	2003-12-01 03:00:45 (CET)	2003-12-20 05:51:40 (CET)	2003-12-01 02:54:22 (CET)	4162	0	0	231-128-4	
<input type="checkbox"/>	★	it passed L.jpg	2003-09-27	2003-12-01	2003-12-20	2003-12-01	4477	0	0	232-128-4	

2009/10 New Features: Data Interpreter

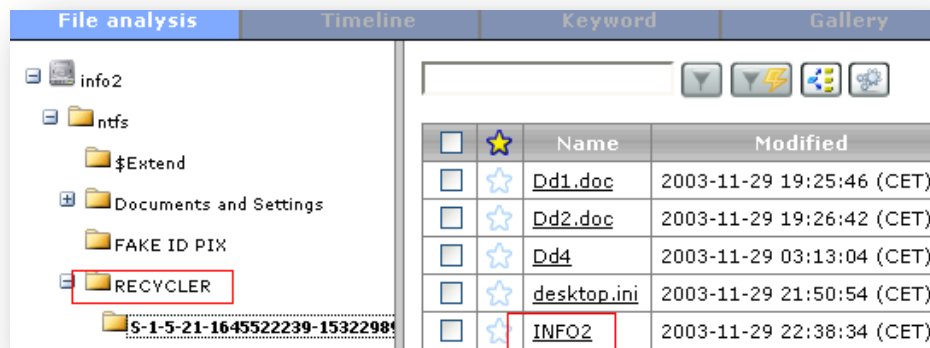
(currently full version only, via Php PTK Implementation)



Why is the data interpreter so important?

- Computer forensic tools are used to parse and interpret data correctly. Data can be found in different formats:

Size	Value
8-BIT INTEGER (SIGNED AND UNSIGNED)	BINARY DIGITS
16-BIT INTEGER (SIGNED AND UNSIGNED)	WINDOWS TIMESTAMP
32-BIT INTEGER (SIGNED AND UNSIGNED)	DATE DOS
64-BIT INTEGER (SIGNED AND UNSIGNED)	TIMESTAMP DOS
FLOAT REAL DOUBLE	UNIX DATE



2009/10 New Features: Data Interpreter

(currently full version only)



It's available in data analysis tab and converts on the fly data in HEX to common type format. The implementation is made via PTK own source code.

INFO2 file analysis

Folder Details Hex value interpreter	
<input checked="" type="radio"/> little endian <input type="radio"/> big endian	
type	value
signed integer	5120
Filetime (utc)	
Filetime (local)	
Date (DOS)	
Time (DOS)	
Time_t (utc)	1/1/1970 1:25:20
Time_t (local)	1/1/1970 2:25:20

Folder Details Hex value interpreter	
<input checked="" type="radio"/> little endian <input type="radio"/> big endian	
type	value
signed integer	127146148238750000
Filetime (utc)	29/11/2003 21:27:3
Filetime (local)	29/11/2003 22:27:3
Date (DOS)	
Time (DOS)	
Time_t (utc)	
Time_t (local)	

```
00000120 30 71 54 88 bf b6 c3 01 00 14 00 00 44 00 3a 00 |0qT.....D...|
00000130 5c 00 44 00 6f 00 63 00 75 00 6d 00 65 00 6e 00 |\.D.o.c.u.m.e.n.|
00000140 74 00 73 00 20 00 61 00 6e 00 64 00 20 00 53 00 |t.s. .a.n.d. .S.|
00000150 65 00 74 00 74 00 69 00 6e 00 67 00 73 00 5c 00 |e.t.t.i.n.g.s.\.|
00000160 42 00 61 00 64 00 20 00 47 00 75 00 79 00 20 00 |B.a.d. .G.u.y. .|
00000170 32 00 4b 00 5c 00 44 00 65 00 73 00 6b 00 74 00 |2.K.\.D.e.s.k.t.|
00000180 6f 00 70 00 5c 00 4e 00 54 00 46 00 53 00 20 00 |o.p.\.N.T.F.S. .|
00000190 52 00 65 00 63 00 79 00 63 00 6c 00 65 00 64 00 |R.e.c.y.c.l.e.d.|

00000110 00 00 00 00 00 00 00 00 01 00 00 00 03 00 00 00 |.....|
00000120 30 71 54 88 bf b6 c3 01 00 14 00 00 44 00 3a 00 |0qT.....D...|
00000130 5c 00 44 00 6f 00 63 00 75 00 6d 00 65 00 6e 00 |\.D.o.c.u.m.e.n.|
00000140 74 00 73 00 20 00 61 00 6e 00 64 00 20 00 53 00 |t.s. .a.n.d. .S.|
00000150 65 00 74 00 74 00 69 00 6e 00 67 00 73 00 5c 00 |e.t.t.i.n.g.s.\.|
00000160 42 00 61 00 64 00 20 00 47 00 75 00 79 00 20 00 |B.a.d. .G.u.y. .|
00000170 32 00 4b 00 5c 00 44 00 65 00 73 00 6b 00 74 00 |2.K.\.D.e.s.k.t.|
```

2009/10 New Features: Job Manager

(currently full version only)

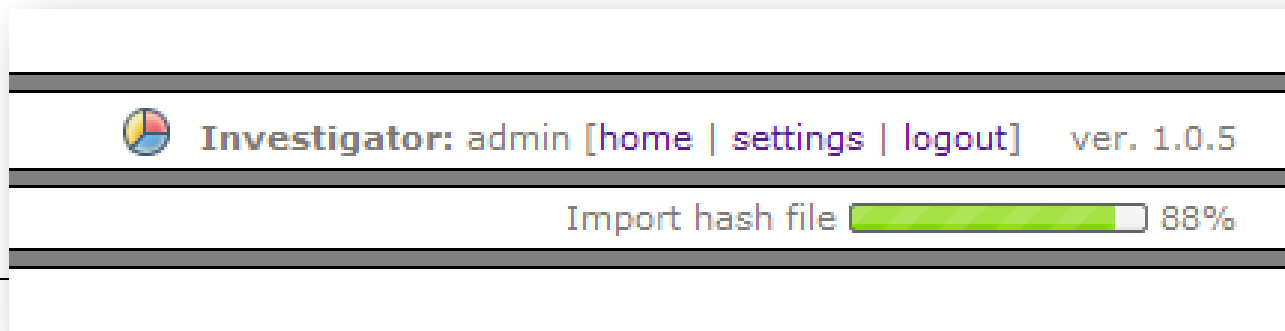


In order to improve and make PTK Forensics easier to use we have inserted the first version of PTK Forensics **Job Manager**.

Why is the PTK Forensics Job Manager so important?

- It allows to insert the concept “**thread like**” (**cannot be called multithreading because PHP structure do not permit it**); However, during the indexing activity, user can still work on the evidence-

It allows to monitor the status of import and indexing operations in real time.

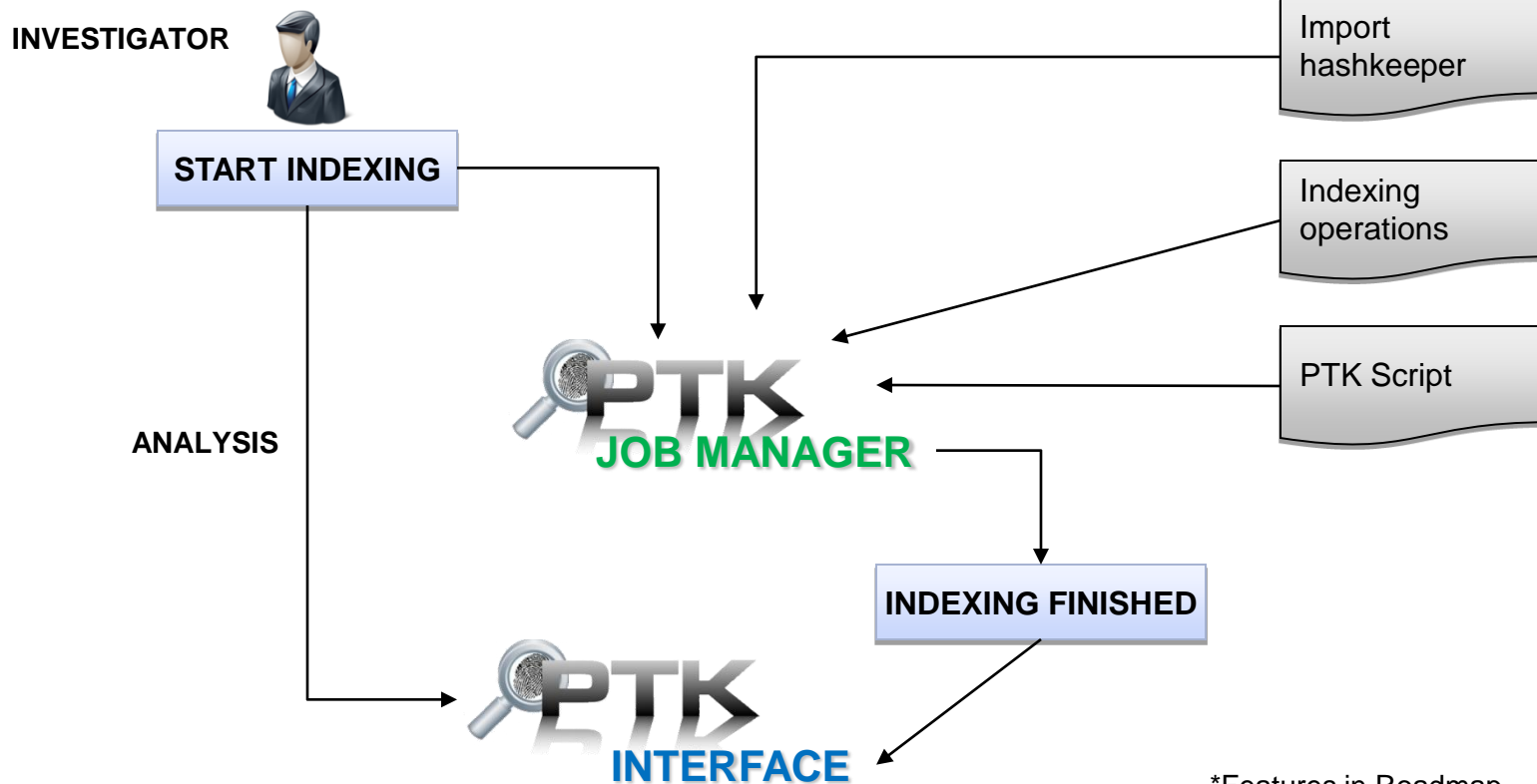


2009/10 New Features: Job Manager

(currently full version only)



Now it is possible to start working while indexing tasks are running.



*Features in Roadmap

Current Status: Indexing engine, main features



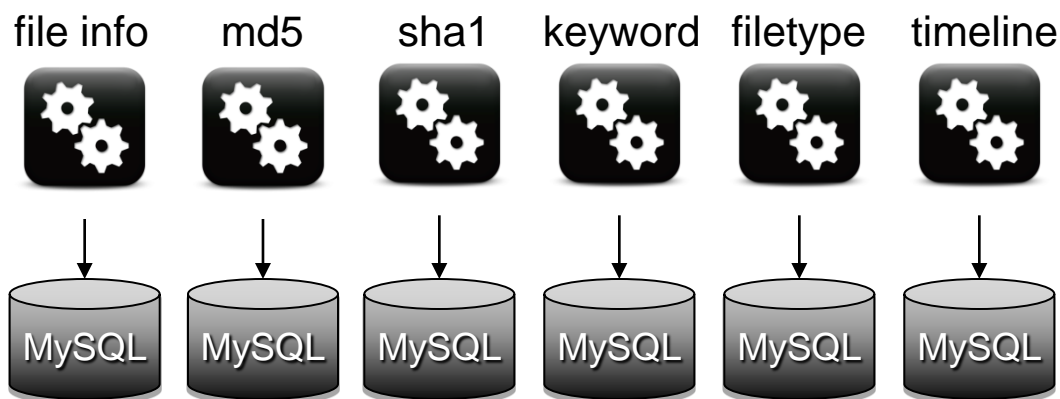
- **String extraction** (Ascii&Unicode) from the space:
 - **Allocated;**
 - **Un allocated.**
 - **Slack (NTFS and FAT);**
- Identification of **the know(n) good and the know(n) bad** (Hashset libraries); (Full version only)
- File Signature analysis;
- **File categorization** (graphics, documents, executables, etc.);
- Metadata and hash generation of the files present on the evidence;
- **Timeline generation;**
- **Data carving**, zero-storage technique; (Full Version only)
- The results of the preliminary operations are stored into the database for a better and faster interrogation/inquiry.

Indexing engine, First Version (2008)



First engine (Sequential approach, based upon several icat instances)

- String Extraction, Timeline generation, Hash of all files, Categorization and Keyword search;

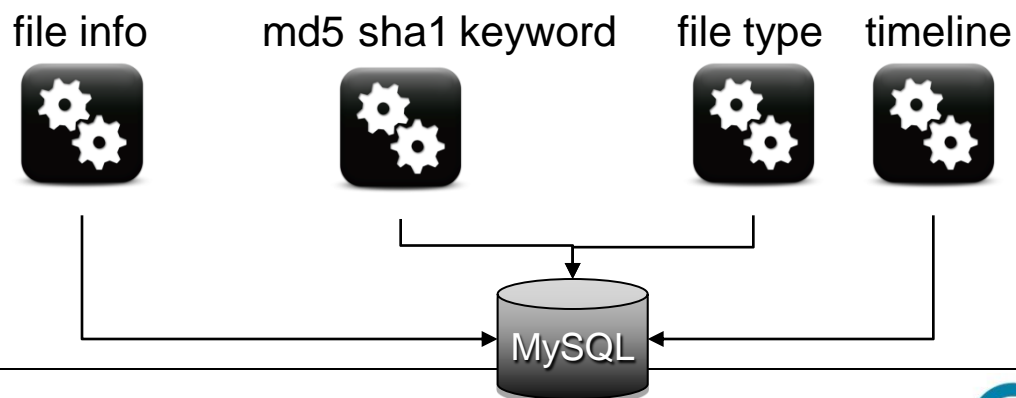


Indexing engine 1.0 (First Stable Version 2009)



Thread engine

- Besides previous operations, Data Carving was also inserted;
- Optimized use of the icat command: the icat output is used to generate in one shot Md5, SHA1 and Keyword list.
- Reduced number of queries towards MySQL

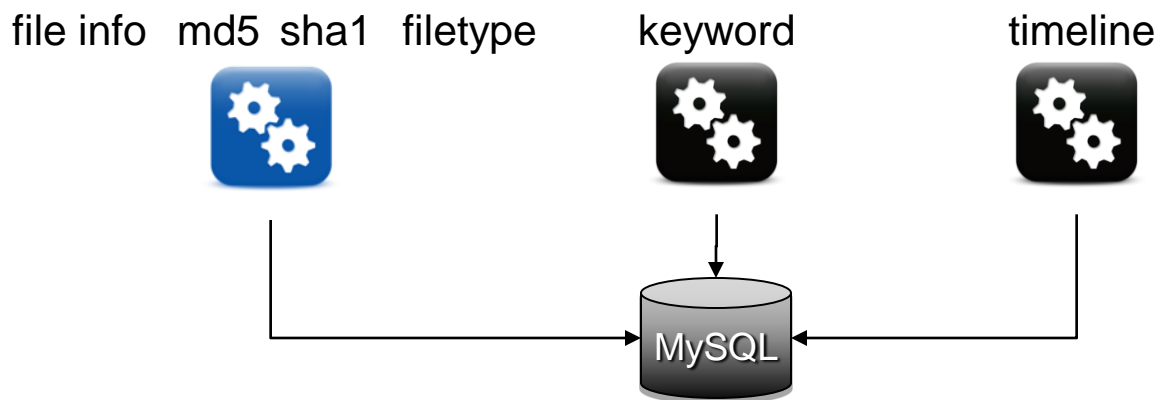


New indexing engine, v2.0



TSK API Based Engine

- API: allow the interface with evidence without using external tools such as TSK
- Performance increase;



Indexing engine: benchmark



PTK Engine v. 1.0

- Indexing options:
 - File info;
 - File type flag*;
 - MD5, SHA1.

PTK Engine v. 2.0

- Indexing options:
 - File info;
 - File type flag*;
 - MD5, SHA1.

FTK 1.81

- Indexing options:
 - File info;
 - File type flag*;
 - MD5, SHA1.

*The benchmark were carried out on the following evidence:

- 1,2 Gb (raw) – Filesystem: Ext2
- 46 Gb (E01) – Filesystem: NTFS

HW TECHNICAL SPECIFICATIONS

- Processor Quad Core Xeon X3323, 2.5GHz,
- 4GB DDR2 667MHz Memory

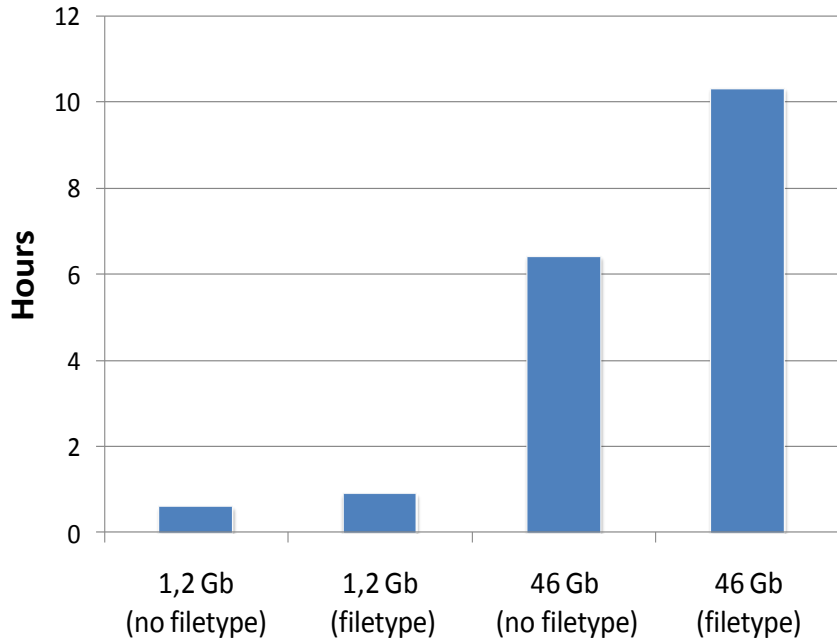
*We estimated both situations: enabling and disabling the file type flag for both products;

The file type option allows file categorization based on type (Documents, Graphics, Audio, etc.)

PTK Engine v1.0 Vs PTK Engine v2.0

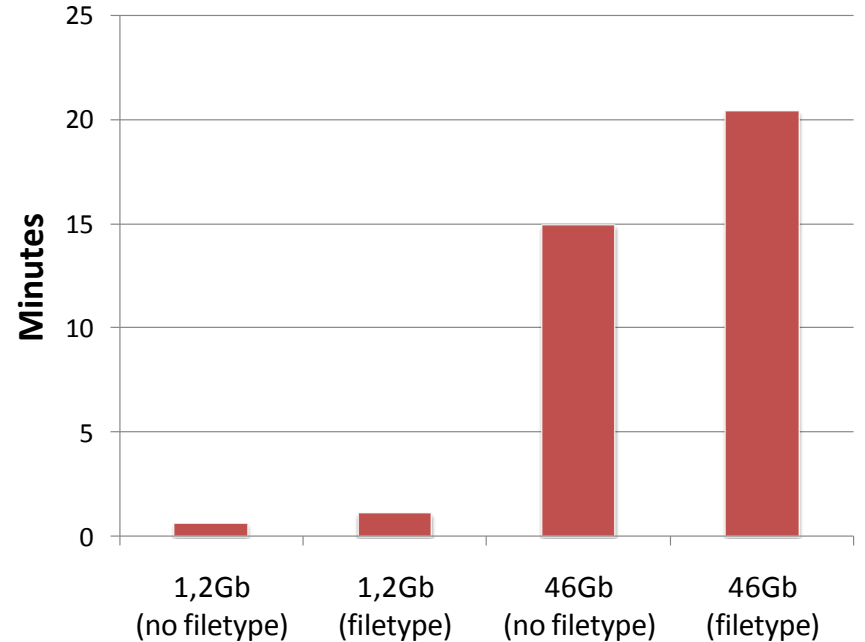


PTK Engine v1.0



Computation in hours

PTK Engine v2.0



Computation in minutes

The performance increase was possible thanks to inserting the API of TSK. The timetable indicated include the analysis and the insertion in the respective database.

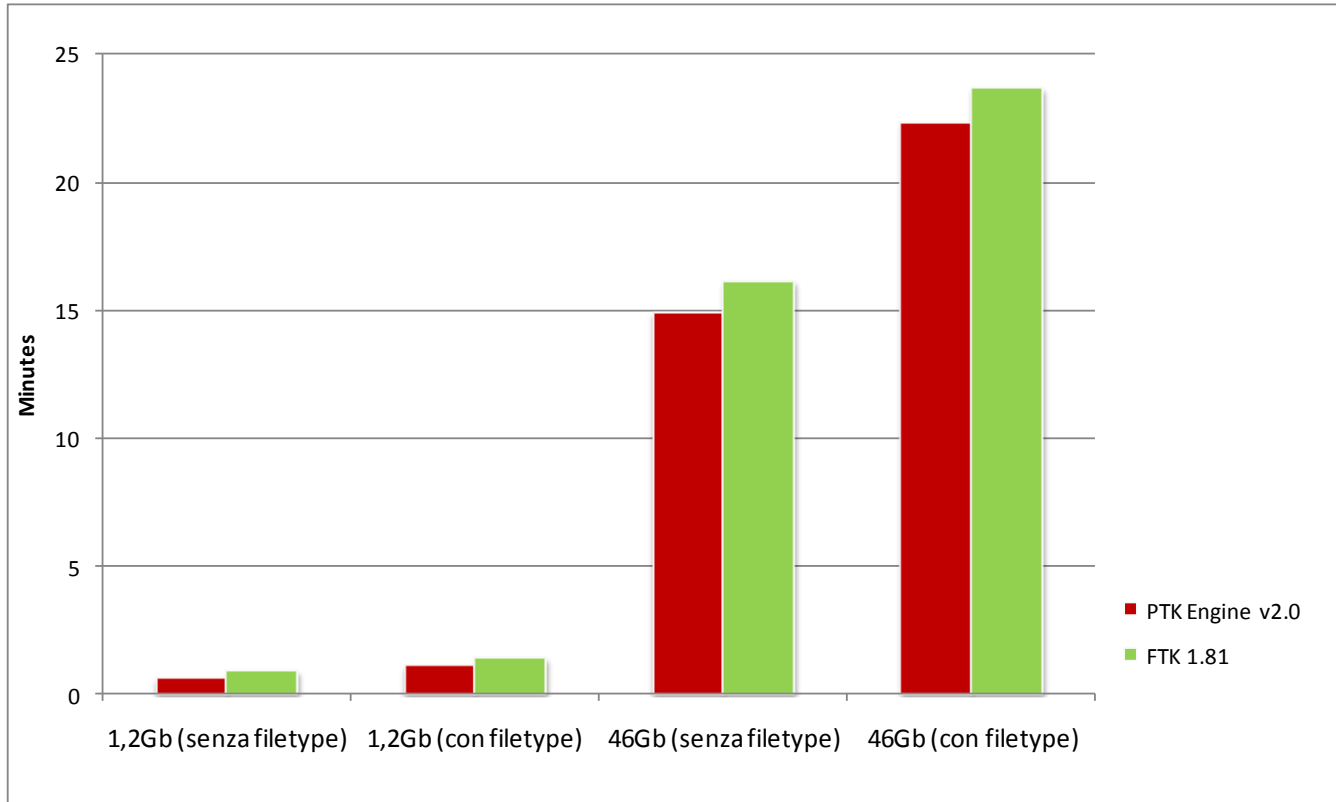
PTK Engine v1.0 Vs PTK Engine v2.0



Benchmark table:

Evidence	PTK Engine v1.0	PTK Engine v2.0
12 Gb (no filetype)	6 minutes	6 seconds
12 Gb (filetype)	9 minutes	1 minutes 1seconds
46 Gb (no filetype)	6 hours 4 minutes	14 minutes 8 seconds
46 Gb (filetype)	10 hours 3 minutes	20 minutes 34 seconds

PTK Engine v2.0 Vs FTK 1.81



The timetable indicated include the analysis and the insertion in the respective database.

FTK 2.x and 3.x being tested.

Integrating PTK with external tools: IncMan Suite



DFLabs Incman Suite, is an incident tracking software that enables the management of every kind of information security incidents.

Incident Management Software supports the entire incident management process from security to fraud, ***including digital forensics, case management, evidence and incident tracking.*** IncMan, Incident Management Suite, supports all certification and accreditation processes required by sections 3505 and 3544 of the Federal Information Security Management Act (FISMA), as well as the ability to report and manage incidents associated with government facilities and systems.

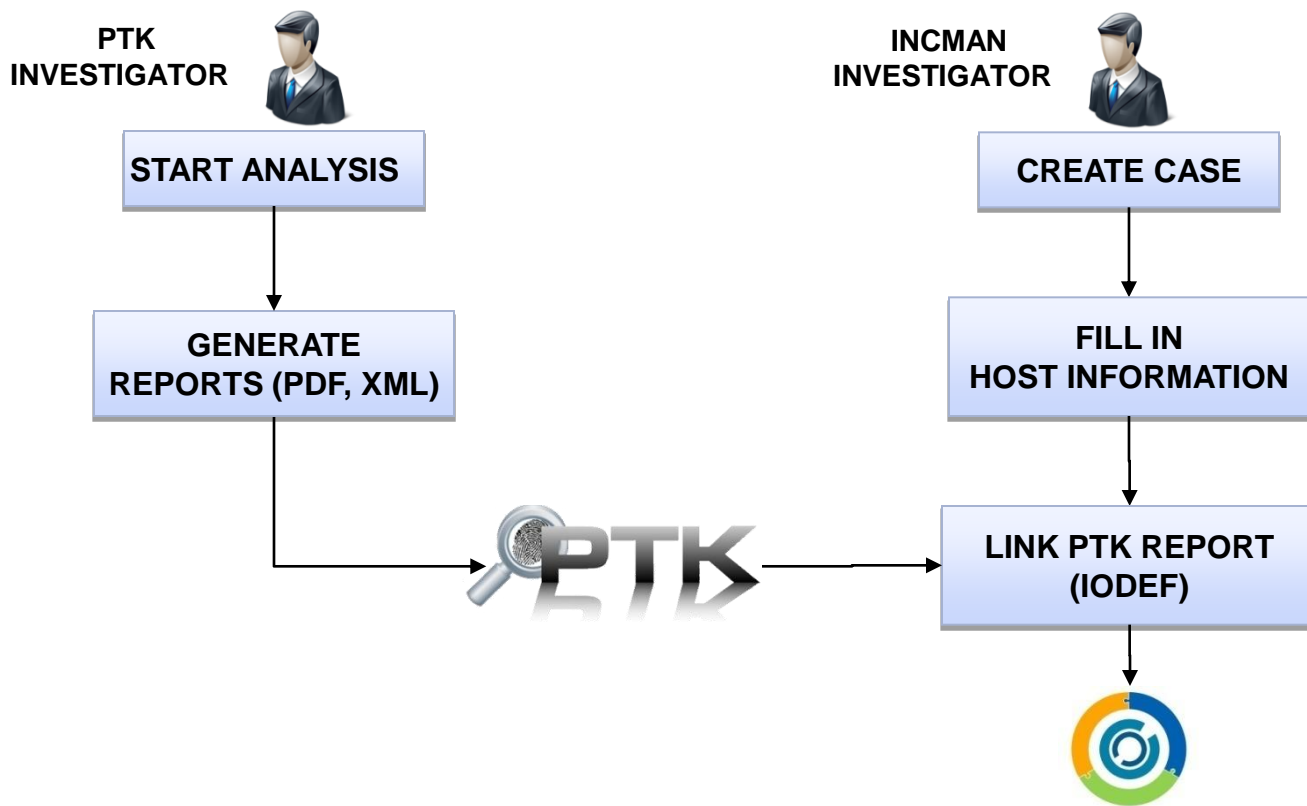
More than 100 enterprise customers worldwide, including the DIM Module.



PTK Forensics with IncMan Suite



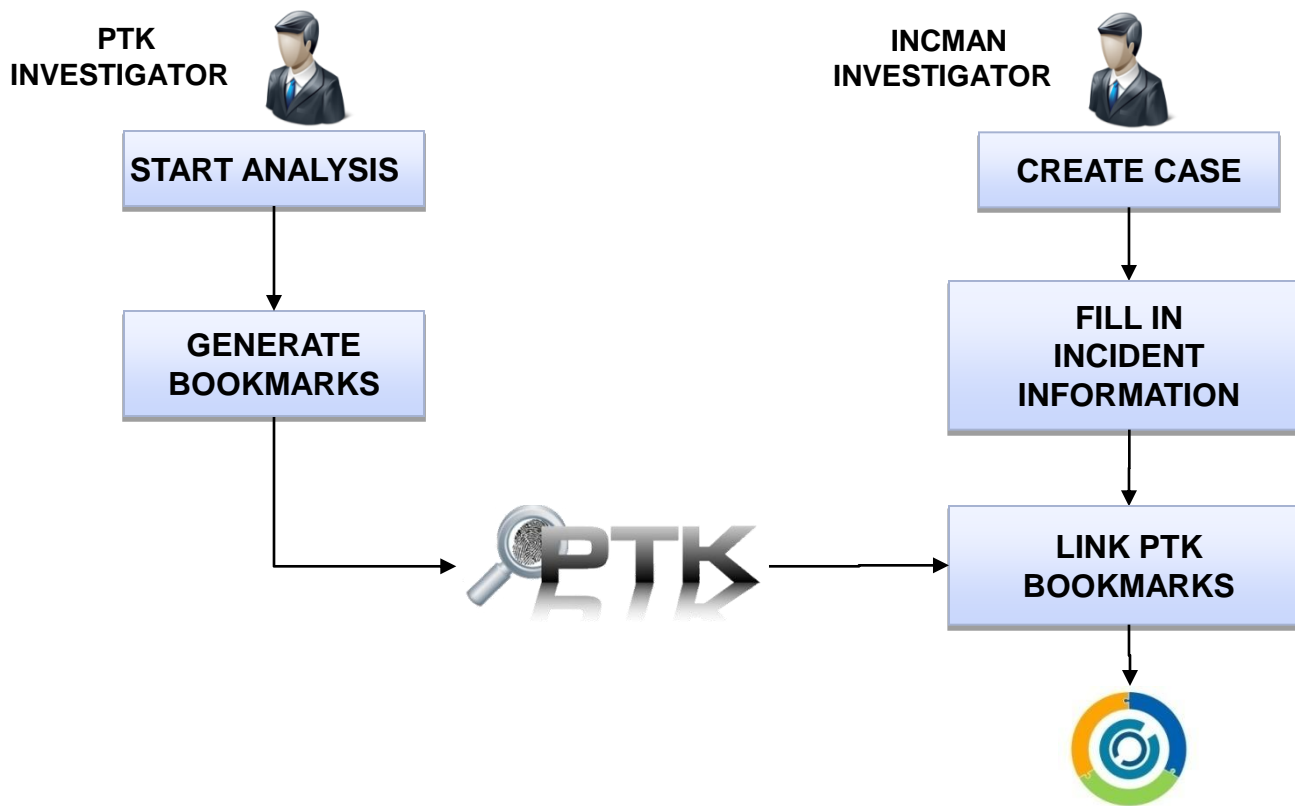
IncMan Suite allows the complete integration with PTK Forensics. It is thus possible to include the analysis report among the information regarding the host. There is also a module, called DIM, which allows investigators to track their digital evidences and manage their forensic cases.



PTK Forensics with IncMan Suite



In the final incident report it is possible to include the bookmarks saved by each investigator.



Where will be: 12months Further steps



PTK Forensics roadmap:

- **Background processing:** improve the job management system.
- **Recursive export:** enable the recursive support of a folder or of several files; (Free version)
- **Report form:** allow the dynamic generation and custom report. Integration with DIM – Incman Suite -
- **Metadata extraction:** file office, pdf, exif: add further information in the file analysis section.
- **PTK Script:** allow plug-in in order to automatically extract information from the evidence.
- **Timeline Filter:** improve the graphic representation with the possibility to apply complex filters.
- **Mail Archive Analysis:** allow the analysis of the best known email formats through a dedicated section and a user-friendly interface.
- **More external tools integration** both opensource and commercial

Conclusions



PTK Forensic is the demonstration that OpenSource, Free and Commercial Software can co-exist

We will be happy to cooperate with both communities to continue to build a computer forensic framework based upon open source and commercial software, able to valorize both sides of the industry

This will give to the community chance to approach forensics with lower budgets. In fact:

- Free version will constitute the basic level for starting to work with Digital forensics and will be kept updated thanks to the FIFO Shifting Paradigm
- Full version will give an advanced framework at very competitive price

Thanks!





Internet: www.dflabs.com

EMAIL: info@dflabs.com

Youtube: youtube.com/dflabs