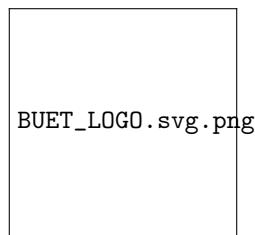


CSE 406
Computer Security Sessional

**Documentation Report
of
AUTOPSY**

Abdullah Nayem Wasi Emran
1905034

Rakib Abdullah
1905047



BUET_LOGO.svg.png

**Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology**

1 An Introduction to Autopsy

1.1 Introduction

Autopsy stands as a dynamic digital forensic solution, aiding investigators in dissecting and managing evidence seamlessly. Renowned for its user-friendly interface and robust functionalities, Autopsy simplifies the intricate processes of forensic analysis. It offers comprehensive assistance throughout investigations, earning the trust of law enforcement, military personnel, and corporate examiners alike.

1.2 Capabilities of Autopsy

Autopsy offers a wide range of functionalities, including:

- File investigation
- Keyword search
- Archive parsing
- Hash filtering
- Data integrity checking
- Data recovery
- Picture examination and image analysis
- Database exploration and analysis
- Registry analysis
- Email exploration
- Malware detection
- Event viewing
- History review
- Bookmarking
- Report creation

1.3 Sequential Steps in Autopsy Investigation

The steps in Autopsy for analyzing a data source involves the following steps:

1. **Case Creation:** Start by creating a case, which acts as a container for data sources and reports.
2. **Data Source Addition:** Add one or more data sources to the case.

3. **Ingest Module Execution:** Run ingest modules on the data source. These modules work in the background to analyze the data.
4. **Manual Analysis:** Conduct manual analysis by navigating through the data and reviewing the results from ingest modules to identify evidence.
5. **Report Generation:** Generate a final report based on selected tags or results.

1.4 Managing Cases and Data Sources

Autopsy works by organizing cases and their related data sources. Each case needs at least one data source for analysis. Types of data sources supported in Autopsy include:

- Disk images or virtual machine files
- Local disks
- Logical files
- Files from unallocated space
- Autopsy Logical Imager results
- XRY Text Export

To analyze data using Autopsy, start by creating a case and adding the data source(s). Each data source is linked to a host. Then, select which modules to run on the data source. The results are stored in the case for analysis, and we can generate reports based on these results.

1.5 Ingest Modules

Ingest Modules are features that examine the information in the data sources. They thoroughly check files and understand what they contain. Autopsy offers various Ingest Modules, including:

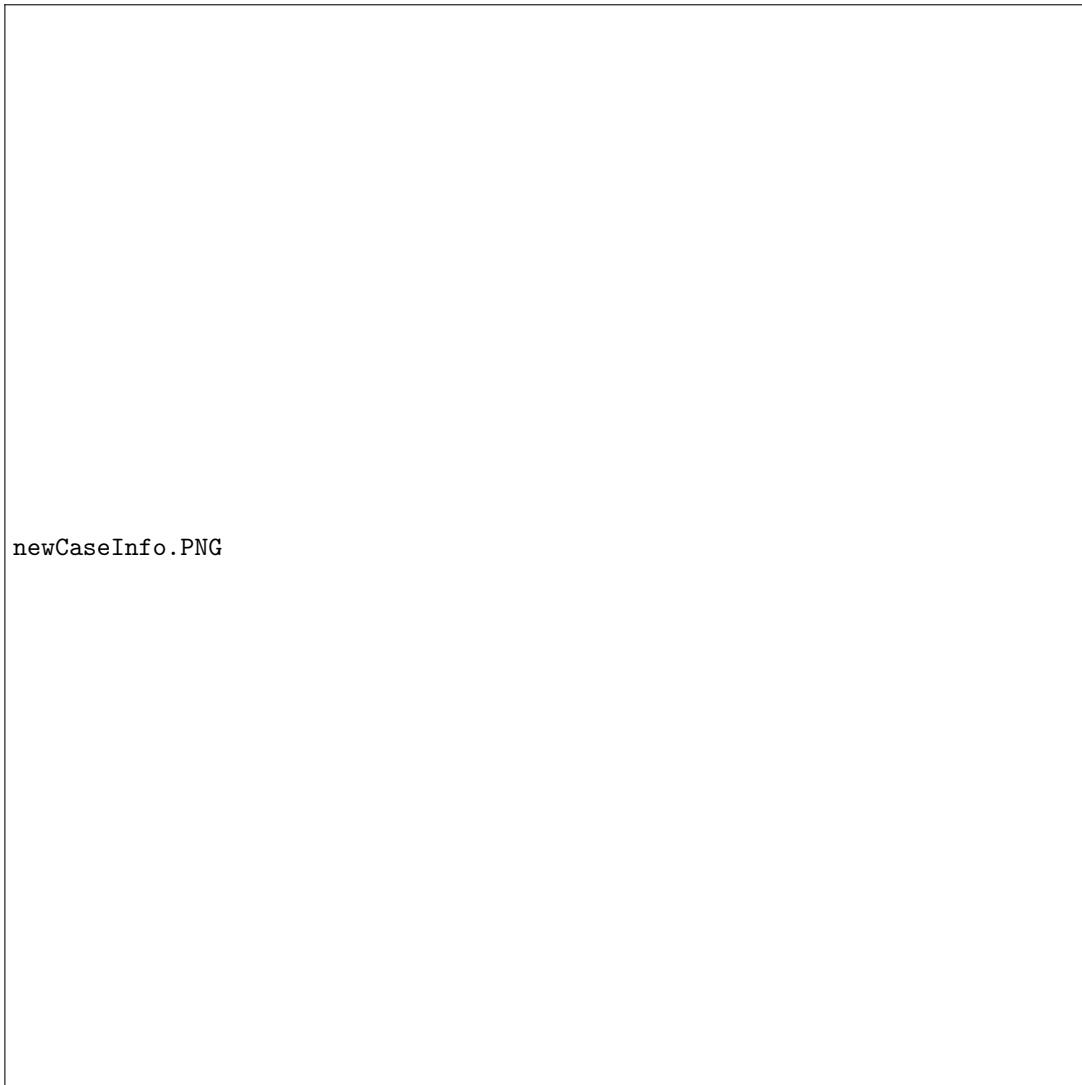
- Hash Lookup
- File Type Identification
- Extracting Embedded Files
- Picture Analysis
- Keyword Search
- Email Parsing
- Checking for Extension Mismatches
- Detecting Encryption
- Android Analysis
- Identifying Interesting Files

- PhotoRec Carver
- Checking Data Source Integrity
- GPX Parsing

These tools are essential for uncovering important information during investigations.

2 Creating a Case and Analysis

Launching Autopsy will display options to create a new case or open an existing case. To create a new case, click on the New Case button. Add a case name and a base directory for the case.



Additionally, we can add a Case type: Single User or Multi User. Multi User cases are used when multiple users are working on the same case.

Then we will be prompted to add additional information about the case shown below:



All fields on this panel are optional. Additionally, the Organization section will only be active if the central repository is enabled.

After creating the case, we will be prompted to add a data source shown below:



`addDataSource.PNG`

We need to select the host for the data source. There are three options:

- Generate new host based on data source name - this will typically create a host with a name similar to our data source with the ID used in the database appended for uniqueness.
- Specify new host name - this will allow us to create a new host with a custom name.
- Use existing host - this will allow us to select an existing host from the database.

After selecting the host, we need to select the data source type.

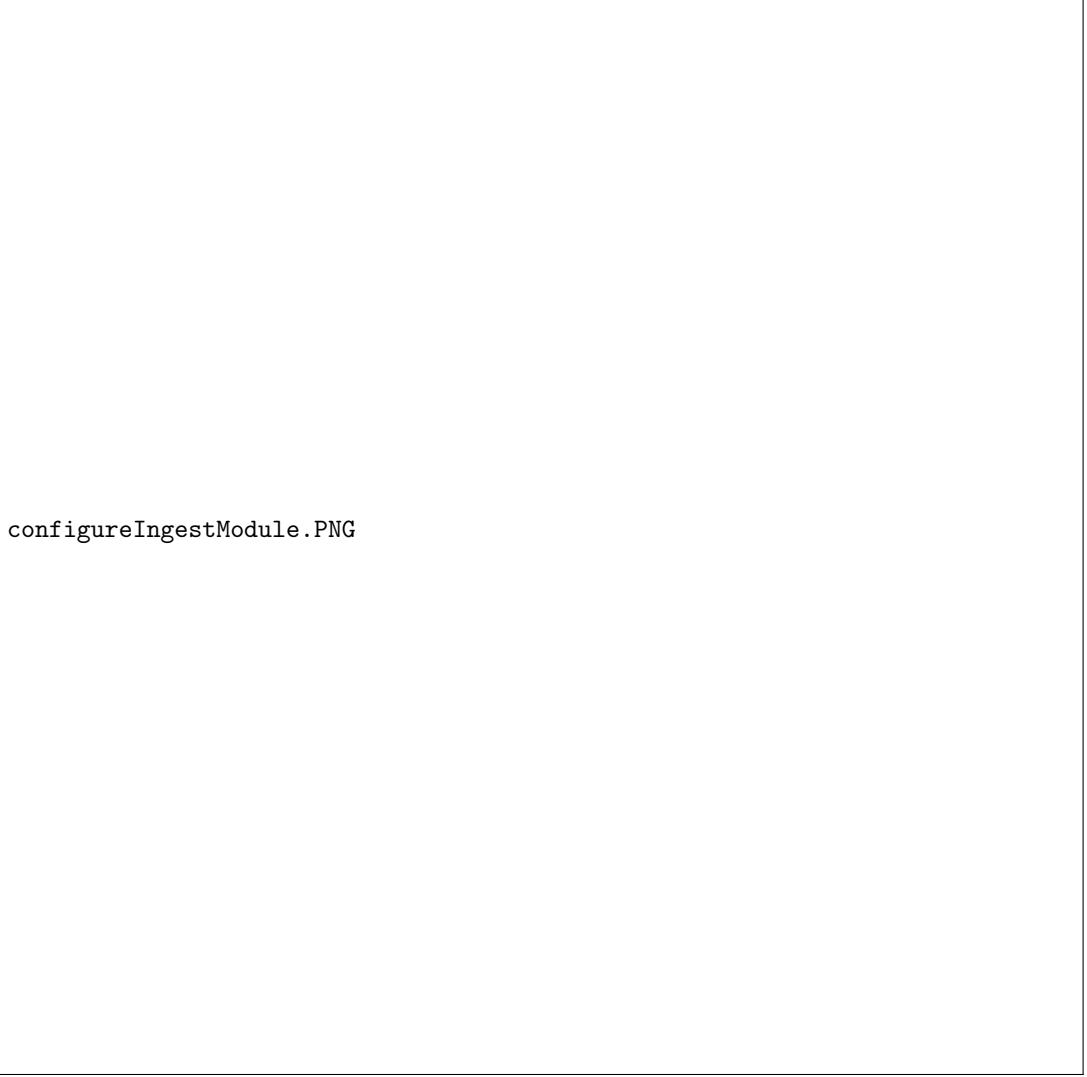


```
selectDataSourceType.PNG
```

Next we will be prompted to select the data source file.

`selectDataSourceFile.PNG`

After selecting the data source file, we will be prompted to configure the Ingest Modules:



configureIngestModule.PNG

Then we need to wait while Autopsy performs a basic examination of the data source and populates an embedded database with an entry for each file in the data source. After the basic examination of the data source is complete, the ingest modules will likely still be running but we can start browsing through the files in our data source.

3 Examination utilizing Ingest Modules

3.1 Recent Activity

The Recent Activity module in Autopsy gathers user engagement data from web browsers, installed programs, and the operating system, providing insights into online searches, visited websites, system

actions, and connections made. Additionally, it initiates Regripper to analyze Registry hive data, enhancing the scope of information retrieval. This module allows users to scrutinize activities spanning the last week, offering a comprehensive overview of device usage and interactions.

Configuring Custom Web Categories

1. Open Autopsy.
2. Go to **Tools** in the menu bar.
3. Select **Options**.
4. Navigate to **Custom Web Categories**.

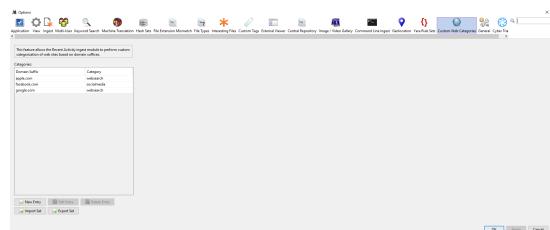


Figure 1: Recent Activity Settings

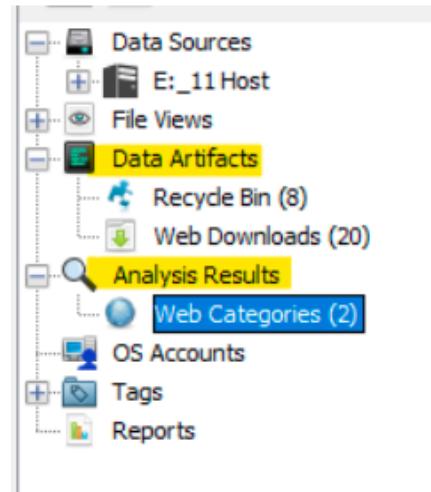


Figure 2: Results of Recent Activity

Results

The default settings' outcomes are displayed in the **Data Artifacts** section, while the results of customized settings are shown in the **Analysis Results** section, one after the other.

3.2 Hash Lookup

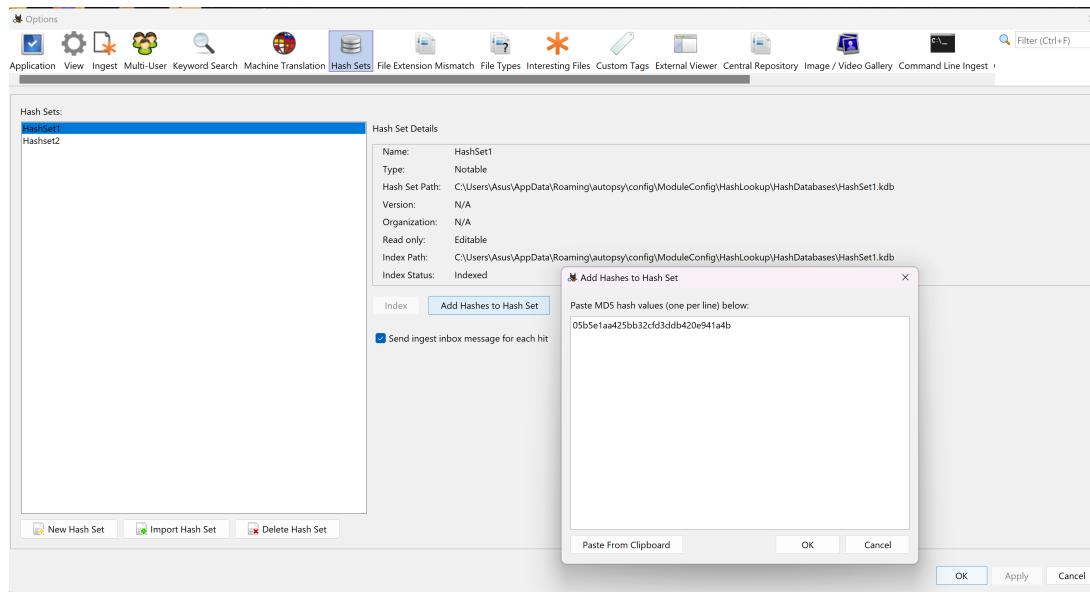
The Hash Lookup Module computes MD5 hash values for files and cross-references them in a database to classify files as notable, known, or unknown. In Autopsy, the Hash Sets tab enables users to maintain a list of files categorized as known good, meaning they're confirmed safe and can be excluded from routine file checks, saving time and system resources. Additionally, Hash Sets can manage files marked as notable or known bad, helping identify malicious or suspicious content during analysis to enhance security measures. This feature aids in flagging potential security threats or criminal activities for further investigation.

Configuring Hash Sets

To set up custom hash sets in Autopsy, follow these steps:

1. Open Autopsy.
2. Navigate to **Tools** in the menu bar.
3. Choose **Options**.
4. Access the **Hash Sets** dialog box.

In the **Hash Sets** dialog box, you can add known or notable hash sets. Utilize known hash sets such as the NSRL Reference Data Set (RDS) to quickly identify "known" files on a disk image, saving significant time and resources during analysis. Additionally, you have the flexibility to configure known or notable hash sets using your own dataset.



Results

In the **Analysis Results** under **Hashset Hits**, three matches are identified for files with hash values that correspond to those found in the **HashSet1** dataset.

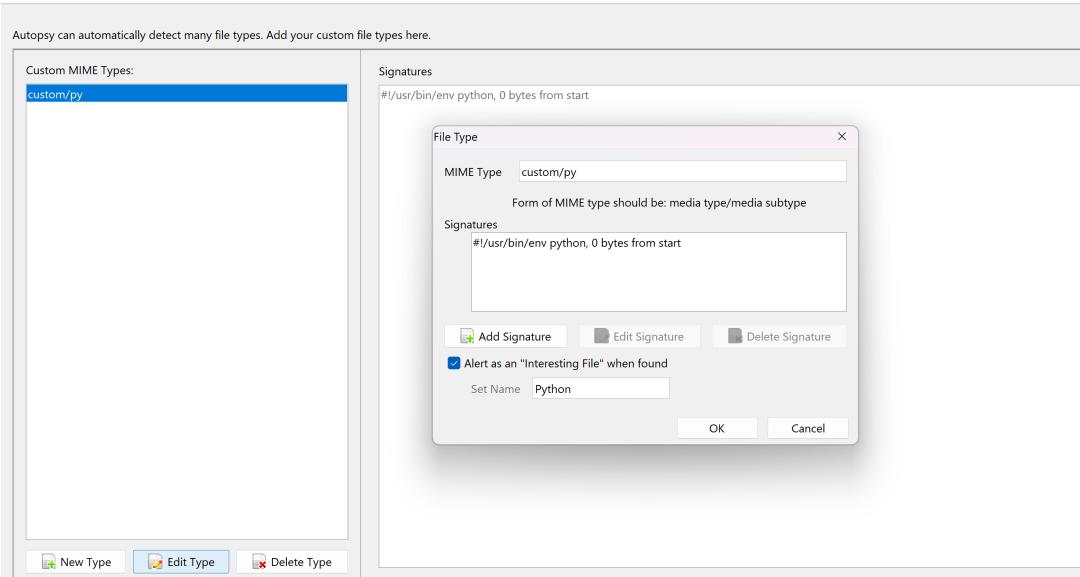
Source Name	S	C	O	MD5 Hash	Comment	File Path
XSDemo.txt	0			dfcc0ed978d146b5064c29f708b0113e		/LogicalFileSet1/Demo/Nothing/Something/Who/Assi...
Ch12_RandAlg.class_09.ppt	0			ce1b7763642d5856774dd4c2b6ac1d9c		/LogicalFileSet1/Demo/Nothing/Ch12_RandAlg.class...
CSE 406 Web Security Assignment.pdf	0			21dacd92fe68f27e3e768bbb015cca1b		/LogicalFileSet1/Demo/Nothing/Something/Who/Assi...

3.3 File Type Identification

Autopsy's File Type ID module is crucial for identifying files based on their inherent signatures, rather than relying solely on file extensions. Utilizing the Tika library, Autopsy can customize criteria to accurately detect primary file IDs. This module plays a pivotal role within Autopsy, enabling the identification of file types within the data source. Moreover, it serves as a foundation for other modules like the **Extension Mismatch Detector Module** and **Keyword Search Module**, aiding in comprehensive file type identification within the data set.

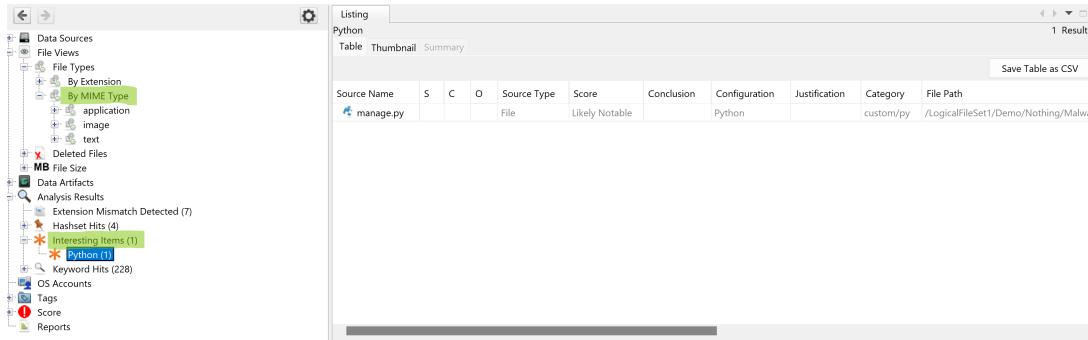
Configuring Global File Type Identification Settings

In global file type identification settings, we can configure custom file type of our interest



Results

In the File Types tab of Autopsy, the results from the **File Type** Identification module are displayed in two groups: **By Extension** and **By Mime Type**. **By Extension** groups files based on their file extensions, which may sometimes include incorrect files due to corrupted extensions. On the other hand, By Mime Type groups files based on their MIME types. Additionally, results from custom settings are shown grouped according to the specified MIME names in the **Interesting Items** tab.



The screenshot shows the Autopsy interface with the 'File Types' tab selected. Under 'File Views', 'By Extension' and 'By Mime Type' are listed, with 'By Mime Type' being the active view. The main pane displays a table titled 'Python' with one result: 'manage.py'. The table columns are: Source Name, S, C, O, Source Type, Score, Conclusion, Configuration, Justification, Category, and File Path. The row for 'manage.py' shows: Source Name (manage.py), S (File), C (Likely Notable), O (Python), Source Type (File), Score (N/A), Conclusion (Python), Configuration (Python), Justification (N/A), Category (custom/py), and File Path (/LogicalFileSet1/Demo/Nothing/Malwa).

3.4 Embedded File Extractor

The Embedded File Extractor module in Autopsy can open various archive formats such as ZIP, RAR, and others, as well as document formats like Doc, Docx, PPT, PPTX, XLS, and XLSX. It then sends the extracted files from these formats back through the ingest pipeline for further analysis. This module plays a crucial role in expanding archive files, allowing Autopsy to thoroughly analyze all files within the system. It also enables features like keyword search and hash lookup to examine files contained within these archives. However, it's worth noting that certain media content embedded within document formats like Doc, Docx, PPT, PPTX, XLS, and XLSX may not be extracted by the module.

Results

Each extracted file appears in the data source tree view as a subordinate of the containing archive and as an archive within the **Views** section under **File Types** and **Archives**.

Name	S	C	O	Modified Time	Change Time	Access Time	Created Time	Size	Flags(Dir)	Flags(Met.)
7.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	714899	Allocated	Allocated
6.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	1891900	Allocated	Allocated
8.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	12414	Allocated	Allocated
7.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	74673	Allocated	Allocated
9.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	229442	Allocated	Allocated
8.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	1053	Allocated	Allocated
9.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	8128	Allocated	Allocated
9.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	108694	Allocated	Allocated
codes.zip	0			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	3831	Allocated	Allocated
index.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	126716	Allocated	Allocated
index.pack.gz				0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	87010	Allocated	Allocated
july2023-A2-B2.zip	0			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	46049	Allocated	Allocated

3.5 Extension Mismatch Detector

The Extension Mismatch Detector module utilizes the findings from the File Type Identification module to identify files with extensions that do not typically match their detected file type.

Configuring Extension Mismatch Detector Settings

Within the Tools > Options > File Extension Mismatch dialog box, users have the capability to both add and remove MIME types along with their corresponding extensions.

File Types:	Allowed Extensions for image/jpeg:
audio/mpeg4	gif
audio/mpga	jpg
audio/mpeg	jif
audio/rogg	jif
audio/x-aac	pe
audio/x-aiff	jpeg
audio/x-flac	jpg
audio/x-mpeg	jpgms-thumbnail
audio/x-mpegurl	png
audio/x-ms-wma	tile
audio/x-wav	
image/bmp	
image/gif	
image/jpeg	
image/png	
image/tiff	
image/vnd.adobe.photoshop	
image/x-icon	
image/x-ms-bmp	
image/x-raw-nikon	
text/html	
text/plain	
video/3gpp	
video/3gppp	
video/mp2t	
video/mpeg	
video/quicktime	
video/x-fly	
video/x-m4v	
video/x-ms-asf	
video/x-ms-wmv	
video/x-msvideo	

Results

Results are shown in Analysis Results > Extension Mismatch Detected.

Source Name	Justification	Extension	MIME Type	File Path
calorieCalculator.tsx	File has MIME type of text/plain	tsx	text/plain	/LogicalFileSet1/Demo/Nothing/calorieCalculator
data.ts	File has MIME type of text/plain	ts	text/plain	/LogicalFileSet1/Demo/Nothing/Malware/data.ts
abcd.drawio	File has MIME type of text/plain	drawio	text/plain	/LogicalFileSet1/Demo/Nothing/Something/abcd
serotonin.pdf	File has MIME type of image/png	pdf	image/png	/LogicalFileSet1/Demo/Nothing/Something/sero
rice_gluten_free.ppt	File has MIME type of image/jpeg	ppt	image/jpeg	/LogicalFileSet1/Demo/Nothing/Something/rice
RDS_2023.12.1_modern_minimal_database.dhash	File has MIME type of text/plain	dhash	text/plain	/LogicalFileSet1/Demo/Nothing/Malware/Worm
Sample IO.png	File has MIME type of application/zip	png	application/zip	/LogicalFileSet1/Demo/Nothing/Malware/Worm/

3.6 Picture Analyzer

The Picture Analyzer module in Autopsy retrieves important information from images, such as location, date, time, camera model, and settings, known as EXIF metadata. This data is then added to the Blackboard for further analysis, providing insights into the photograph's details like where and when it was taken, and what camera was used. Additionally, the module converts HEIC/HEIF photos to JPG format while keeping their EXIF data intact, ensuring they are treated and saved just like regular JPG images.

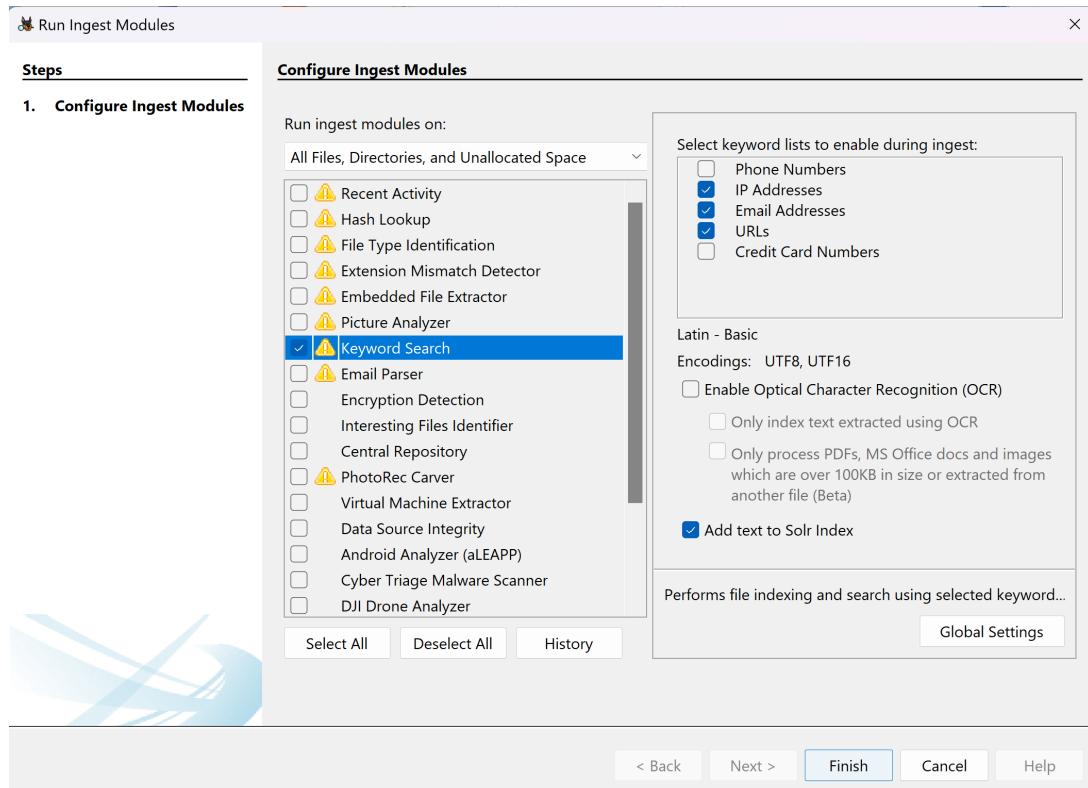
Results

The outcomes are displayed in the result trees.

Name	Size	Flags(Meta)	Location	MD5 Hash	SHA-256 Hash
image1.png	27412	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	288b0945ac492488158a03b12a2202	31244ab093a53d42f63d9bab81f92d4082adacda0e9ab2c0
image10.png	32110	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	923541a695413a73f94aa71b8a5eed3	8315699416b500d7559a17b0574350539832be6cad01181
image11.png	37399	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	39632acdfe07a0705e95944ff0109	c603959e2aca3c95958084e49505a5e93d053aae90080
image12.png	24603	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	ab29f761c7f9509e9bce1983dcf11	f3651a607349d169d94191295aae25b83f7a5f74a32
image13.png	39267	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	00b3a3b8e42d4a02be0bc1cf7fa211a	a21aa4fb7e0d0566ba2921e915a2c05caaa49982d8b1
image14.png	49529	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	c8b33d63b8d1be984879af584697	5a7e093810ed2f16bcaa049c51c0f6ac9414a58c2815
image15.png	11652	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	5640949b9e03721fc9b0730a1b7359	a4ca5fbdbdf4932c145b26e7988514e93d3cb420170
image16.png	41823	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	20682d7d309e224a35b2d05757d43	516d0bd23385c02a2dfe82cfe2d3096bf0c9c7a21b4
image17.png	19525	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	19a3cd1afad08bdc5d240f1399d2d3d09	134389d3c3e7eb5bf5d4fd1d3b3e8f919bae396be8de7
image18.png	19487	Allocated	/LogicalFileSet1/Demo/504_Buffer_Overflow.pptx/ima...	09165311620511f19376d2017ce6b	aeb49911a9a43ee0fbcb09a5f96a4b7d0783bbe6119e

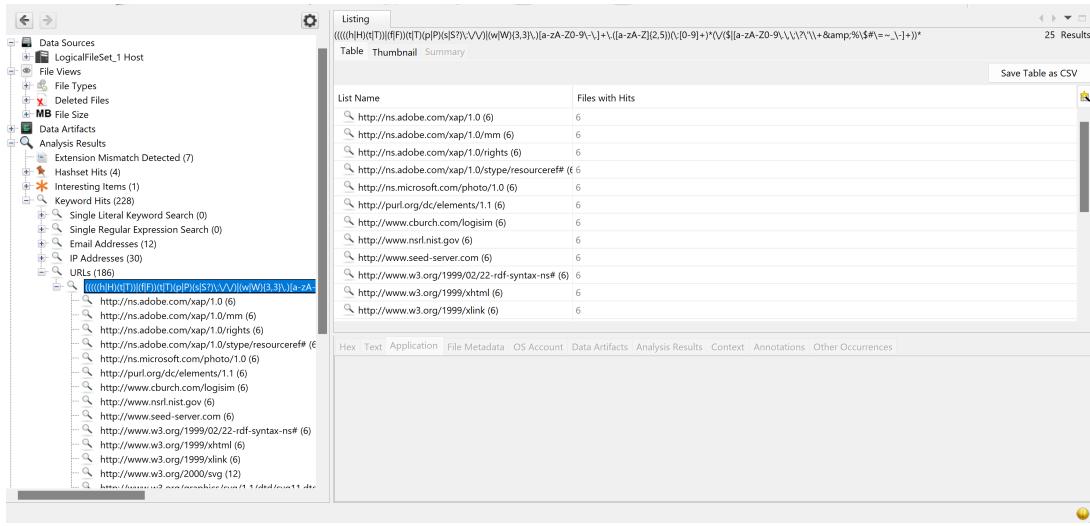
3.7 Keyword Search

The Keyword Search module enables searching during the ingest process and also supports manual text searches post-ingest (refer to Ad Hoc Keyword Search). It extracts text from ingested files, selected reports generated by other modules, and results from other modules. Users can specify keywords to search for, add regular expressions for searching, and include keywords to exclude from the search.



Results

In the global settings options, users can incorporate custom keyword rules for searching. When this module is executed, it searches for specified keywords within files and their metadata. The search results are displayed in **Analysis Results > Keyword Hits**, grouped together by the detected keyword.



3.8 Interesting Files Identifier

The Interesting Files Identifier module automatically flags files and directories that adhere to pre-defined criteria. This feature proves beneficial for verifying the presence of files with specific names or paths within the data source, or when files of particular types consistently warrant attention.

Configuring Interesting Files Settings

- Users can define a set of rules by accessing the **Tools > Options > Interesting Files** dialog box.
- For instance, in our demonstration, we established a rule set named "Suspicious."
- Within this set, we incorporated two specific rules:
 1. If any file contains the substring "Worm" in its name, it will be categorized as an "interesting file."
 2. Similarly, if any file contains the substring "virus" in its name, it will also be categorized as an "interesting file."

Results

After running the "Interesting Files Identifier" ingest module, the results can be viewed in **Analysis Results > Interesting Items**. Three files have been identified as interesting items due to their names containing the substrings "worm" or "virus," as defined within the "Suspicious" set of rules.

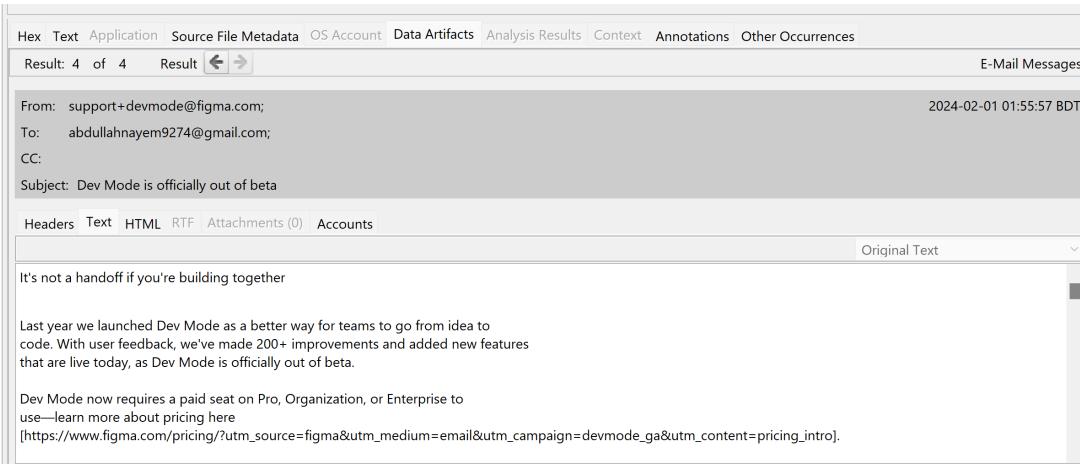
3.9 Email Parser

The Email Parser module recognizes MBOX, EML, and PST format files by their signatures. It extracts emails from these files and adds the results to the Blackboard. Additionally, it identifies the email accounts used in the discovered emails.

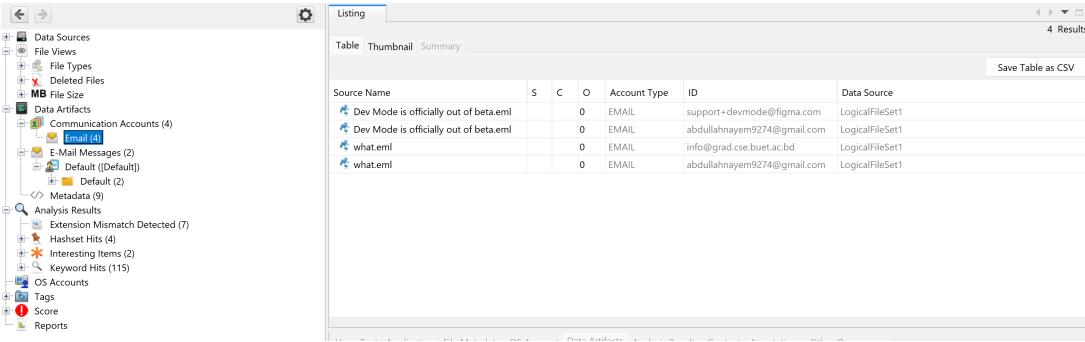
Results

Source Name	S	C	O	E-Mail From	E-Mail To	Subject
Dev Mode is officially out of beta.eml				support+devmode@figma.com	abdullahnayem9274@gmail.com	Dev Mode is officially out of beta
what.eml				info@grad.cse.buet.ac.bd;	abdullahnayem9274@gmail.com;	You have submitted your assignment su

After running the "Email Parser" ingest module, the results can be viewed in **Data Artifacts > E-mail messages**. Two email files have been identified by the module.



We can also see the details of a selected email in the bottom part of the result.



Source Name	S	C	O	Account Type	ID	Data Source
Dev Mode is officially out of beta.eml	0	0	0	EMAIL	support+devmode@figma.com	LogicalFileSet1
Dev Mode is officially out of beta.eml	0	0	0	EMAIL	abdullahnayem9274@gmail.com	LogicalFileSet1
what.eml	0	0	0	EMAIL	info@grad.cse.buet.ac.bd	LogicalFileSet1
what.eml	0	0	0	EMAIL	abdullahnayem9274@gmail.com	LogicalFileSet1

From **Data Artifacts > Communication Accounts > Email**, we can view the email accounts that have been used in the emails identified by the module.

3.10 Encryption Detection

The Encryption Detection Module is created to find files that might be encrypted or password-protected. It does this by using both general calculations of randomness and customized tests for different types of files.

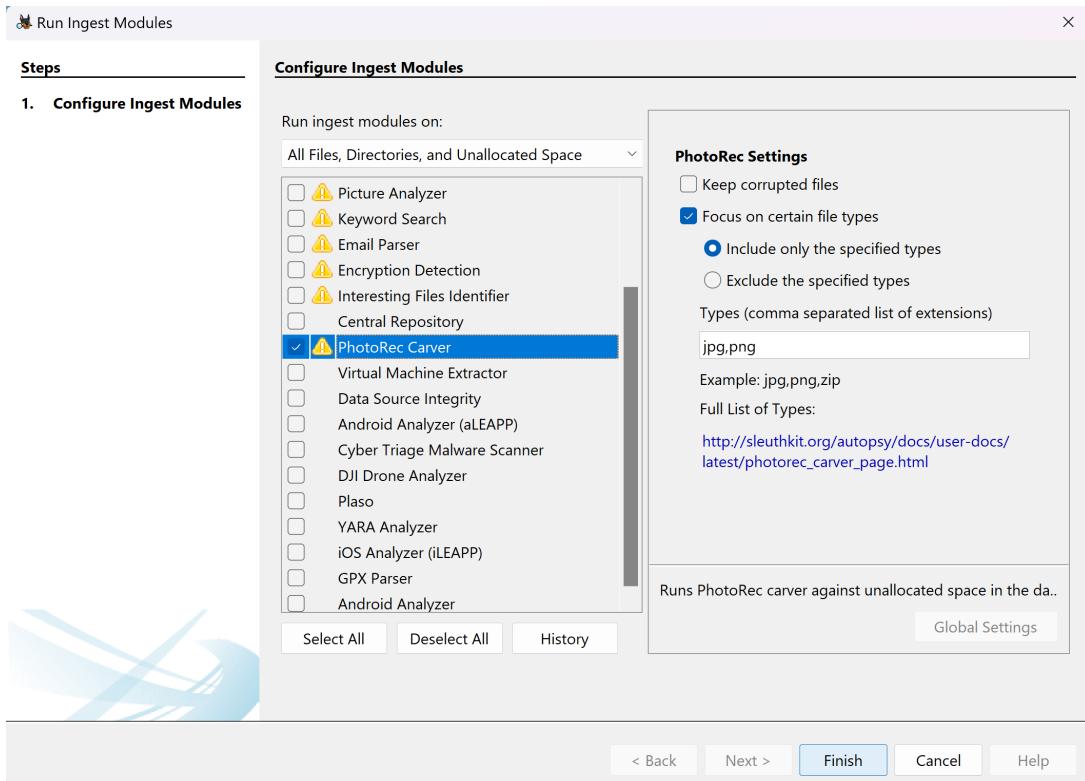
Results

The screenshot shows the Encase Forensic software interface. On the left, there is a navigation tree with categories like Data Sources, File Views, File Types, Deleted Files, MB file size, Data Artifacts, Communication Accounts, E-Mail Messages, Metadata, Analysis Results, OS Accounts, Tags, Score, and Reports. Under Analysis Results, there is a section for Encryption Detected with one item listed: 'Encryption Detected (1)'. The main pane is titled 'Listing' and shows a table titled 'Encryption Detected'. The table has columns: Source Name, S, C, O, Source Type, Score, Conclusion, Configuration, Justification, and Comment. One row is visible for 'RecipeShare-API.xlsx' with the following details: Source Type is 'File', Score is 'Notable', Conclusion is 'Password protection detected.', Configuration is 'Password protection detected.', and Justification is 'Password protection detected.'. Below the table, there is a search bar with fields for Hex, Text, Application, File Metadata, OS Account, Data Artifacts, Analysis Results, Context, Annotations, Other Occurrences, Strings, Extracted Text, and Translation. There are also buttons for Save Table as CSV, Page: 1 of 1 Page, Matches on page: - of -, Match, 100%, and Reset.

After executing the "Encryption Detection" ingest module, the outcomes can be observed in **Analysis Results > Encryption Detection**. one file has been recognized as requiring a password for access.

3.11 PhotoRec Carver

The PhotoRec Carver module retrieves files from the unallocated space within the data source and sends them for further processing. This assists users in uncovering additional data about previously deleted files on the device. These files are found in areas of the device's storage that appear empty. Users can customize the module's runtime settings to choose whether to retain corrupted files and specify which file types to include or exclude.



Results

The results of carving show up on the tree under the appropriate data source with the heading (CarvedFiles)

Name	S	C	Modified Time	Change Time	Access Time
f0056032.ber			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
f0000878.xml			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:01
f0000880.pdf			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:01
f0001016_python_3_8_3_amd64_exe			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:01
f0002070.cab			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:01
f0002699.cab			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:01
f0055328.sqlite			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:01
f0056032.ber			0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00

```

0x00000000: 41 41 41 41 41 41 41 41 53 61 6D 70 6C 65 20 66 AAAAAAAA
0x00000010: 69 6C 65 20 66 6F 72 20 50 68 6F 74 6F 52 65 63 file for PhotoRec
0x00000020: 20 64 6F 63 75 6D 65 6E 74 61 74 69 6F 6E 2B 00 documentation..
0x00000030: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000040: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000070: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000080: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000090: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000000A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000000B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

```

3.12 GPX Parsing Module: Importing and Analyzing GPS Data

In Autopsy, we utilize a GPX parser as a tool to interpret and analyze GPX files. GPX, which stands for GPS Exchange Format, is a common file format used to store GPS data such as waypoints, tracks, and routes.

With our GPX parser, investigators can import GPX files collected from various sources, such as GPS devices or mobile phones, into our forensic investigation platform. Once imported, our GPX parser allows users to extract and visualize geographical data, including location coordinates, timestamps, and other metadata associated with the GPS data.

This functionality enables our forensic analysts to map out movement patterns, routes, and points of interest recorded in the GPX files, providing valuable insights into the geographic context of a digital investigation. Additionally, by integrating GPX parsing capabilities, Autopsy enhances its ability to analyze and correlate GPS data with other digital evidence, facilitating comprehensive forensic examinations.

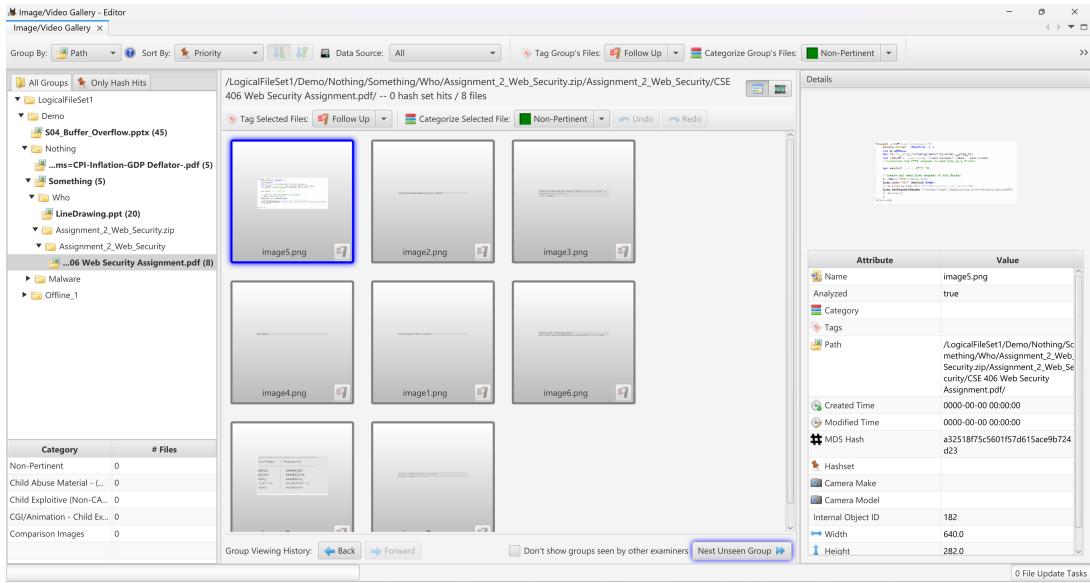


GPX Parser.png

4 Specialize Views

4.1 Images/Video Gallery

The Image Gallery feature in Autopsy assists in investigations related to images and videos. It categorizes photos into folders and properties, simplifying the management of large collections and focusing on significant content. It allows for immediate image viewing during ingestion, eliminating the need to wait for the entire process to finish.



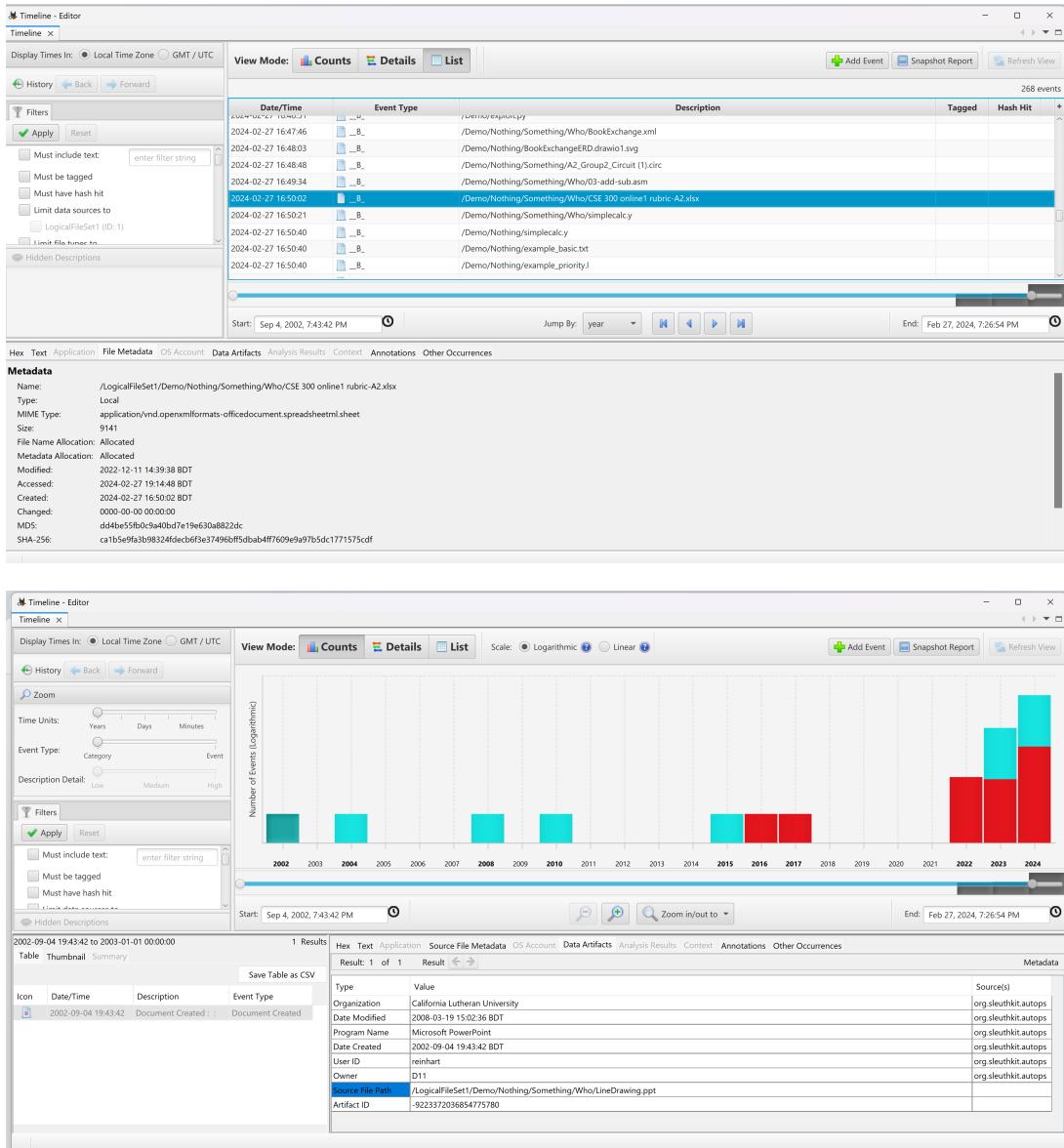
4.2 Timeline

Autopsy's Timeline is a robust digital forensics tool that logs important events like web activity, external device connections, EXIF photo additions, and their connections with file system modifications.

To fully utilize the Timeline feature, it's essential to run the Hash Lookup, Recent Activity, and Picture Analyzer modules beforehand.

The Timeline tool is event-based, with each event having a timestamp, category, and description. While each event is distinct, users can manually group them together if needed.

Autopsy's Timeline feature gathers and organizes data from various sources into event types, such as for File System Changes (Access, Creation, Modification, Deletion).

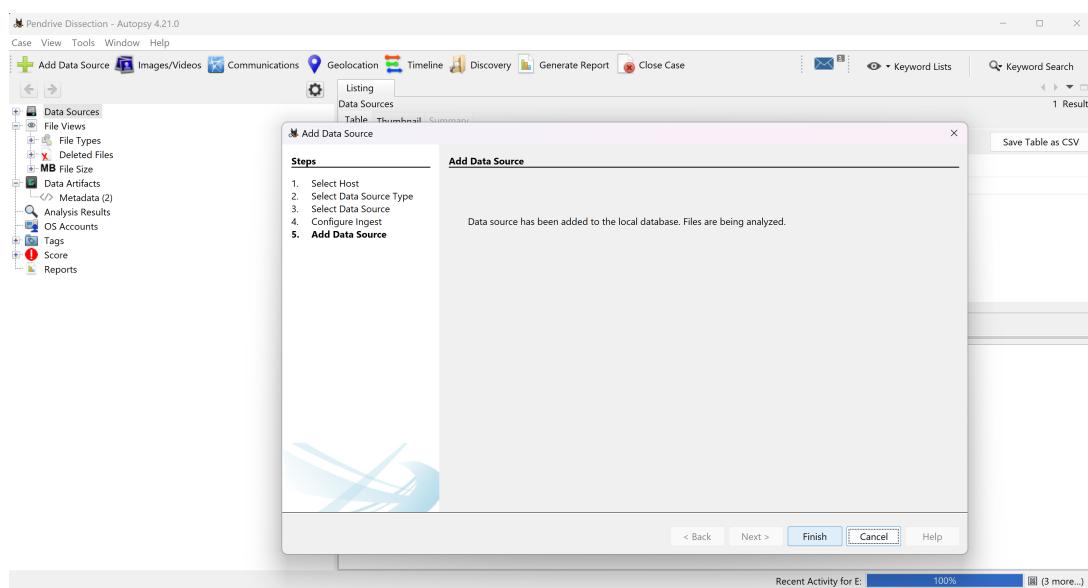
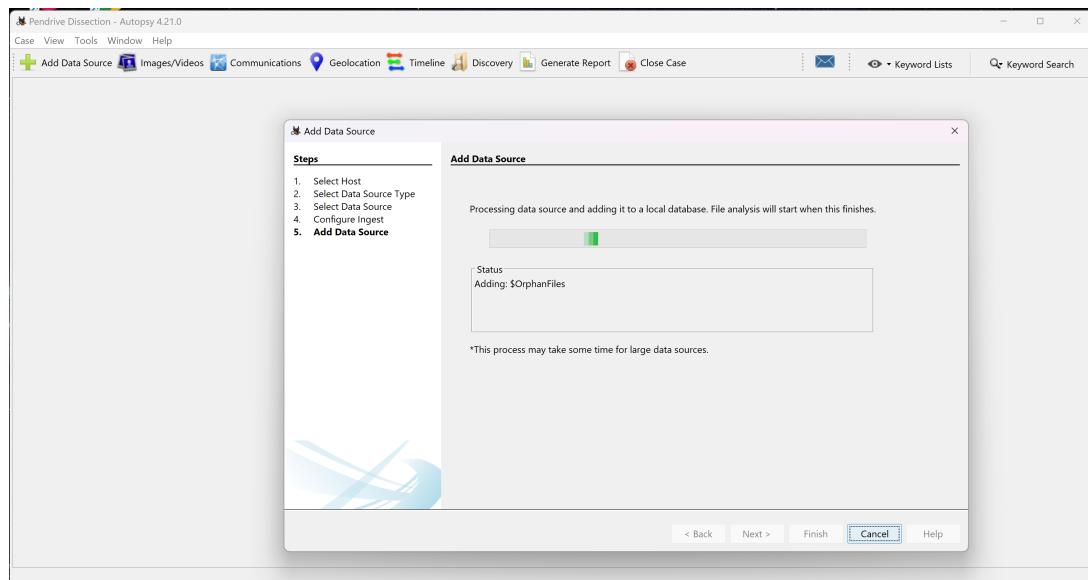


4.3 File Deletion and Recovery on USB Pendrives

In this exploration, we delve into the intricate process of deleting files from USB pendrives and subsequently recovering them using Autopsy, a powerful digital forensic tool. We uncover the digital trails left behind after deletion, analyzing metadata such as timestamps and file allocation tables within the Autopsy interface. Furthermore, we navigate through Autopsy's specialized features and modules designed for scanning and retrieving deleted files, providing detailed insights for forensic investigators seeking to extract crucial evidence from USB pendrives.

Here we will recover and export a file named "47-Collaboration Diagram" (It was previously in our pendrive but we deleted it) to our local machine.

Configuring Setup for File Recovery



Results

File Explorer (Windows File Explorer) showing the contents of the 'UBUNTU 22_0 (E)' drive:

```

UBUNTU 22_0 (E)
This PC > UBUNTU 22_0 (E)

Name Date modified Type Size
This folder is empty.
0 items

```

Pendrive Dissection - Autopsy 4.21.0 interface showing file analysis results:

- Case View Tools Window Help**
- Geolocation Timeline Discovery Generate Report Close Case**
- Listing Office Table Thumbnail Summary**
- Save Table as CSV**
- 6 Results**

File Types listing:

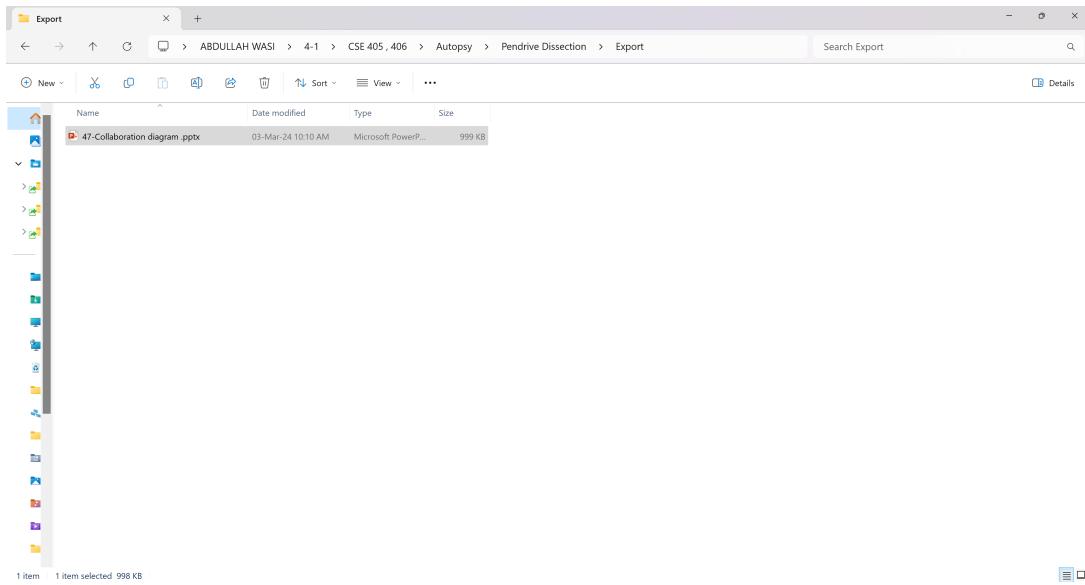
- By Extension:
 - Images (24)
 - Videos (0)
 - Audio (0)
 - Archives (4)
 - Databases (0)
 - Documents:
 - HTML (0)
 - Office (6)
 - PDF (1)
 - Plain Text (89)
 - Rich Text (0)
 - Executable:
 - exe (10)
 - dll (56)
 - .bat (0)
 - .cmd (0)
 - .com (0)
 - By MIME Type
 - Deleted Files
 - All (2050)
- MB File Size
- Data Artifacts
- Analysis Results
- OS Accounts
- Tags
- Score

Table view showing file details:

Name	Created Time	Size	Flags(Meta)	Flags(Dir)	Modified Time	Change Time	Access Time
_47-Collaboration diagram.pptx	2023-08-29 14:00:05 BDT	4096	Allocated	Allocated	2023-08-29 14:00:16 BDT	0000-00-00 00:00:00	2023-08-29 00:00:00
_CSE 325 Presentation 2.pptx	2023-07-25 14:10:54 BDT	4096	Allocated	Allocated	2023-07-25 14:10:56 BDT	0000-00-00 00:00:00	2023-08-29 00:00:00
X _47-Collaboration diagram.pptx	2023-08-29 10:23:39 BDT	1022013	Unallocated	Unallocated	2023-08-29 10:23:38 BDT	0000-00-00 00:00:00	2023-08-29 00:00:00
X BookExchange.pptx	2023-07-25 15:48:06 BDT	4481120	Unallocated	Unallocated	2023-07-25 15:48:10 BDT	0000-00-00 00:00:00	2024-03-03 00:00:00
X CSE 325 T							
X ~\$BookExch							

Save Table as CSV dialog:

- Save in: Export
- Recent Items:
 - Desktop
 - Documents
 - This PC
- File name: 47-Collaboration diagram.pptx
- Save
- Cancel



5 Report Generation

5.1 Tagging Files

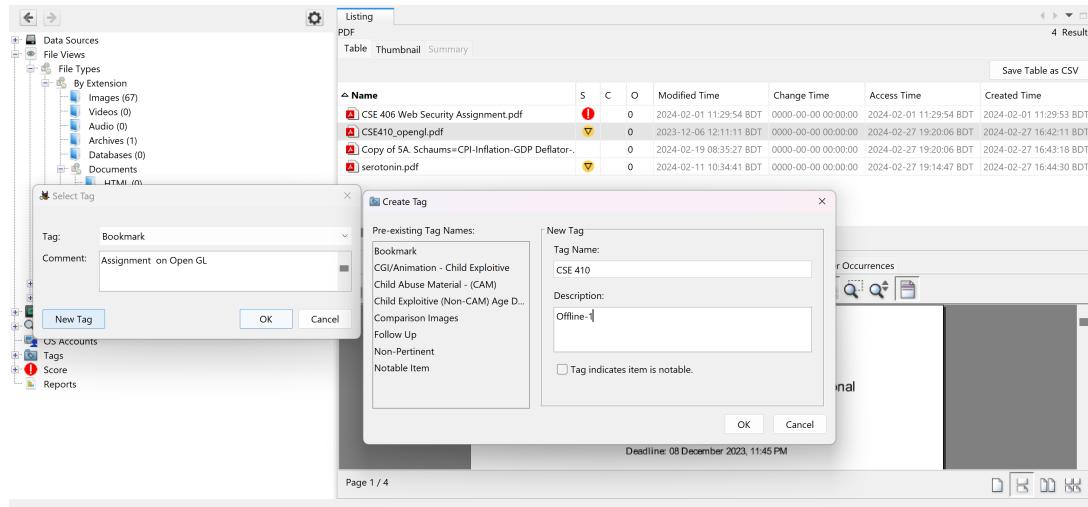
When users discover something of interest, they can tag it by right-clicking on the item and selecting one of the tag options available:

If the file itself is noteworthy, users can choose the **TagFile** option. Alternatively, if the results of the analysis are significant, they can opt for the **TagResult** option.

Name	S	C	O	Modified Time	Change Time	Access Time	Created Time
CSE 406 Web Security Assignment.pdf	0			2024-02-01 11:29:54 BDT	0000-00-00 00:00:00	2024-02-01 11:29:54 BDT	2024-02-01 11:29:53 BDT
CSE410_opengl.pdf	0			2023-12-06 12:11:11 BDT	0000-00-00 00:00:00	2024-02-27 19:20:06 BDT	2024-02-27 16:42:11 BDT
Copy of 5A Schaums=CPI-Inflation-GDP.C							
serotonin.pdf							

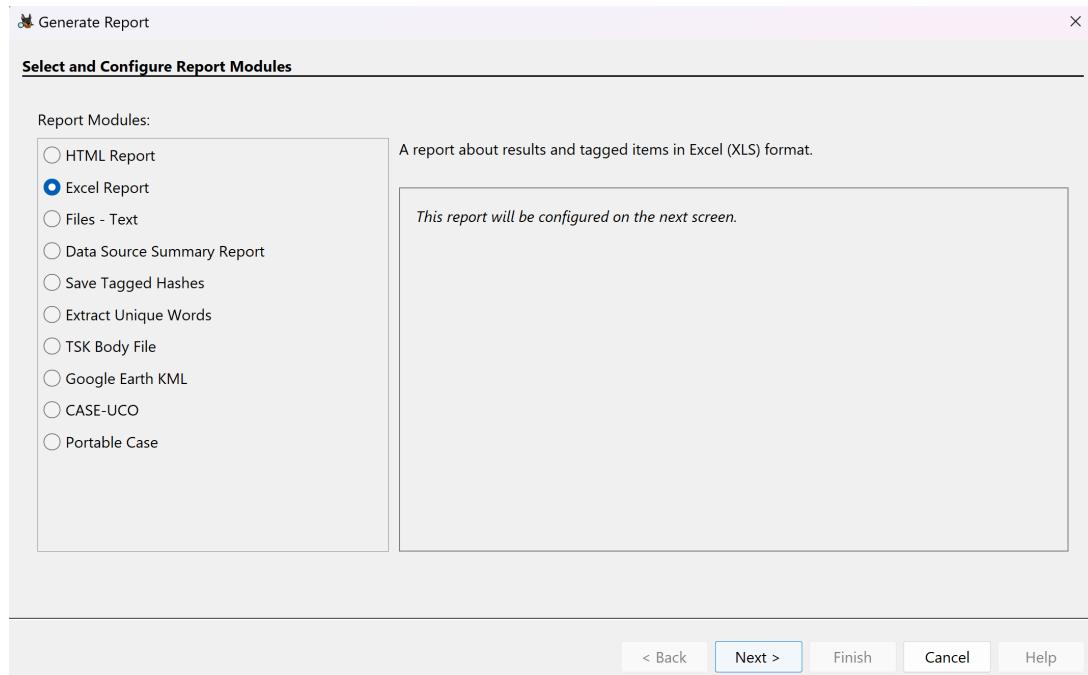
5.2 Tagging and Commenting

Within Autopsy, commenting entails appending remarks and annotations to digital artifacts throughout a forensic examination. This supports documentation, collaboration, and context development among investigators, thus improving evidence organization and enhancing investigative insights and clarity. This functionality also allows for the application of multiple tags to a single file and aids in tracking the progress of the investigation.

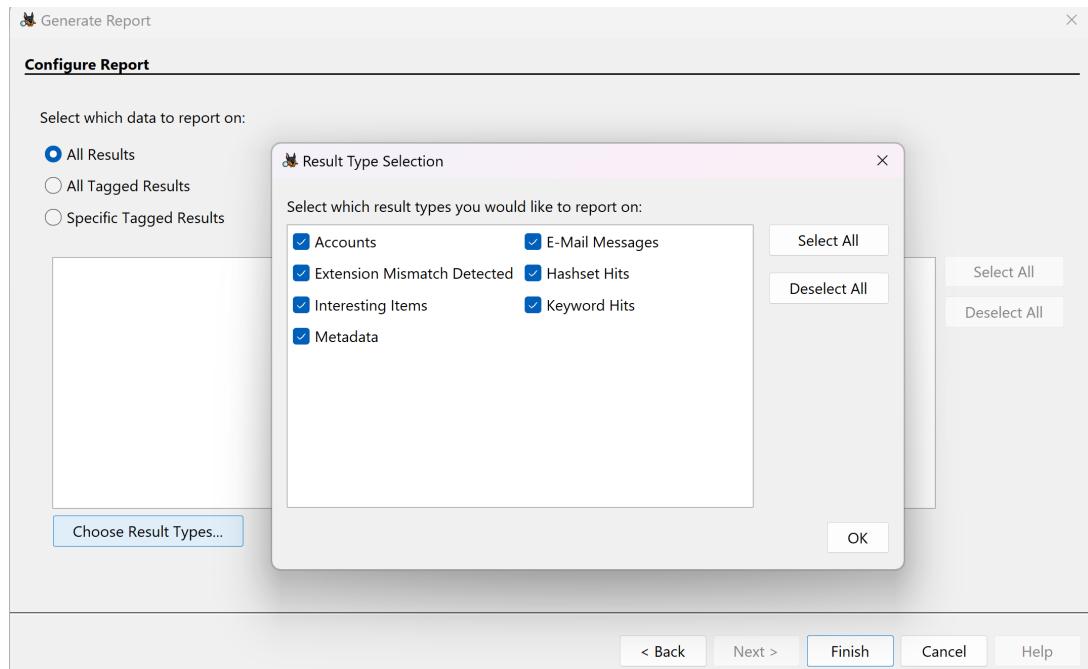


5.3 Reporting

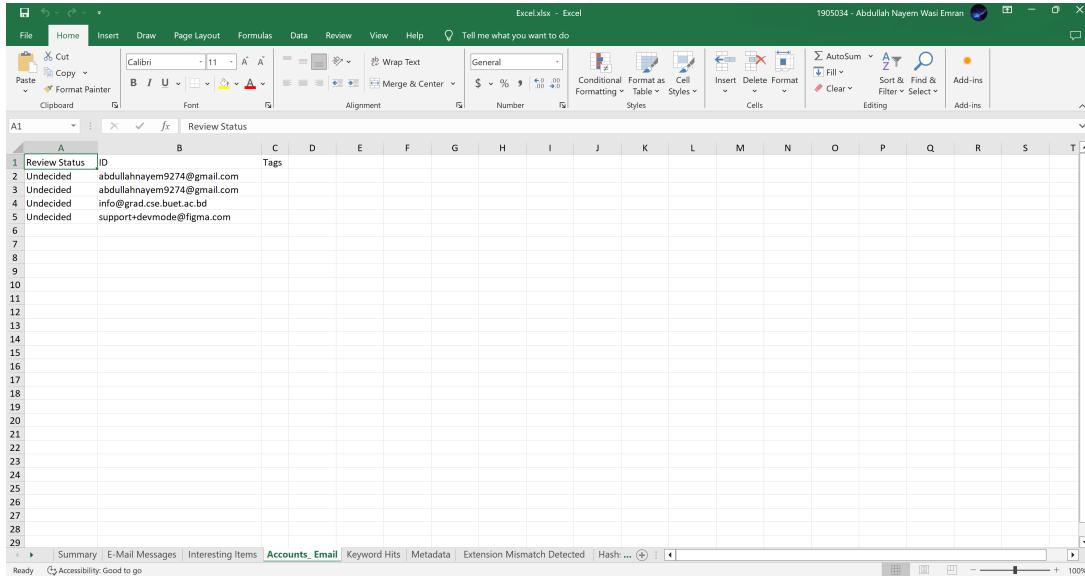
The report modules enable users to extract crucial information from a case in diverse formats. This encompasses generating HTML or Excel reports containing all extracted content, keyword hits, and more from a case, as well as creating KML files from any coordinates discovered for integration into software such as Google Earth.



Utilizing the tagged files, Autopsy has the capability to generate a report encompassing comprehensive information regarding the files. These reports can be generated in various formats and accessed through the **Reports** tab.



and generated report is



A screenshot of a Microsoft Excel spreadsheet titled "Excel.xlsx - Excel". The spreadsheet contains a single sheet with data starting from row 1. Row 1 has columns A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T. Column A is labeled "Review Status" and column B is labeled "ID". The data rows are as follows:

Review Status	ID	Tags
Undecided	abdullahnayem9274@gmail.com	
Undecided	abdullahnayem9274@gmail.com	
Undecided	info@grad.cse.buet.ac.bd	
Undecided	support+devmode@figma.com	

6 Autopsy Functionality Testing

6.1 Extended DOS Partition Test

Introduction

Many DOS partition utilities typically restrict users from creating a third entry within an extended partition. To investigate this limitation, an experiment involved manually altering the partition table using a hex editor to create a third entry. Subsequently, the system was booted, and both Windows and Linux successfully detected and permitted mounting of the third entry within the extended partition. This validation aimed to ensure that forensic tools, like Autopsy, also facilitate investigators in examining partitions within the third entry of an extended partition.

Source

The disk image can be downloaded from here: [Disk Image Download Link](#)

Outputs from Autopsy

Name	ID	Starting Sector	Length in Sectors	Description	Flags
vol1 (Unallocated: 0-62)	1	0	63	Unallocated	Unallocated
vol2 (DOS FAT16 (0x04): 63-52415)	2	63	52353	DOS FAT16 (0x04)	Allocated
vol3 (DOS FAT16 (0x04): 52416-104831)	3	52416	52416	DOS FAT16 (0x04)	Allocated
vol4 (DOS FAT16 (0x04): 104832-157247)	4	104832	52416	DOS FAT16 (0x04)	Allocated
vol7 (Unallocated: 157248-157310)	7	157248	63	Unallocated	Unallocated
vol8 (DOS FAT16 (0x04): 157311-209663)	8	157311	52353	DOS FAT16 (0x04)	Allocated
vol9 (Unallocated: 209664-209726)	9	209664	63	Unallocated	Unallocated
vol10 (DOS FAT16 (0x04): 209727-262079)	10	209727	52353	DOS FAT16 (0x04)	Allocated
vol13 (Unallocated: 262080-262142)	13	262080	63	Unallocated	Unallocated
vol14 (DOS FAT16 (0x06): 262143-312479)	14	262143	50337	DOS FAT16 (0x06)	Allocated

Conclusion

- Autopsy accurately identifies and displays partitions within a disk image.
- Autopsy correctly identifies and displays extended partitions (logical partitions) within primary extended partitions.
- Autopsy can present files and their properties within partitions, even if the file is empty.

6.2 FAT Keyword Search

Introduction

The provided test image comprises a **FAT file system** containing numerous ASCII strings. The objective of this examination is to determine which tools are capable of identifying various types of strings. Thus, it's important to note that not all strings listed in the table below will necessarily be detected by each tool. Failure to detect a specific string does not necessarily indicate an error in the tool. For instance, the string '1slack1' may extend beyond the end of a file and into its slack space, which may or may not be detected by different tools. As long as the functionality of the tool is adequately documented, it is the responsibility of the user to employ the tool appropriately to collect potential evidence.

Source

The disk image can be downloaded from here: [Disk Image Download Link](#)

Outputs from Autopsy

This screenshot shows the Autopsy 4.21.0 interface. The left sidebar displays a tree view of data sources, including 'fat-img-kw.dd_1 Host' which contains 'fat-img-kw.dd', '\$Orphanfiles (0)', and '\$Unalloc (1)'. The 'Keyword Hits' node under 'Analysis Results' is selected. The main pane shows a table of keyword hits:

List Name	Number of Children
Single Literal Keyword Search (315)	6
Single Regular Expression Search (8)	1

Below the table are tabs for Hex, Text, Application, File Metadata, OS Account, Data Artifacts, Analysis Results, Context, Annotations, Other Occurrences, Strings, Extracted Text, and Translation. The bottom status bar shows 'Page: 1 of 1' and 'Script: Latin - Basic'.

This screenshot shows the Autopsy 4.21.0 interface with the 'Keyword search 15 - \$SECOND' tab selected. The left sidebar shows the same data source structure as the first screenshot. The 'Keyword Hits' node is expanded, showing two entries under 'Single Literal Keyword Search (315)': 'second' and 'file2.dat'. The table below provides more detail:

Name	Keyword Preview	Location	Modified Time	Change Time	Access Time
second	<second>	/img_fat-img_kw.dd/second	2003-08-21 01:31:40 BDT	0000-00-00 00:00:00	2003-08-21 00:00:00 BDT
file2.dat	T tr?-(OCJDDR(L)-\$SECOND=TE52Hp9(Isla	/img_fat-img_kw.dd/file2.dat	2003-08-21 01:34:22 BDT	0000-00-00 00:00:00	2003-08-21 00:00:00 BDT

The bottom status bar shows 'Page: 1 of 1' and 'Script: Latin - Basic'.

The MD5 of the image is bac12239bd466fa6c86ceb0b0426da0a.

Conclusion

In Autopsy, users have the capability to search for keywords within the data source. The search operation extends to file names, file content, and file metadata, including unallocated space within the data source. In this particular test, we conducted a keyword search within a FAT file system

disk image. It's noteworthy that Autopsy, being a robust forensic tool, offers similar functionalities across a range of file systems such as NTFS, EXT3FS, and others.

6.3 FTS Undelete Test

Introduction

The test image comprises a 6MB FAT file system containing six deleted files and two deleted directories. These files vary in size, from single cluster files to multiple fragments. It's important to note that no data structures were altered during this process to impede recovery efforts. The files were initially created and deleted in Windows XP and subsequently imaged in a Linux environment.

Source

The disk image can be downloaded from here: [Disk Image Download Link](#)
The MD5 of the image is 4aeb06ecd361777242ab78735d51ace6.

Outputs from Autopsy

Name	Modified Time	Change Time	Access Time	Created Time	Size
_rag1.dat	2004-02-14 12:51:16 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:50:48 BDT	1584
_rag2.dat	2004-02-14 12:52:54 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:51:01 BDT	3873
_img.dat	2004-02-14 12:52:06 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:52:04 BDT	780
_ult1.dat	2004-02-14 12:52:44 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:52:25 BDT	3801
_jr1	2004-02-14 12:53:42 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:53:41 BDT	1024
[current folder]	2004-02-14 12:53:42 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:53:41 BDT	1024
[parent folder]	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	16384
dir2	2004-02-14 12:53:50 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:53:48 BDT	1024
[current folder]	2004-02-14 12:53:50 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:53:48 BDT	1024
[parent folder]	2004-02-14 12:53:42 BDT	0000-00-00 00:00:00	2004-02-14 00:00:00 BDT	2004-02-14 12:53:41 BDT	1024

File Metadata

Name:	/img_6-fat-undel.dd/_rag1.dat
Type:	File System
MIME Type:	application/octet-stream
Size:	1584
File Name Allocation:	Unallocated
Metadata Allocation:	Unallocated
Modified:	2004-02-14 12:51:16 BDT
Accessed:	2004-02-14 00:00:00 BDT
Created:	2004-02-14 12:50:48 BDT
Last Accessed:	~~~~~ ~~~ ~~~ ~~~ ~~~

Conclusion

Autopsy is capable of recovering and presenting deleted files and directories found within a file system.

6.4 JPEG Search Test

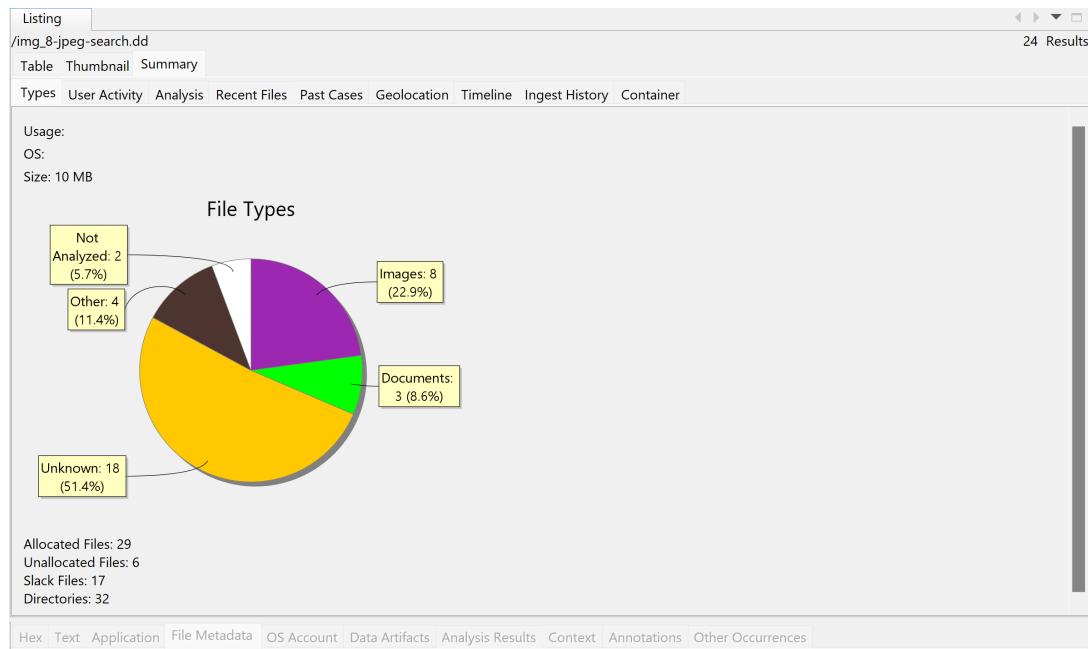
Introduction

The provided test image represents an NTFS file system housing a collection of 10 JPEG images. Among these images are files with misleading extensions, JPEGs embedded within ZIP and Word files, as well as those stored in alternate data streams. The primary aim of utilizing this test image is to evaluate the effectiveness of automated tools designed for locating JPEG images.

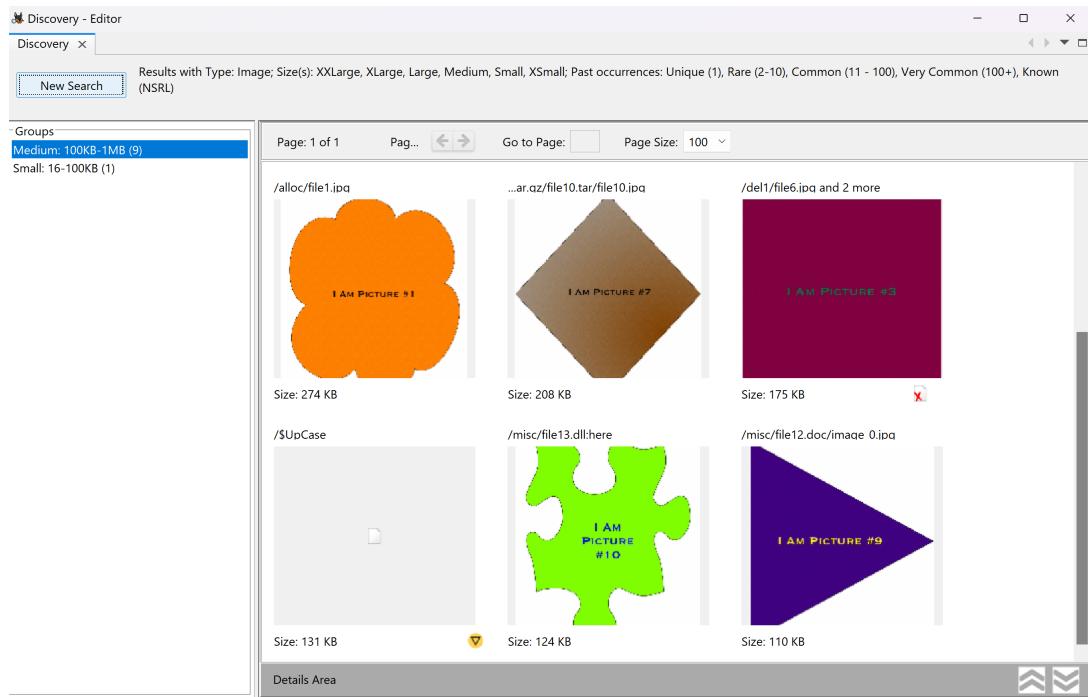
Source

The disk image can be downloaded from here: [Disk Image Download Link](#)
The MD5 of the image is 9bdb9c76b80e90d155806a1fc7846db5.

Outputs from Autopsy



As depicted in the figure below, Autopsy demonstrates its ability to identify JPEG images within the data source, even when the file extension is inaccurate. Furthermore, in cases where a file is labeled with a JPEG extension but does not contain JPEG image data, Autopsy accurately detects the correct file type and displays it accordingly.



JPEG Search Test - Autopsy 4.21.0

Case View Tools Window Help

Add Data Source Images/Videos Communications Geolocation Timeline Discovery Generate Report Close Case Keyword Lists Keyword Search

Data Sources

- 8-jpeg-search.dd_1 Host
 - \$OrphanFiles (0)
 - \$CarveFiles (1)
 - \$Extend (5)
 - SUNalloc (1)
 - alloc (4)
 - archive (5)
 - del1 (3)
 - del2 (3)
 - invalid (5)
 - misc (6)
 - RECYCLER (3)
- System Volume Information (3)

File Views

- Data Artifacts
- Analysis Results
- OS Accounts
- Tags
- Score
- Reports

Listing /img_8-jpeg-search.dd/del2

Table Thumbnail Summary

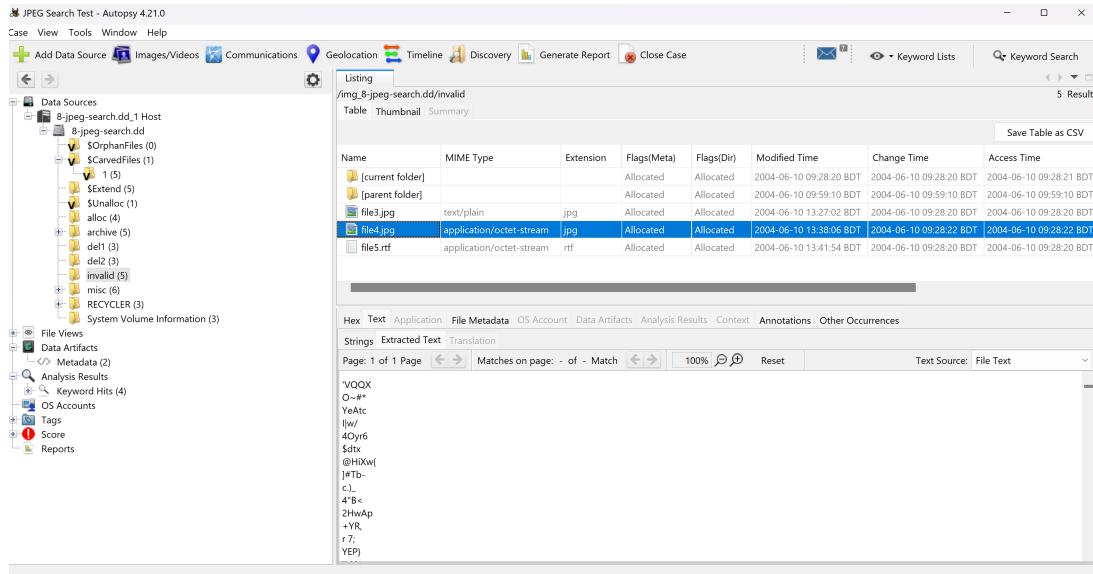
Save Table as CSV

Name	MIME Type	Extension	Modified Time	Change Time	Access Time	Created Time	Size
[current folder]			2004-06-10 09:59:23 BDT	2004-06-10 09:59:23 BDT	2004-06-10 09:59:23 BDT	2004-06-10 09:43:19 BDT	48
[parent folder]			2004-06-10 09:59:10 BDT	2004-06-10 09:59:10 BDT	2004-06-10 09:59:10 BDT	2004-06-10 09:22:22 BDT	56
file7.hmm	image/jpeg	hmm	2004-06-10 12:49:18 BDT	2004-06-10 09:43:44 BDT	2004-06-10 09:43:38 BDT	2004-06-10 09:28:00 BDT	326859

Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences

0% 35% Reset

I AM PICTURE #4



Conclusion

Autopsy excels in identifying JPEG files, whether they have standard or non-standard extensions. It adeptly distinguishes false positives, even when files claim to be JPEGs but lack actual JPEG content. Autopsy can handle partial JPEGs, accurately recognizing valid signatures even without complete headers and footers. It effectively recovers deleted JPEGs, regardless of their extensions, and deals proficiently with deleted JPEGs with incorrect extensions. Autopsy's capabilities extend to locating and extracting embedded JPEGs within archive formats and unconventional file structures. It also excels in recognizing and extracting JPEGs embedded within document files and detecting hidden JPEGs within alternate data streams, making it a valuable tool for forensic analysis.

6.5 NTFS Autodetect Test

Introduction

This test includes four images. One image is a disk image comprising two partitions, with each partition also provided as an individual file. The fourth image is an additional partition image. The objective of this test case is to evaluate the file system detection capabilities of analysis tools. While a typical partition usually contains only one file system, certain file system layouts allow for multiple file systems within a single partition. In this scenario, each partition within the disk image contains two file systems. Specifically, the first partition is formatted for NTFS and Ext2, the second for NTFS and UFS2, and the third for NTFS and UFS1. Both file systems are legitimate and can be mounted within their respective operating systems. The test aims to determine whether a tool will correctly identify the presence of two valid file systems or if it will only display one while concealing the other.

Source

The disk image can be downloaded from here: [Disk Image Download Link](#)
This test case has one 'raw' disk image and two 'raw' partition images. In total, the images are 275 MB, but they compress to under 1 MB.

The MD5 of the disk image is 9225ff95b92311a28b2224e9dc324231,
Partition 1 is 6bd741152ccedd50e12623af5eeba803,
Partition 2 is 0b768efb1011b047c9831c1e00d1706c,
Partition 3 is 5984253cad72d4950c15a9e679139daf

Image Details

The disk image `10-ntfs-disk.dd` contains a DOS partition table with two partitions: `10-ntfs-part1.dd` and `10-ntfs-part2.dd`. Both partitions have a type of 0x07, indicating NTFS file systems. Let's explore the contents of each partition:

- `10-ntfs-part1.dd`: Originating from the first partition within `10-ntfs-disk.dd`, this image was initially formatted as NTFS using Windows XP, housing the `ntfs.txt` file. Subsequently, the partition was formatted as Ext2, resulting in the creation of the `ext2.txt` file. Windows error-checking detected no issues with the NTFS file system.
- `10-ntfs-part2.dd`: Derived from the second partition of `10-ntfs-disk.dd`, this image was initially formatted as NTFS within Windows XP, containing the `ntfs.txt` file. However, a subsequent formatting process converted it to UFS2, inadvertently creating the `ufs1.txt` file. Despite the misformatting, Windows error-checking detected no issues. Notably, this partition was intended to be UFS1 but was erroneously formatted as FreeBSD UFS2.
- Additionally, the image `10-ntfs-part3.dd` does not reside within `10-ntfs-disk.dd` but was created due to the incorrect formatting of `10-ntfs-part2.dd` as UFS2 instead of UFS1. Initially formatted as NTFS using Windows XP, this partition contains the `ntfs.txt` file. Subsequent formatting transformed it into UFS1, resulting in the creation of the `ufs1.txt` file. Once again, Windows error-checking reported no issues with the NTFS file system.

This series of events underscores the complexity and potential errors inherent in partition formatting, while also showcasing the resilience of NTFS file systems under scrutiny.

Outputs from Autopsy

Screenshot 1 (Top): NTFS Autodetect Test - Autopsy 4.21.0

Name	Created Time	Size	Flags(Meta)	Flags(Dir)	Modified Time	Change Time	Access Time	Known
\$CarvedFiles	0000-00-00 00:00:00	0	Allocated	Allocated	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	un
Unalloc_5_49351680_98703360	0000-00-00 00:00:00	49351680	Unallocated	Unallocated	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	un

Screenshot 2 (Bottom): NTFS Autodetect Test - Autopsy 4.21.0

Name	Created Time	Size	Flags(Meta)	Flags(Dir)	Modified Time	Change Time	Access Time	Known
Unalloc_3_0_32256	0000-00-00 00:00:00	32256	Unallocated	Unallocated	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	unknown

Conclusion

Autopsy can correctly detect and display multiple file systems within a partition.

7 References

To download Autopsy, visit: [Autopsy Download Link](#)

For more information about Autopsy, visit: [Autopsy Information Link](#)

For digital forensics test images, visit: [Digital Forensics Test Images Link](#)