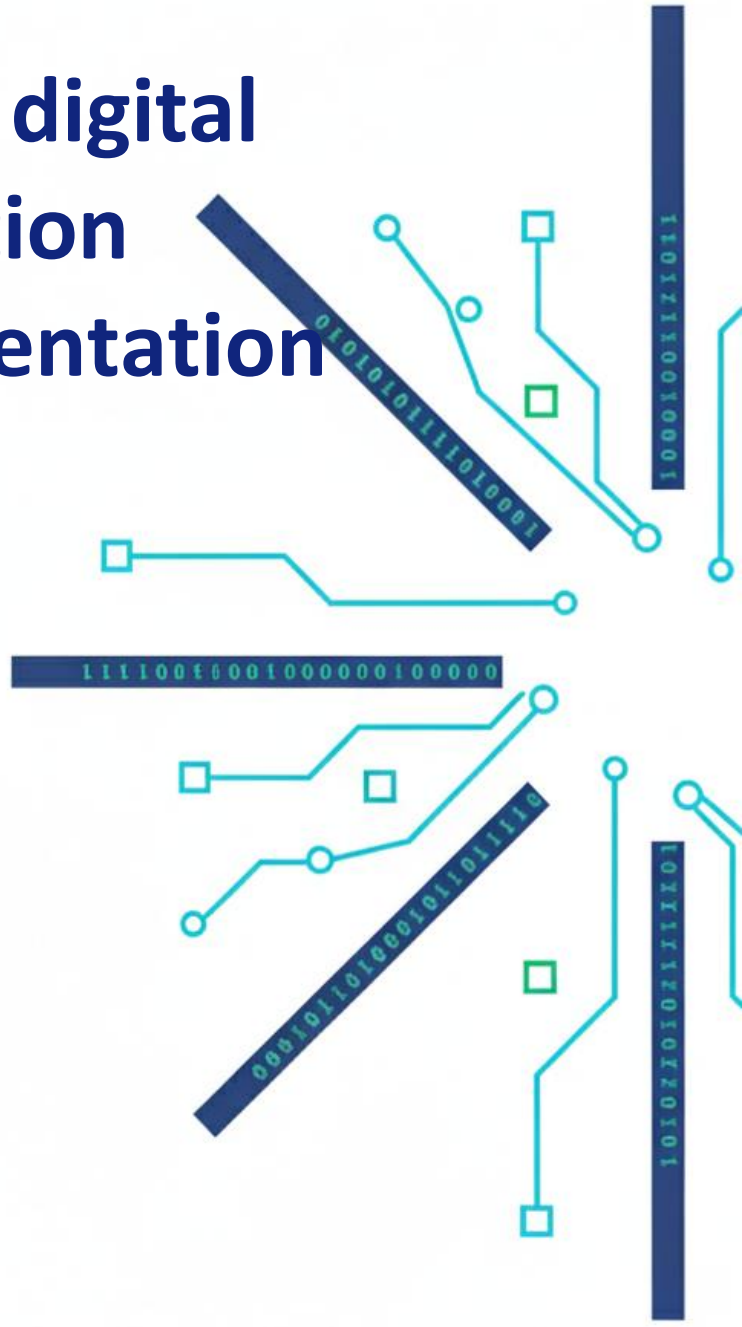
D15 – Model for digital evidence extraction to matrix representation



The project Automatization of digital forensics and incident response (ADFIR) funded by the European Union – Next GenerationEU through the Recovery and Resilience Plan of the Slovak Republic under project No. č. 09-I05-03-V02-00079.

Contents

1 Project description

The project **Automatization of digital forensics and incident response** (hereinafter referred to as **"ADFIR"**) is funded by the **European Union – Next GenerationEU through the Recovery and Resilience Plan of the Slovak Republic** under project No. č. 09-I05-03-V02-00079. This project addresses one of the key challenges in cybersecurity and information security – how to process the massive volume of digital evidence generated during cybersecurity incidents or forensic investigations. Currently, this process is highly demanding in terms of human resources and time. Therefore, automation using machine learning methods can significantly **improve the quality of digital forensic analysis** and reduce the time required to perform it. Overall, this enables security teams to respond more effectively to cyber threats. Main benefits of this project are:

- **Accelerated Resolution of Cybersecurity Incidents**. The project ADFIR introduces automated approaches to collecting, processing, and analyzing digital traces. As a result, security teams can identify the causes of incidents more quickly and adopt effective measures to address them.
- **Reduced Workload for Forensic Analysts**. Routine and time-consuming tasks involved in processing digital traces will be replaced by automated methods. This will allow analysts to focus on more complex cases and strategic decision-making.
- **Higher Quality and Consistency of Outputs**. The use of unified methodologies and tools ensures that the processed digital traces will be more accurate, consistent, and easily verifiable. This significantly reduces the risk of errors caused by human factors.
- **Potential Use in Criminal Proceedings.** The project outputs will be developed in compliance with legal requirements and standards, allowing the digital traces to be accepted as relevant evidence for investigations and court proceedings.

2 Introduction

In the field of digital forensic analysis, processing large amounts of heterogeneous data presents a major challenge, particularly with regard to its further use in automated analytical methods. Although Supertimeline, as the output of tools such as Plaso, provides a comprehensive and information-rich view of the sequence of events in the system, its structure is not directly suitable for the application of machine learning methods or formal analytical approaches.

This document focuses on proposing a model for extracting digital evidence into a matrix representation, which will enable the transformation of the original, predominantly unstructured or textual data into a form suitable for further processing. The aim is to systematically identify relevant attributes, encode them into binary, categorical or numerical variables, whilst minimising information loss during this transformation.

The proposed approach stems from the need to bridge the gap between manual forensic analysis and automated data processing methods. Whilst traditional approaches focus primarily on the interpretation of individual artefacts on a timeline, the aim of this model is to create a representation that enables the effective use of advanced analytical techniques, such as machine learning methods or dimensionality reduction.

This output also forms the basis for subsequent data processing within the ADFIR project, specifically for the aggregation and linking of digital evidence, where the transformed data enables a higher level of abstraction, correlation and interpretation of events.

3 Digital footprint processing

The input data for the proposed model for extracting digital evidence into a matrix representation was obtained using the Plaso tool [7]. This tool enables the aggregation and correlation of forensic artefacts from various sources into a unified timeline, known as a supertimeline. The resulting data structure takes the form of a table consisting of 17 attributes, which are specified in more detail in Table 1.

Attribute	Description	Type
Date	date on which the event occurred	object
Time	time when the event occurred	object
Timezone	time zone	object
MACB	timestamps (Modification, Access, Creation, Birth)	object
Source	source name abbreviation (e.g. REG – register records)	object
Sourcetype	source description	object
Type	timestamp type (e.g. last entry)	object
User	user name (if available) associated with the event	object
Host	host name (if available) associated with the event	object
Short	contains a field with a short description in which the text is stored	object
Desc	the field containing most of the analysed information	object
Version	timestamp version number	int64
Filename	the name of the file associated with the event	object
Inode	inode number of the analysed file	object
Notes	space for storing additional information	object
Format	input module used for analysis	object
Extra	array containing parsed information, which is concatenated and stored here	object

Table 1 - Description of supertimeline attributes

The supertimeline represents a comprehensive and information-rich data source, primarily intended for the purposes of manual digital forensic analysis. In this context, it enables the analyst to reconstruct the chronological sequence of events and identify relevant activities within the system under investigation. Despite its high informative value, however, this form of data is not directly suitable for the application of automated analytical methods, such as machine learning methods, formal conceptual analysis or graph theory-based approaches.

The main problem lies primarily in the data representation of individual attributes. With the exception of the Version attribute, which is represented by a numeric type (int64), all other attributes are stored as the object data type. In practice, this means that the data is predominantly textual, often in the form of unstructured or only partially structured strings. Whilst such a heterogeneous and semantically rich representation provides flexibility for manual interpretation, it significantly complicates further processing within the context of formal and quantitative analytical methods.

Transforming these attributes into a suitable form presents a non-trivial problem, as it requires the identification of relevant features, the extraction of semantic information, and its subsequent encoding into a structured representation. Specifically, this involves converting data into binary, categorical or numerical variables that are compatible with the requirements of machine learning algorithms and other analytical tools. In this process, it is also essential to minimise information loss, which could negatively impact the quality of subsequent analysis.

The aim of this output is therefore to propose a systematic approach to the pre-processing of data from the supertimeline, which will enable its **transformation into a matrix representation** suitable for further processing. The emphasis is placed primarily on preserving as much relevant information as possible, reducing redundancy and ensuring compatibility with selected analytical methods. The result should be a data representation that enables the effective use of advanced data analysis techniques when examining digital evidence.

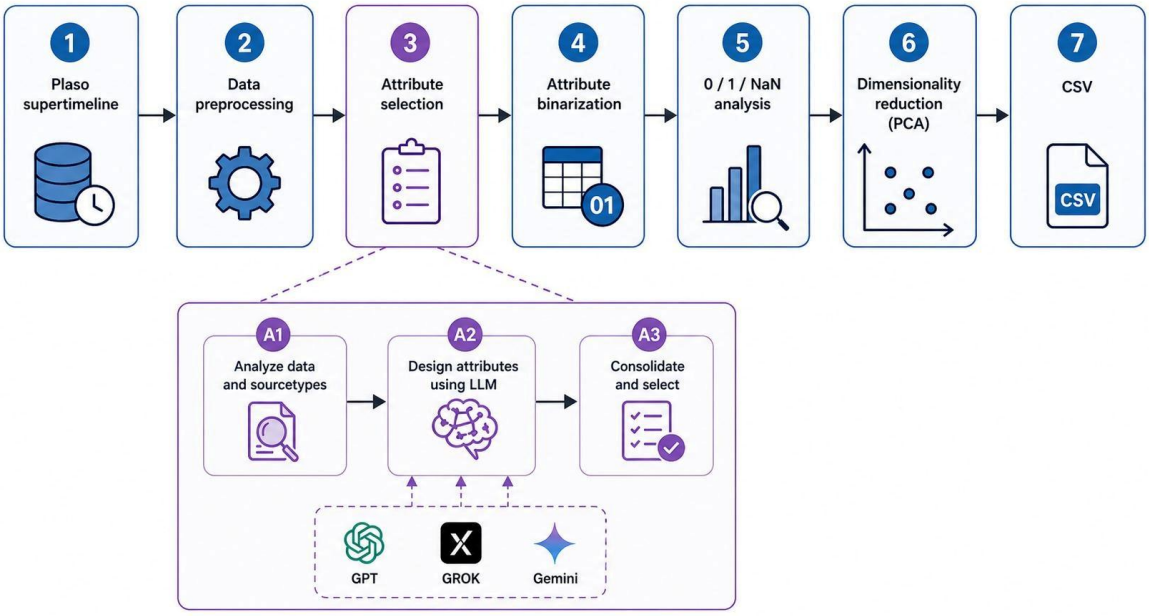


Figure 1 – Digital evidence processing workflow

Figure 1 illustrates the process of processing digital evidence from their acquisition through to the creation of a final dataset suitable for further analysis. The pipeline begins with input data in the form of a Plaso supertimeline, which undergoes a pre-processing phase. This is followed by attribute selection and design, which represents a key step in the entire process.

This step is described in more detail in the lower part of the diagram. It involves the analysis of data and available source types, the generation of candidate attributes using large language models (GPT, Grok, Gemini), and the subsequent consolidation and selection of the final list of attributes.

Once the attributes have been defined, they are extracted and transformed into a binary representation (0/1/NaN), where NaN values indicate that the attribute is irrelevant for the given record. Subsequently, an analysis of the distribution of values and dimensionality reduction are performed using the PCA method. The result of the entire process is a final dataset in CSV format, ready for further processing, such as aggregation or the application of machine learning methods.

4 Feature selection

Attribute selection represents a key phase in the transformation of forensic data into a suitable matrix representation that enables further analytical processing. The aim of this phase is to identify attributes that best capture the nature of the artefacts under investigation whilst retaining information relevant to incident detection. The attribute selection process was carried out in multiple stages, combining statistical data analysis, the use of large language models, and subsequent consolidation of results. An important aspect was ensuring a balance between the complexity of the representation and its interpretability. Particular attention was paid to eliminating redundant and uninformative attributes.

4.1 Analysis of data and source types

In the initial phase of data processing, we focused on analysing the distribution of individual artefacts within the cases under investigation. To this end, basic quantitative statistics were compiled, enabling us to identify the most frequently occurring types of records and data sources. This analysis provided an overview of the dominant artefacts in the data and also served as a basis for the next steps in designing the representation.

The specific results of the analysis are presented in the appendix to this output – "D15 – Model for digital evidence extraction to matrix representation– appendix" in the section "sourcetype_source_counts_summar".

The analysed summary table *sourcetype_source_counts_summar* is used to compare the occurrence of individual **sourcetypes (source types)** across multiple datasets. Each **row** represents a specific sourcetype, whilst **the** individual **columns** represent separate datasets or scenarios. **The value in a cell** indicates the number of occurrences of the given sourcetype in a specific dataset. Colour coding is used for quick interpretation of suitability:

- **green** indicates **suitable source types** (sourcetype) for further processing and matrix representation,
- **red** indicates **unsuitable source types** (sourcetype) for further processing and matrix representation,
- **orange** indicates **borderline source types** (sourcetypes) tied to a specific environment (e.g. a particular operating system).

The evaluation methodology is based on a combination of frequency of occurrence and consistency across datasets. Ensuring the representativeness of attributes for various types of forensic scenarios was also an important criterion.

4.1.1 Analysis of suitable source types

Source types suitable for further processing and matrix representation can be divided into two groups, which differ in terms of frequency of occurrence.
Source types (sourcetype) with a high frequency of occurrence across all datasets. These sources are suitable because they are consistently present, rich in information, and of high value for forensic investigation. Among the most significant are those sources that achieve maximum or near-maximum values across all datasets:

- **FILE | File entry shell item, FILE | File stat, FILE | NTFS USN change, FILE | NTFS file stat** – file system metadata with high availability and significant content of temporal and system information,
- **EVT | WinEVTX** – event logs, which represent a key source for event reconstruction,
- **REG | Registry Key** – entries in the Windows operating system registry that represent generic registry entries without further categorisation or specific context.

The second type of source (sourcetype) consists of sources with a lower frequency of occurrence but high consistency. Their significance for forensic investigation lies in the fact that they provide specific yet reusable information. These are the sources which, although not occurring in large numbers, are present in most datasets:

- **WEBHIST artefacts** (**WEBHIST | MSIE WebCache (container, cookies, records), WEBHIST | Chrome (Cache, Cookies, History)**) – important for analysing user activity,
- **REG | Task Cache Registry Key** – artefacts of scheduled tasks relevant to the persistence of attacks,
- **REG | Registry Key – Service, REG | Service/Driver Configuration Registry Key, REG | BagMRU Registry Key**) – entries in the Windows operating system registry that provide structured and context-rich information tied to a specific area of the system, such as the configuration of services and drivers (persistence, system startup) or user activity (e.g. BagMRU – navigation within the file system). These artefacts have greater forensic value as they allow for a more accurate interpretation of system or user behaviour,
- **LNK | Windows Shortcut** – provides information on user access to files and applications, occurring across multiple datasets with a relatively stable frequency,
- **LOG | WinPrefetch** – although it does not appear in all datasets (notably missing in some server scenarios), it is a significant source for analysing application launches,

- **OLECF | OLECF Item / Dest list entry** – these represent artefacts related to the user's work with documents (e.g. recent files), appearing in multiple datasets and providing supplementary information on user activity.

4.1.2 Analysis of unsuitable source types

These sources are characterised by a very low frequency of occurrence, significant inconsistency across datasets, or limited interpretative value. In many cases, these are artefacts (sourcetypes) that appear only in a single specific scenario or in a specific system configuration, which significantly reduces their usefulness for a general matrix representation. Typical examples include artefacts such as **AMCACHEPROGRAM | Amcache Programs Registry Entry**, which appear in only a very small number of datasets, or **REG | Registry Key – RDP Connection**, which is present in just one dataset. Similarly, **PLIST | Plist file** represents a platform-specific artefact, appearing only marginally in the analysed data.

Another group consists of logs and metadata artefacts with limited context, such as **LOG | Google Drive Sync Log** or **META | Open XML Metadata**, which, although they may appear in greater numbers in some datasets, their occurrence is strictly tied to a specific application or scenario (e.g. cloud synchronisation or document processing). Similarly, artefacts of the type **OLECF | OLECF Document Summary Info and OLECF | OLECF Summary Info** provide only supplementary metadata information without a direct link to security-relevant events, which reduces their significance in the detection of cyber security incidents.

A specific category is represented by artefacts related to binary files, such as **PE | PE Compilation time** or **PE | PE Import Time**, which, whilst informative from the perspective of malicious code analysis, exhibit a high degree of imbalance – in some datasets they are completely absent, whilst in others they occur in high numbers.

Artifacts with minimal occurrence, such as **RECBIN | Recycle Bin**, are also unsuitable; whilst they may contain relevant information, their sporadic occurrence (ranging from a few to tens of records) prevents their effective use. Similarly, **WEBHIST | Chrome Autofill** or **WEBHIST | MSIE WebCache partitions records** are heavily dependent on specific user behaviour and the type of web browser.

Overall, it can be concluded that these sources do not contribute to a robust and consistent representation of the data. Their inclusion would lead to increased noise and potential bias in the of analytical models. For this reason, these artefacts (sourcetypes) were excluded during the attribute selection process and were not included in the final matrix representation.

4.1.3 Analysis of boundary source types

AMCACHE artefacts

Artifacts (sourcetype) of the AMCACHE type (**AMCACHE | Amcache Registry Entry**) represent a significant source of information about applications running in the Windows operating system environment. They contain records of .exe files, including their paths, digital fingerprints (hashes) and timestamps. Their main advantage is the ability to capture historical application launches, even if the files themselves are no longer present on the system. However, in the analysed datasets, they do not appear consistently in all cases, which is related to differences in operating system versions and data collection methods. In some scenarios, these artefacts may be unavailable or incomplete. Nevertheless, they provide high forensic value, particularly in the identification of malicious software. Their use is particularly suitable when analysing compromised systems. From a methodological perspective, they are suitable as a supplementary source of information. However, their lower versatility limits their use in global models. They are therefore classified as boundary artefacts.

JOB (Scheduled Tasks) artefacts

Artifacts (sourcetype) of the JOB type (**JOB | Windows Scheduled Task Job and JOB | Windows Scheduled Task Trigger**) represent scheduled tasks in the Windows operating system. They provide information about automated processes that are executed based on time-based or system triggers. Their significance lies primarily in the detection of attackers' persistence mechanisms. In datasets, these artefacts appear only in cases where scheduled tasks were actively utilised. This means that their occurrence is heavily dependent on the specific scenario. Nevertheless, they may contain highly valuable information about malicious activities. We distinguish between the job definitions themselves and their trigger mechanisms. Their combined analysis allows us to reconstruct the timing of attacks. A disadvantage is their inconsistent presence across datasets. They are therefore particularly suitable for specialised analyses.

LOG artefacts

Logs represent one of the most important categories of forensic data. They include system logs, device installation logs (**LOG | Setup API Log**) and resource usage monitoring. These artefacts (sourcetypes) provide a chronological record of events in the system. Their advantage lies in their high informational value and relatively frequent occurrence in datasets. On the other hand, their structure and content vary depending on the system configuration. Some logs, such as **LOG | System Resource Usage Monitor**, are only available in certain versions of the operating system— nd Windows. This results in their inconsistent occurrence. Logs can also be voluminous and contain noise. Their effective use therefore requires pre-processing and filtering. Despite these limitations, they are key to incident reconstruction. In the context of this analysis, some logs are classified as borderline due to their variability.

PE (Portable Executable) artefacts

Artifacts (sourcetype) of **the PE | PE/COFF file type** provide information about executable files on the system. They contain data such as compilation time, imported libraries, and the structure of the binary file. This information is important for the analysis of malicious software. However, their occurrence in datasets is contingent upon the presence of exe files. In some scenarios, therefore, they do not occur at all. Nevertheless, they are of great value in reverse engineering and the attribution (identification) of attacks. They enable the identification of suspicious or non-standard binary files. The disadvantage is their limited versatility. Not all dataset scenarios contain relevant binary artefacts. They are therefore more suitable for specialised analyses. In global models, they play a supplementary role.

REG artefacts

Artifacts (sourcetype) stored within the Windows operating system registry constitute a vast and highly significant category of forensic data in the Windows operating system environment. They contain information regarding system configuration, user activity and application launches. Artifacts (sourcetype) such as **the REG | AppCompatCache Registry Key** or **the REG | Background Activity Moderator Registry Entry** provide valuable data on programme execution. However, specific keys, such as **the REG | Microsoft Outlook MRU Registry Key**, are tied to particular applications. This results in their low occurrence in datasets.

Artifacts (sourcetype) stored within the Windows operating system registry are highly detailed but also fragmented. Their analysis requires knowledge of the registry's structure. An advantage is their ability to capture historical system behaviour. A disadvantage is their dependence on a specific operating system and its configuration. Therefore, not all keys are available in all datasets. In this analysis, some artefacts (sourcetypes) found in the Windows operating system registry are therefore classified as borderline.

WEBHIST artefacts

Web artefacts represent user activity in web browsers. In this case, these are primarily Internet Explorer artefacts (**WEBHIST | MSIE Cache File URL record**). They contain information about visited URLs, cache files and sessions. Their significance lies in the reconstruction of user behaviour. However, these artefacts are rarely found in modern datasets. The reason is the decline in the use of Internet Explorer. Therefore, they are available in only a limited number of datasets. Nevertheless, they can be very valuable in some cases. Their analysis can reveal communication with suspicious domains. The disadvantage is their low versatility. In the context of this study, they are therefore classified as borderline.

4.2 Attribute design using LLM

The second phase of selecting suitable attributes for the matrix representation involves attribute design using large language models (LLMs). Based on the previous analysis, we designed an initial prompt for large language models, the aim of which was to automatically generate proposals for binary attributes representing individual forensic records. The prompt for these language models was formulated as follows:

Initial prompt:
"Load the file containing a sample of 5 datasets from CTF for the purposes of digital forensic analysis. The dataset contains the following columns: datetime, source, MACB, filename, desc, extra, dataset. Datetime is a timestamp, source is the data source, MACB represents timestamps for the source data, dataset is the identifier for the specific dataset, and the desc and extra fields contain the actual data. This is the WinEVTX data source. Based on this data and your knowledge of the output from the Plaso *tool for this data source, generate a proposal for binary attributes that would represent this type of record. For each binary attribute, provide the name, a short description of the attribute, and the method of extraction from the individual fields in Python (for Jupyter Notebook)."*

Based on this initial prompt, specialised prompts were subsequently created for individual source types (*sourcetype*), with a uniform structure that reflected the specific characteristics of each data source. As an example, here is the prompt for the *WinEVTX* source (Windows Event Logs):

"WinEVTX
You have access to a dataset of outputs from the PLASO tool for the WinEVTX (Windows Event Logs) data source. Each row represents a single forensic record containing the following fields:

- datetime: event timestamp,
- source: source type (e.g. EVT),
- MACB: operation timestamps (Modified, Accessed, Changed, Birth),
- filename: name of the event source file,
- desc: event description (including text from the Windows Event Log),
- extra: additional attributes such as message_identifier, recovered, sha256,
- dataset: identifier of a specific CTF dataset.

Propose **a set of binary attributes** suitable for representing these records for the purposes of machine learning or forensic classification. For each attribute, specify:

1. **Attribute name** (in the style of variable names),
2. **A brief description** of what the attribute means,
3. **Extraction method** in Python suitable for Jupyter Notebook.

Focus on attributes that could:

- characterise the type of event,
- identify important or suspicious behaviour,
- be related to anomalies or restored records,

- indicate specific logs (e.g. system, network),
- reflect typical structures in PLASO/EVTX data."

Prompts were prepared in a similar manner for other types of data sources (*sourcetypes)*:

- EVT – WinWVTX,
- FILE - File Entry Shell Item,
- FILE - File Stat,
- FILE – NTFS File Stat,
- FILE – NTFS USN Change,
- LNK – Windows Shortcut,
- LOG – WinPrefetch,
- OLECF – OLECF Dest List Entry,
- OLECF – OLECF Item,
- WEBHIST – Chrome Cache,
- WEBHIST – Chrome Cookies,
- WEBHIST – Chrome History,
- WEBHIST – MSIE WC Container rec,
- WEBHIST – MSIE WC Containers re,
- WEBHIST – MSIE WebCache cookies,
- REG – BagMRU Registry Key,
- REG – MRUList Registry Key,
- REG – MRUListEx Registry Key,
- REG_Registry Key – BagMRU,
- REG_Registry Key – MRUList,
- REG_Registry Key – MRUListEx,
- REG_Registry Key – Network Driv
- REG_Registry Key – Run Key,
- REG_Registry Key – Service,
- REG_Registry Key – Typed URLs,
- REG_Registry Key – USB Entries,
- REG_Registry Key – User Account,
- REG_Registry Key – UserAssist,
- REG_Registry Key – Winlogon,
- REG_Registry Key Shutdown Entry,
- REG_Registry Key,
- REG_Run_Run – Once Registry Key,
- REG_Service_Driver Configuration,
- REG_Shutdown Registry Key,
- REG_Task Cache Registry Key,
- REG_Task Cache,
- REG_Typer URLs Registry Key,
- REG_USB Registry Key,
- REG_USBStor Registry Key.

- REG_User Account Information Registry Key,
- REG_UserAssist Registry Key,
- REG_Winlogon Registry Key.

The prompts defined in this way were subsequently used to generate attribute proposals using several large language models, specifically: ChatGPT 4o[1] , Gemini 2.5[2] and Grok 2[3] . The aim was to compare the outputs of the individual models and identify common patterns in the proposed attributes.

This step resulted in a set of candidate binary attributes for each analysed *sourcetype*, with the number varying depending on the data type and the model used. A summary of the number of attributes for individual combinations of sources and models is given in Table 2.

Sourcetype	Total number of attributes	Number of attributes selected using Chat/GPT 4	Number of attributes selected using Gemini	Number of attributes selected using Grok
EVT - WinEVTX	**25**	**10**	**12**	**11**
FILE - File Entry Shell Item	**21**	**10**	**13**	**15**
FILE - File Stat	**20**	**14**	**14**	**15**
FILE - NTFS File Stat	**25**	**13**	**15**	**18**
FILE - NTFS USN Change	**24**	**10**	**12**	**14**
LNK - Windows Shortcut	**23**	**7**	**12**	**12**
LOG - WinPrefetch	**14**	**4**	**5**	**10**
OLECF - OLECF Dest List Entry	**13**	**10**	**7**	**8**
OLECF - OLECF Item	**16**	**10**	**11**	**12**
WEBHIST - Chrome Cache	**15**	**10**	**11**	**13**
WEBHIST - Chrome Cookies	**21**	**10**	**11**	**14**
WEBHIST - Chrome History	**19**	**10**	**11**	**11**
WEBHIST - MSIE WC Container rec	**20**	**9**	**7**	**9**
WEBHIST - MSIE WC Containers re	**20**	**8**	**10**	**15**

[1] ChatGPT 4o model; model information available on the website: https://developers.openai.com/api/docs/models/gpt-4o
[2] Gemini 2.5 model, model information available on the website: https://ai.google.dev/gemini-api/docs/models/gemini-2.5-flash
[3] Grok 2 model, model information available on the website: https://x.ai/news/grok-2

WEBHIST - MSIE WebCache cookies	**20**	**8**	**9**	**16**
REG - BagMRU Registry Key	**18**	**10**	**7**	**10**
REG - MRUList Registry Key	**21**	**10**	**6**	**10**
REG_MRUListEx Registry Key	**10**	**7**	**5**	**8**
REG_Registry Key – BagMRU	**13**	**6**	**6**	**10**
REG_Registry Key – MRUList	**15**	**8**	**6**	**10**
REG_Registry Key – MRUListEx	**12**	**5**	**4**	**8**
REG_Registry Key – Network Drive	**15**	**6**	**4**	**8**
REG_Registry Key – Run Key	**16**	**9**	**8**	**11**
REG_Registry Key – Service	**21**	**10**	**5**	**10**
REG_Registry Key – Typed URLs	**10**	**5**	**5**	**10**
REG_Registry Key - USB Entries	**12**	**8**	**9**	**10**
REG_Registry Key - User Account	**11**	**6**	**5**	**10**
REG_Registry Key - UserAssist	**19**	**7**	**7**	**10**
REG_Registry Key - Winlogon	**18**	**5**	**6**	**8**
REG_Registry Key Shutdown Entry	**16**	**6**	**6**	**10**
REG_Registry Key	**16**	**5**	**10**	**8**
REG_Run_Run Once Registry Key	**21**	**10**	**11**	**10**
REG_Service_Driver Configuration	**21**	**9**	**9**	**13**
REG_Shutdown Registry Key	**19**	**5**	**7**	**10**
REG_Task Cache Registry Key	**21**	**6**	**8**	**10**
REG_Task Cache	**11**	**6**	**6**	**10**
REG_Typed URLs Registry Key	**12**	**7**	**7**	**8**
REG_USB Registry Key	**17**	**9**	**9**	**10**

REG_USBStor Registry Key	**14**	**5**	**6**	**8**
REG_User Account Information Re	**15**	**7**	**6**	**8**
REG_UserAssist Registry Key	**13**	**7**	**7**	**10**
REG_Winlogon Registry Key	**20**	**8**	**10**	**10**
TOTAL	**769**	**355**	**370**	**477**

Table 2 - Summary of the number of attributes for individual combinations of sources

4.3 Consolidation and selection of attributes

In the third and final phase of selecting suitable attributes for the matrix representation, we proceeded to implement the extraction of the proposed attributes. The original intention was that only those attributes identified by at least two independent language models would be included in the final processing. This approach was intended to ensure a higher degree of robustness and eliminate less relevant or randomly generated attributes.

Despite this reduction, however, it turned out that the resulting number of attributes remained high, which led to increased computational complexity and complications in further data processing. Furthermore, it became apparent that the manual implementation of extraction rules for a large number of attributes is time-consuming and prone to errors.

For these reasons, we proceeded to the next step, which was data dimensionality reduction. For this purpose, we selected the *Principal Component Analysis (PCA)* method, which allows the original attribute space to be transformed into a lower-dimensional space whilst preserving as much of the data's variability as possible. This approach enabled more efficient data processing whilst eliminating redundancy between individual attributes.

5 Data dimension reduction

5.1 Introduction to data dimensionality reduction

In the field of digital forensic analysis, high-dimensional data is frequently handled, such as system event records, logs, or forensic artefacts extracted from various sources. This data may contain tens to hundreds of attributes, with the number increasing further when data is aggregated into a supertimeline format. Data obtained in this way can lead to increased computational complexity, model overfitting and reduced interpretability of results. Dimension reduction methods are therefore used, with the aim of reducing the number of attributes whilst retaining as much relevant information as possible [4].

Dimension reduction can generally be divided into two main groups:

- supervised methods and
- unsupervised methods.

5.1.1 Supervised dimension reduction methods

Supervised methods utilise information about the target variable and aim to select or transform attributes so that they contribute most effectively to the prediction.

The most commonly used approaches include:

- **Decision trees** – naturally perform attribute selection during the learning process, with less relevant attributes being eliminated.
- **Logistic regression with regularisation** – the use of L1 regularisation leads to model sparsification, which results in the elimination of less significant attributes [3].
- **Linear Discriminant Analysis (LDA)** – projects data into a lower-dimensional space to maximise the separation between classes.

In the context of digital forensic analysis, these approaches are used, for example, in event classification or the identification of malicious activities; however, their use is contingent upon the availability of annotated data [3].

5.1.2 Unsupervised dimension reduction methods

Unlike supervised approaches, unsupervised methods do not use class information. Their aim is to uncover the structure of the data or remove redundancy between attributes.

The main approaches include:

- **Clustering** – methods such as k-means enable the identification of groups of similar events or behaviours within the data.

- **Data visualisation methods (e.g. t-SNE)** – these are mainly used to visualise high-dimensional forensic data.
- **Principal Component Analysis (PCA)** – is one of the most widely used methods of dimensionality reduction. It transforms the original attributes into a new space of principal components that capture the maximum variability of the data [5,6].

5.2 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is one of the most significant dimension reduction techniques within unsupervised approaches. Its main advantage is the ability to transform the original attributes into a smaller number of principal components that capture most of the data variability without the need for a target variable [5,6]. PCA also helps to eliminate multicollinearity between attributes, as the resulting components are orthogonal to one another and therefore uncorrelated.

In this analysis, there is no target variable available that would allow the application of supervised dimension reduction methods. For this reason, it is necessary to use unsupervised approaches that do not require labelled data. PCA is a suitable choice in this context, as it allows for the effective removal of redundancy between attributes whilst preserving the essential information contained in the data [6].

For this reason, dimension reduction using PCA was applied separately for individual artefacts, or source types. This approach allows for a better capture of the specific characteristics of individual data types, as different forensic artefacts may contain distinct distributions of attributes and patterns of behaviour. If PCA were applied to all artefacts together, the resulting components could be distorted, as dominant patterns from one type of data could suppress significant characteristics of other artefacts. Such an approach could lead to a loss of relevant information and a reduction in the quality of the subsequent analysis.

When PCA was applied to the analysed dataset, the number of attributes was reduced from the original 769 to 218 principal components, whilst a substantial portion of the data's variability was retained. This reduction helps to lower the computational complexity of subsequent analyses and, at the same time, improves the clarity of the data.

Based on the above properties, the PCA method was selected as the primary dimension reduction tool in this work.

5.2.1 Design and implementation of the PCA model

The process of dimensionality reduction using the Principal Component Analysis (PCA) method consisted of several sequential steps, the aim of which was to transform the original high-dimensional dataset into a lower-dimensional space whilst retaining as much information as possible.

In the first step, the input data was standardised. This step is essential, as PCA is sensitive to the scale of individual attributes. Without standardisation, attributes with a wider range of values would have a disproportionately large influence on the resulting components. The data were therefore transformed to have a mean of zero and a standard deviation of one.

Subsequently, a PCA model was applied to the data prepared in this way, without restricting the number of components. The aim of this step was to analyse the distribution of explained variability among the individual principal components and to gain an overview of how many components are needed to retain a substantial portion of the information contained in the data.

Based on the results, a so-called scree plot was created, which displays the proportion of explained variability for individual components. This graph was used to determine the optimal number of components, with a threshold of 95% cumulative explained variability selected as the criterion. Based on this analysis, the final number of components used in further processing was determined.

After determining the optimal number of components, the PCA model was reapplied, this time restricted to the selected number of principal components. The result of this transformation was a new feature space in which the individual components represent linear combinations of the original attributes.

In the next step, the loadings were analysed, which express the contribution of individual original attributes to each principal component. Based on the absolute values of these loadings, a ranking of attributes was compiled for each component according to their significance. This ranking makes it possible to identify which attributes contribute most to the variability captured by a specific component.

The procedure designed in this way enables not only effective data dimensionality reduction, but also the preservation of the model's interpretability through the analysis of the contributions of the original attributes to individual components.

5.2.2 PCA Results

This chapter presents the results of dimensionality reduction using Principal Component Analysis (PCA) applied to data from individual forensic artefacts in the supertimeline. PCA was performed separately for each artefact to capture the specific characteristics and variability of individual data types. For each artefact, the optimal number of principal components was determined based on a scree plot and a threshold of 90–95% cumulative explained variance. Subsequently, the loadings of the original attributes were analysed and rankings of the most important attributes were compiled for each artefact, providing an overview
of which attributes contribute most to data variability. The results are presented in tables (Table 3 – Table 7) separately for individual data sources (CTF), which allows the significance of attributes to be compared across different datasets.

Artifact	Original attributes	Number of components	% of explained variability	Number of selected attributes
WinEVTX	27	15	93.7%	17
File_entry_shell_item	23	5	92.2%	8
NTFS_file_stat	27	11	93.1%	15
File_stat	22	11	95.9%	15
Windows_Shortcut	25	4	93.6%	7
Amcache_Registry_Entry				
AppCompatCache_Registry_Key				
BagMRU_Registry_Key	20	3	93.7%	5
Chrome_Cache	17	8	94.8%	10
Chrome_Cookies	22	7	93.8%	8
Chrome_History	21	4	94.8%	5
MRUList_Registry_Key	23	1	100%	1
MRUListEx_Registry_Key	12	4	100%	4
MSIE_WebCache_container_record	22	7	97.2%	10
MSIE_WebCache_containers_record	23	5	95.3%	8
MSIE_Cache_File_URL_record				
Network_Connection_Registry_Key				
OLECF_Dest_list_entry	15	6	97.4%	7
OLECF_Item	17	1	100%	1
OLECF_Summary_Info	2	1	100%	1
Open_XML_Metadata				
PE_Compilation_time				
PE_COFF_file	2	1	100%	1
Registry_Key	16	11	93.7%	12
Registry_Key___BagMRU				
Registry_Key___UserAssist				
Registry_Key___Run_Key				
Registry_Key___MRUList				
Registry_Key___MRUListEx				
Registry_Key___Typed_URLs				
Run_Run_Once_Registry_Key	22	3	99.9%	8
System	2	1	100%	1
Service_Driver_Configuration_Registry_Key				

Setup_API_Log				
Shutdown_Registry_Key				
Task_Cache_Registry_Key	20	5	91.7%	6
USB_Registry_Key	16	1	100%	1
User_Account_Information_Registry_Key				
WinPrefetch	16	8	97.6%	10
Winlogon_Registry_Key				

Table 2 - PCA for the Magnet_CTF_2019_Windows_Desktop case

Artifact	Original attributes	Number of components	% of explained variance	Number of selected attributes
WinEVTX	27	15	91.7%	17
File_entry_shell_item	23	5	94.4%	7
NTFS_file_stat	27	11	95.4%	15
File_stat	22	11	96.9%	15
Windows_Shortcut	25	5	95.5%	9
Amcache_Registry_Entry				
AppCompatCache_Registry_Key	2	1	100%	1
BagMRU_Registry_Key	20	4	92.9%	6
Chrome_Cache	17	8	93.6%	10
Chrome_Cookies	22	9	90.6%	11
Chrome_History	21	5	89.4%	6
MRUList_Registry_Key	23	3	100%	4
MRUListEx_Registry_Key	12	2	100%	2
MSIE_WebCache_container_record	22	8	98.1%	10
MSIE_WebCache_containers_record	23	4	94.0%	5
MSIE_Cache_File_URL_record	2	1	100%	1
Network_Connection_Registry_Key	2	1	100%	1
OLECF_Dest_list_entry	15	3	92.0%	4
OLECF_Item	17	3	100%	3
OLECF_Summary_Info	2	1	100%	1
Open_XML_Metadata	2	1	100%	1
PE_Compilation_time				
PE_COFF_file	2	1	100%	1

Registry_Key	16	13	97.2%	14
Registry_Key___BagMRU				
Registry_Key___UserAssist				
Registry_Key___Run_Key				
Registry_Key___MRUList				
Registry_Key___MRUListEx				
Registry_Key___Typed_URLs				
Run_Run_Once_Registry_Key	22	4	92.7%	8
System	2	1	100%	1
Service_Driver_Configuration_Registry_Key	2	1	100%	1
Setup_API_Log	2	1	100%	1
Shutdown_Registry_Key	8	1	100%	1
Task_Cache_Registry_Key	20	4	100%	4
USB_Registry_Key	16	2	100%	2
User_Account_Information_Registry_Key	2	1	100%	1
WinPrefetch	16	8	95.6%	11
Winlogon_Registry_Key	2	1	100%	1

Table 4 - PCA for the Magnet_CTF_2020_Windows desktop case

Artifact	Original attributes	Number of components	% of explained variance	Number of selected attributes
WinEVTX	27	16	95.9%	17
File_entry_shell_item	23	5	95.1%	8
NTFS_file_stat	27	10	95.5%	14
File_stat	22	11	92.7%	16
Windows_Shortcut	25	5	97.1%	8
Amcache_Registry_Entry	2	1	100%	1
AppCompatCache_Registry_Key				
BagMRU_Registry_Key	20	3	100%	4
Chrome_Cache	17	8	94.7%	9
Chrome_Cookies	22	10	97.8%	11
Chrome_History	21	7	98.3%	8
MRUList_Registry_Key	23	2	100%	5
MRUListEx_Registry_Key	12	1	100%	1
MSIE_WebCache_container_record	22	5	94.5%	6

MSIE_WebCache_containers_record	23	3	97.2%	5
MSIE_Cache_File_URL_record				
Network_Connection_Registry_Key				
OLECF_Dest_list_entry	15	5	95.9%	7
OLECF_Item	17	1	100%	1
OLECF_Summary_Info				
Open_XML_Metadata				
PE_Compilation_time				
PE_COFF_file	2	1	100%	1
Registry_Key	16	11	97.8%	12
Registry_Key___BagMRU				
Registry_Key___UserAssist				
Registry_Key___Run_Key				
Registry_Key___MRUList				
Registry_Key___MRUListEx				
Registry_Key___Typed_URLs				
Run_Run_Once_Registry_Key	22	3	98.3%	10
System	2	1	100%	1
Service_Driver_Configuration_Registry_Key				
Setup_API_Log				
Shutdown_Registry_Key				
Task_Cache_Registry_Key	20	5	94.4%	6
USB_Registry_Key	16	1	100%	1
User_Account_Information_Registry_Key				
WinPrefetch	16	9	97.3%	12
Winlogon_Registry_Key				

Table 3 - PCA for the Magnet_CTF_2022_Windows laptop

lArtefact	Original attributes	Number of components	% of explained variance	Number of selected attributes
WinEVTX	27	18	95.0%	22
File_entry_shell_item	23	3	94.7%	6
NTFS_file_stat	27	10	95.9%	15

File_stat	22	10	94.4%	15
Windows_Shortcut	25	3	96.7%	7
Amcache_Registry_Entry				
AppCompatCache_Registry_Key				
BagMRU_Registry_Key				
Chrome_Cache				
Chrome_Cookies				
Chrome_History				
MRUList_Registry_Key				
MRUListEx_Registry_Key				
MSIE_WebCache_container_record	22	4	98.7%	9
MSIE_WebCache_containers_record	23	4	96.3%	6
MSIE_Cache_File_URL_record				
Network_Connection_Registry_Key				
OLECF_Dest_list_entry	15	3	100%	3
OLECF_Item	17	1	100%	1
OLECF_Summary_Info				
Open_XML_Metadata				
PE_Compilation_time				
PE_COFF_file				
Registry_Key	16	12	97.7%	13
Registry_Key___BagMRU	12	3	95.0%	4
Registry_Key___UserAssist	18	3	100%	4
Registry_Key___Run_Key	2	1	100%	1
Registry_Key___MRUList				
Registry_Key___MRUListEx				
Registry_Key___Typed_URLs				
Run_Run_Once_Registry_Key				
System	2	1	100%	1
Service_Driver_Configuration_Registry_Key				
Setup_API_Log				
Shutdown_Registry_Key				
Task_Cache_Registry_Key				
USB_Registry_Key				
User Account Information Registry Key				

WinPrefetch				
Winlogon_Registry_Key				

Table 6 - PCA for the SSS DC case

Artifact	Original attributes	Number of components	% of explained variability	Number of selected attributes
WinEVTX	27	18	93.5%	20
File_entry_shell_item	23	4	93.0%	5
NTFS_file_stat	27	11	95.8%	15
File_stat	22	11	94.5%	16
Windows_Shortcut	25	4	100%	6
Amcache_Registry_Entry				
AppCompatCache_Registry_Key				
BagMRU_Registry_Key				
Chrome_Cache				
Chrome_Cookies				
Chrome_History				
MRUList_Registry_Key				
MRUListEx_Registry_Key				
MSIE_WebCache_container_record	22	1	100%	1
MSIE_WebCache_containers_record				
MSIE_Cache_File_URL_record				
Network_Connection_Registry_Key				
OLECF_Dest_list_entry	15	4	95.1%	5
OLECF_Item	17	1	100%	1
OLECF_Summary_Info				
Open_XML_Metadata				
PE_Compilation_time	2	1	100%	1
PE_COFF_file				
Registry_Key	16	12	95.2%	13
Registry_Key___BagMRU	12	3	97.2%	4
Registry_Key___UserAssist	18	2	100%	2
Registry_Key___Run_Key	2	1	100%	1
Registry_Key___MRUList	2	1	100%	1
Registry_Key___MRUListEx	2	1	100%	1
Registry_Key___Typed_URLs	2	1	100%	1

Run_Run_Once_Registry_Key				
System	2	1	100%	1
Service_Driver_Configuration_Registry_Key				
Setup_API_Log				
Shutdown_Registry_Key				
Task_Cache_Registry_Key				
USB_Registry_Key				
User Account Information Registry Key				
WinPrefetch	16	8	96.5%	12
Winlogon_Registry_Key				

Table 4 – PCA for the SSS Desktop case

The results are presented in five separate tables (Table 3 – Table 7), with each table corresponding to a specific source type (sourcetype) to which PCA was applied independently. This approach allows for a more detailed capture of the specific characteristics of individual data types, as different source types contain distinct sets of artefacts and attributes.

In some cases, the tables contain blank values. These may occur for two reasons:

- either the artefact was not present in the specific source type, or
- it was present, but it was not possible to apply PCA to it. This situation arises particularly in cases where the attributes of a given artefact exhibit zero variability (i.e. contain constant values), and therefore do not contribute to explaining the variability of the data and it is not possible to extract principal components from them.

This incompleteness therefore does not represent a processing error, but is a consequence of the nature of the forensic data analysed.

The results presented in the tables show that the application of PCA led to a significant reduction in the number of attributes whilst maintaining a high degree of explained variability in all analysed artefacts. In most cases, it was possible to represent the original data using a significantly smaller number of principal components, whilst the model retained more than 90% of the information contained in the data.

However, the degree of reduction varies between individual artefacts, suggesting differing levels of redundancy and complexity in their attributes. Artefacts with a higher degree of reduction likely contain more correlated or redundant variables, whilst artefacts with a lower reduction rate require a greater number of components to capture variability, which points to their greater informational diversity.

Analysis of attribute loadings also enabled the identification of key attributes that contribute most to data variability. These attributes can be considered the most significant from the perspective of further processing, for example in classification or anomaly detection.

Overall, it can be concluded that PCA is an effective tool for reducing the dimensionality of forensic data from the supertimeline, whilst allowing a substantial portion of the information to be retained whilst simplifying subsequent analytical tasks.

5.3 Combining PCA data

Due to limited computational capacity, it was not possible to process all 769 attributes at once. Therefore, the approach was based on applying the Principal Component Analysis (PCA) method separately for each *source type*.

However, the aim was not to work with multiple separate tables, but to create a unified representation of the data. After selecting the relevant components using PCA, pre-processing was therefore applied to the individual datasets, resulting in the extraction of 178 selected attributes for each dataset.

These attributes were subsequently merged into a single shared table. To preserve information about the origin of individual records, auxiliary attributes of the type *`mask_sourcetype`* were also added, which explicitly indicate the record's affiliation with a specific data source.

Table 8 provides an overview of the number of selected attributes for individual artefacts (sourcetypes) following the application of PCA, whilst the table also illustrates the degree of dimensionality reduction within individual data sources.

Artifact	Original attributes	Number of selected attributes
WinEVTX	27	23
File_entry_shell_item	23	9
NTFS_file_stat	27	15
File_stat	22	16
Windows_Shortcut	25	10
Amcache_Registry_Entry	2	1
AppCompatCache_Registry_Key	2	1
BagMRU_Registry_Key	20	6
Chrome_Cache	17	10
Chrome_Cookies	22	11
Chrome_History	21	9
MRUList_Registry_Key	23	9

MRUListEx_Registry_Key	12	5
MSIE_WebCache_container_record	22	13
MSIE_WebCache_containers_record	23	9
MSIE_Cache_File_URL_record	2	1
Network_Connection_Registry_Key	2	1
OLECF_Dest_list_entry	15	8
OLECF_Item	17	3
OLECF_Summary_Info	2	1
Open_XML_Metadata	2	1
PE_Compilation_time	2	1
PE_COFF_file	2	1
Registry_Key	16	14
Registry_Key___BagMRU	12	4
Registry_Key___UserAssist	18	5
Registry_Key___Run_Key	2	1
Registry_Key___MRUList	2	1
Registry_Key___MRUListEx	2	1
Registry_Key___Typed_URLs	2	1
Run_Run_Once_Registry_Key	22	10
System	2	1
Service_Driver_Configuration_Registry_Key	2	1
Setup_API_Log	2	1
Shutdown_Registry_Key	8	1
Task_Cache_Registry_Key	20	7
USB_Registry_Key	16	2
User_Account_Information_Registry_Key	2	1
WinPrefetch	16	12
Winlogon_Registry_Key	2	1

Table5 – File-based merging (image)

Table 9 contains the final list of all attributes after data merging, with the resulting matrix representation consisting of 178 attributes. This table provides a complete overview of the attributes used in further processing and represents a unified representation of data across all datasets.

Column	Artifact	Attribute description
access_count_gt_1	WEBHIST - MSIE WebCache Container record	The object has been visited or used more than once.

accessed_appdata	REG - MRUList Registry Key	Access to files in AppData (often of interest from a malware perspective).
accessed_local_file	WEBHIST - MSIE WebCache Container record	The record represents access to a local file via a file:// path.
accessed_network_path	REG - MRUList Registry Key	Access to a file via a network path (UNC path).
accessed_sensitive_docs	REG - MRUList Registry Key	Indicates access to potentially sensitive documents (Office files, PDFs or the occurrence of the word 'password').
accessed_sensitive_file	WEBHIST - MSIE WebCache Container record	The record contains access to a potentially sensitive file or content.
account_creation	WinEVTX	Explicitly identifies the creation of a new user account.
account_deletion	WinEVTX	Explicitly identifies the deletion of a user account.
container_id	WEBHIST - MSIE WC Containers record	Unique identifier of the WebCache container.
contains_appdata_path	REG - Run_Run Once Registry Key	The executable file is located in AppData.
contains_cab	FILE - File Entry Shell Item	CAB archive (packages, updates).
contains_cur	FILE - File Entry Shell Item	Cursor file (.cur), rather rare.
contains_inf	FILE - File Entry Shell Item	INF file (installation scripts, often for drivers).
contains_maintenance_task	REG - Task Cache Registry Key	The task is related to system maintenance.
contains_path_downloads	REG - MRUListEx Registry Key	The entry contains the path to the Downloads folder.
contains_path_recent	REG - MRUListEx Registry Key	The file comes from the Recent folder (recently opened items).
contains_ppd	FILE - File Entry Shell Item	Printer Description file.
contains_secret_path	WEBHIST - MSIE WebCache Container record	Indicates a local file:// path containing the word secret.
contains_sensitive_string	REG - MRUListEx Registry Key	The record contains sensitive keywords (e.g. password, credentials, secret).

contains_update_task	REG - Task Cache Registry Key	The task relates to updates.
contains_vbs	FILE - File Entry Shell Item	VBScript file (often exploited by malware).
ContainsCmdCommand	REG - Registry Key	The registry contains a CMD command.
ContainsPowershellCommand	REG - Registry Key	The registry contains a PowerShell command.
ContainsScriptExtension	REG - Registry Key	The registry refers to a script (BAT, PS1, VBS, JS).
event_powershell	WinEVTX	PowerShell activity.
event_registry	WinEVTX	Operations on the Windows Registry.
has_exe_target	LNK - Windows Shortcut	The shortcut points to an .exe executable file.
has_hostname_reference	OLECF - Dest List Entry	The entry contains a reference to a hostname (may indicate a network resource).
has_ip_url	WEBHIST - MSIE WebCache Container record	The URL is written directly using the IP address instead of a domain name.
has_macb_accessed	FILE - File Entry Shell Item	The file has been opened (Accessed).
has_macb_modified	FILE - File Entry Shell Item	The file has been modified.
has_modified_flag	FILE - File Entry Shell Item	The file has been modified (modified timestamp).
	FILE - File Stat	
has_modified_time	LNK - Windows Shortcut	The first character of MACB is M, meaning the record indicates a modification in the modified timestamp.
has_persistence_trigger	REG - Task Cache Registry Key	The task has a periodic or conditional trigger (e.g. daily, on inactivity).
has_ps1_target	LNK - Windows Shortcut	The shortcut points to a PowerShell .ps1 script.
has_sha256	FILE - File Stat	The file has an available SHA256 hash (important for identification and threat intelligence).
has_transition_type	WEBHIST - Chrome History	The record contains information about the transition type (e.g. click, redirect, typed URL).

has_unknown_shell_item	REG - Registry Key - BagMRU	Indicates that the record contains an unknown or unrecognised shell item type.
HasAandBNoMC	LNK - Windows Shortcut	The shortcut has both an access and a creation timestamp, but no modification or change timestamp.
HasCommandArguments	REG - Run_Run Once Registry Key	The executable file contains arguments (e.g. program.exe -arg).
is_accessed	FILE - File Stat	The file has been opened (accessed).
	WEBHIST - Chrome Cookies	The cookie was used (accessed), i.e. the page actively utilised it.
is_ad_service	WEBHIST - Chrome Cache	The source is associated with advertising services (Google Ads, DoubleClick).
is_admin_activity	WinEVTX	Activity associated with administrator accounts.
is_advertising_cookie	WEBHIST - Chrome Cookies	Cookie used for advertising purposes (ad targeting).
is_allocated	FILE - File Entry Shell Item	The file is allocated (exists in the file system).
is_analytics_cookie	WEBHIST - Chrome Cookies	Cookie used for analytics (e.g. Google Analytics).
is_apis_google	WEBHIST - Chrome Cache	The source uses Google API endpoints.
is_application_experience	WinEVTX	Application Experience events.
is_bing_domain	WEBHIST - Chrome Cookies	The cookie originates from the Bing domain.
is_browser_exe	LOG - WinPrefetch	The entry belongs to a web browser.
is_cdn_usage	WEBHIST - Chrome Cache	The resource is loaded via a Content Delivery Network (e.g. Akamai, CDN servers).
is_cmd	REG - Registry Key - UserAssist	Command Prompt launched.
is_cmd_or_script	FILE - File Entry Shell Item	Indicates the launch of a command prompt or scripts (CMD, BAT, PowerShell).
is_content_cache	WEBHIST - MSIE WC Containers record	The container contains cached content (INetCache – stored web resources).

is_content_container	WEBHIST - MSIE WC Containers record	Content container (web content – HTML, images, scripts).
is_desktop_or_downloads	REG - Registry Key - BagMRU	The record refers to the Desktop or Downloads folder.
is_error_event	WinEVTX	Error or failed events.
is_event_log_related	WinEVTX	Events related to the Windows Event Log service.
is_evtx_started_state	WinEVTX	The service has entered the running state.
is_evtx_stopped_state	WinEVTX	The service has been stopped.
is_executable	FILE - File Stat	Executable file or system driver.
	REG - MRUList Registry Key	Indicates that an executable file or script has been opened or run.
	OLECF - Dest List Entry	The entry represents an executable file or script.
is_failed_login	WinEVTX	Indicates failed login attempts (e.g. Event ID 4625 or failed logon).
is_file	FILE - File Stat	The record represents a file.
is_file_entry	FILE - File Stat	Alternative file entry detection (exact log format).
is_filename_attr	FILE - File Entry Shell Item	NTFS attribute $FILE_NAME
is_fileshare_access	WEBHIST - MSIE WebCache Container record	Indicates access to a shared file or network path; also captures text files.
is_frequent_execution	LOG - WinPrefetch	The application has been launched very frequently, at least 50 times.
is_from_driverstore	FILE - File Entry Shell Item	The file is located in the DriverStore (system drivers).
is_from_ie	WEBHIST - Chrome History	The entry was imported from Internet Explorer.
is_from_pinned_location	FILE - File Entry Shell Item	The file was launched from a pinned location (Taskbar, Quick Launch).
is_from_suspicious_domain	WEBHIST - Chrome Cookies	The cookie originates from a potentially risky or spam domain (keyword-based heuristics).
is_from_system32	FILE - File Entry Shell Item	The file originates from the System32 system folder.

is_from_user_profile	OLECF - Dest List Entry	The file is located in the user profile (AppData, Roaming, Users).
is_from_users_dir	FILE - File Entry Shell Item	The file is in the user directory.
is_google_domain	WEBHIST - Chrome Cache	The source originates from the google.com domain.
	WEBHIST - Chrome Cookies	The cookie originates from the Google domain.
is_gstatic_domain	WEBHIST - Chrome Cache	The source comes from the gstatic.com domain (static files for Google services).
is_high_access_count	WEBHIST - MSIE WebCache Container record	The object has a high number of accesses, more than 10.
is_history_container	WEBHIST - MSIE WC Containers record	Browsing history container (visited pages).
is_https	WEBHIST - Chrome History	The URL uses the secure HTTPS protocol.
is_iecompat_container	WEBHIST - MSIE WC Containers record	Container related to Internet Explorer compatibility.
is_iecompatua_container	WEBHIST - MSIE WC Containers record	Container for user-agent compatibility (IE modes).
is_image_file	WEBHIST - Chrome Cache	The cache contains an image file.
is_in_appdata	FILE - File Stat	The file is located in the AppData folder (a common target for malware).
is_in_system32	FILE - File Stat	The file is in the System32 system folder.
is_in_temp_or_spool	FILE - File Stat	The file is in a temporary or spool folder (often short-lived or suspicious files).
is_in_winsxs	FILE - File Stat	The file is in WinSxS (Windows Side-by-Side, legitimate system libraries).
is_language_code	OLECF - OLECF Item	The item has a name represented by a 4-digit code, which often corresponds to language or identification codes in the OLECF structure.
is_large_file	FILE - File Stat	File larger than ~10 MB.
is_link_file	FILE - File Entry Shell Item	Indicates a Windows shortcut (.lnk), often used to track programme launches.

is_log_recovered	WinEVTX	The log has been restored following an error or crash.
is_login_event	WinEVTX	Identifies successful user logons (e.g. Event ID 4624 or logon).
is_modified_after_access	WEBHIST - MSIE WebCache Container record	The MACB record begins with M, indicating modification after or during access.
is_multiple_volumes	LOG - WinPrefetch	The operation is associated with multiple volumes, which may indicate access to external or multiple storage devices.
is_network_event	THIS ATTRIBUTE IS NOT PARSED	
is_network_path	FILE - File Entry Shell Item	The file originates from a network path (e.g. UNC path).
	REG - MRUListEx Registry Key	The entry represents access to a file on a network path (UNC path).
	REG - Run_Run Once Registry Key	Launch from a network or remote path.
is_network_related	WinEVTX	Network connectivity and events.
	REG - Registry Key - UserAssist	Indicates that the running programme is related to network communication.
is_network_share	REG - BagMRU Registry Key	Indicates access to a network share via a UNC path, e.g. \\server\share.
is_night_access	WEBHIST - Chrome History	The activity took place at night (22:00 – 06:00).
is_night_execution	LOG - WinPrefetch	Indicates events related to changes in system security policies (e.g. Event ID 4719 or the occurrence of the word 'policy').
is_ntfs	FILE - File Stat	The file is on an NTFS file system.
is_ntfs_event	WinEVTX	NTFS file system events.
is_obfuscated_path	LOG - WinPrefetch	The path contains obfuscation characters, e.g. hexadecimal directory names or suspicious sequences ...

is_pinned	OLECF - Dest List Entry	The file is pinned in the Jump List.
is_policy_change	WinEVTX	Indicates events related to changes in system security policies (e.g. Event ID 4719 or the occurrence of the word 'policy').
is_powershell	REG - Registry Key - UserAssist	PowerShell launch.
is_privilege_change	WinEVTX	Indicates events where privileged rights have been changed or used (e.g. Event ID 4672, 4673 or privilege occurrence).
is_rare_app	LOG - WinPrefetch	The application has been run only rarely, fewer than 5 times.
is_repeated_record	WEBHIST - MSIE WC Containers record	Identifies containers that occur multiple times (multiple records).
is_root_entry	OLECF - OLECF Item	Identifies the root entry of the OLECF file.
is_run_often	LOG - WinPrefetch	The application has been run repeatedly, more than 5 times.
is_runonce_key	REG - Run_Run Once Registry Key	RunOnce key – executed only once at startup.
is_script_file	FILE - File Stat	Script (.bat or PowerShell .ps1).
	WEBHIST - Chrome Cache	JavaScript file.
is_secure	WEBHIST - Chrome Cookies	The cookie has the Secure attribute set (transmitted only via HTTPS).
is_sensitive_file	OLECF - Dest List Entry	The record contains an indication of a sensitive file (e.g. “SECRET”, “password”).
is_service_event	WinEVTX	Events related to system service management (Service Control Manager).
is_social_media	WEBHIST - Chrome History	The URL belongs to social media.
is_specific_container	WEBHIST - MSIE WebCache Container record	The record belongs to a specific WebCache container, e.g. History or Cookies.

is_standard_info	FILE - File Entry Shell Item	NTFS attribute $STANDARD_INFORMATION.
is_suspicious_app	LOG - WinPrefetch	Indicates the launch of a tool frequently exploited in attacks or post-exploitation.
is_suspicious_domain	WEBHIST - Chrome Cache	The domain is not among the known/whitelisted ones (Google, YouTube, Facebook, Twitter), and is therefore potentially less trustworthy.
	WEBHIST - MSIE WebCache Container record	The URL points to a domain ending in .ru or .cn, which is heuristically flagged as potentially suspicious.
is_suspicious_extension	REG - Run_Run Once Registry Key	The file being executed has a suspicious extension (scripts or executable).
is_suspicious_keyword	WEBHIST - MSIE WebCache Container record	The record contains suspicious or sensitive keywords.
is_suspicious_task_name	REG - Task Cache Registry Key	The task name matches a generic or obfuscated name.
is_suspicious_tool	REG - Registry Key - UserAssist	The running program is one of the well-known tools used in attacks (LOLbins).
is_system_hive	REG - Run_Run Once Registry Key	The key belongs to the system registry hive.
is_system_location	REG - BagMRU Registry Key	Indicates that the entry points to a system or special shell location, e.g. Control Panel, Recycle Bin or Network.
is_system_path	LNK - Windows Shortcut	The shortcut is located in a system or shared location, e.g. Windows, System32 or ProgramData.
is_system_process	LOG - WinPrefetch	Indicates that the prefetched program is one of the standard Windows system processes.
is_system_util	LOG - WinPrefetch	The entry belongs to a system utility tool, which may be legitimate or malicious.
is_temp_launch	LOG - WinPrefetch	The program was launched from the TEMP temporary folder.

is_terminal_services	WinEVTX	Remote access (RDP, Terminal Services).
is_third_party	WEBHIST - Chrome Cookies	The cookie originates from a third-party domain (not directly from the visited website).
is_tracking_cookie	WEBHIST - Chrome Cookies	"General indicator for tracking cookies (a combination of analytics and advertising)."
is_txt_file	OLECF - Dest List Entry	The file is a text file (.txt).
is_unallocated	FILE - File Entry Shell Item	"The file has been deleted (unallocated space)."
is_unpinned	OLECF - Dest List Entry	The file is not pinned (standard recent item).
is_unusual_location	FILE - File Entry Shell Item	Identifies files in unusual (often misused) locations such as AppData or Temp, outside standard system folders.
is_url_visit	WEBHIST - MSIE WebCache Container record	Indicates that the record contains a visit to a web URL via HTTP or HTTPS.
is_usb_hub	REG - USB Registry Key	Indicates that the device is a USB hub.
is_user_account_event	WinEVTX	Indicates events relating to user accounts (e.g. account creation – Event ID 4720).
is_user_location	REG - MRUList Registry Key	The file is located in user folders (Documents, Desktop, Downloads, AppData).
is_user_profile_path	REG - Registry Key - BagMRU	The entry contains the path to the user profile in the format C:\Users\<username>.
is_user_profile_shortcut	LNK - Windows Shortcut	The shortcut is located in the user profile, e.g. in Users, AppData or Roaming.
is_xml_file	FILE - File Stat	XML file (often configurations).
is_youtube_domain	WEBHIST - Chrome Cookies	The cookie originates from the YouTube domain.
IsAdministratorProfile	WEBHIST - MSIE WC Containers record	The container belongs to the administrator account.
IsCSS	WEBHIST - Chrome Cache	CSS file (website styles).
IsDynamicLibrary	FILE - File Stat	Dynamic library (.dll).

IsInAppData	FILE - File Entry Shell Item	The file is located in the AppData folder.
IsInRoamingAppData	FILE - File Entry Shell Item	"The file is located in local AppData (not roaming)."
IsSuspiciousPath	REG - Registry Key	The path contains suspicious elements (temp, random names, GUID).
IsSystemHive	REG - Registry Key	The key belongs to the system hive (HKLM).
IsUserHive	REG - Registry Key	The key belongs to the user hive (HKCU).
key_autorun_related	REG - Registry Key	Indicates autorun/autostart mechanisms.
key_contains_run	REG - Registry Key	The registry key contains Run/RunOnce – a classic persistence mechanism.
key_contains_winlogon	REG - Registry Key	The key is associated with Winlogon – a very strong persistence point.
key_new_registry_branch	REG - Registry Key	Indicates the creation of or access to a new registry branch.
key_suspicious_value	REG - Registry Key	The registry contains suspicious executable tools.
macb_anomaly	LNK - Windows Shortcut	A specific non-standard MACB pattern .A.B, which may indicate anomalous timing behaviour of the shortcut.
macb_is_only_access_birth	LNK - Windows Shortcut	The MACB contains only access and/or birth flags without any modification or change to the metadata.
MissingMInMACB	LNK - Windows Shortcut	The shortcut does not have a modification flag (M) in the MACB record.
ModifiesImagePath	REG - Registry Key	Indicates a change to the path of an executable file (often a service or driver).
ModifiesStartType	REG - Registry Key	Change to the service start-up type (e.g. auto-start).
multiple_entries	REG - MRUList Registry Key	The entry contains multiple MRU items (multiple recently used objects).

opened_control_panel	REG - BagMRU Registry Key	Indicates that the user has opened or viewed the Control Panel.
opened_recycle_bin	REG - BagMRU Registry Key	Indicates that the user has opened or viewed the Recycle Bin.
OpenedExcelFile	REG - MRUList Registry Key	Indicates that an Excel file (.xls, .xlsx) has been opened.
OpenedWordDocument	REG - MRUList Registry Key	Indicates the opening of a Word document (.doc, .docx).
potential_malware	WinEVTX	Potentially malicious behaviour.
RunsRundll32	REG - Run_Run Once Registry Key	rundll32.exe is running – a frequently abused tool for launching DLLs.
suspicious_executable	REG - Run_Run Once Registry Key	The name of the executable file mimics legitimate system processes.
task_starts_at_boot	REG - Task Cache Registry Key	The task runs at system startup.
task_starts_at_logon	REG - Task Cache Registry Key	The task runs when the user logs on.
unknown_shell_item_type	REG - BagMRU Registry Key	The entry contains an unknown or unidentified shell item type.
UsesQuotesInPath	REG - Run_Run Once Registry Key	The file path is enclosed in quotation marks.
visit_to_search_engine	WEBHIST - Chrome History	Visit to a search results page (Google or Bing).
visited_microsoft	WEBHIST - Chrome History	Visit to a Microsoft domain.
visited_msn	WEBHIST - Chrome History	Visit to the MSN domain.

Table 9 - Description of selected PCA attributes

6 Summary

Following the data merging process, the resulting data representation contains **178 attributes**, augmented by **40 masking columns** representing individual source types. These columns enable the source context of each record to be explicitly captured and support its further interpretation.

In view of the next processing step, namely aggregation, it was necessary to analyse the content of the newly created datasets in detail. For this purpose, a summary table was created, which contains the number of occurrences of the values 0, 1 and NaN for each attribute and each dataset (CTF). This table (summary_table) forms part of the appendix to this output – „D15 – Model for digital evidence extraction to matrix representation– appendix“ and is located in the „summary_table“ workbook.

The table includes all processed datasets and has been created in a standardised manner for each of them, enabling them to be compared with one another. It thus provides a comprehensive overview of the distribution of individual attributes, their presence (value 1), absence (value 0) and missing values across all datasets. At the same time, it serves as a basis for further processing, particularly when designing aggregation approaches and working with incomplete data.

The table also includes a textual description of individual attributes and their interpretation from the perspective of digital forensic analysis, including an assessment of their relevance to the investigation. It is important to emphasise, however, that a low individual relevance of an attribute does not mean it is insignificant. In combination with other attributes, it can provide significant contextual information and contribute to the identification of more complex patterns of behaviour.

The ‘summary_table’ thus serves not only as a quantitative overview of the data, but also as an interpretative basis for further processing, particularly when designing aggregation approaches and working with incomplete data.

The occurrence of NaN values is a consequence of previous data processing – specifically situations where a given attribute belongs to a certain source type that is not, however, represented in a particular row. In other words, these missing values do not indicate an absence of information, but rather the irrelevance of the attribute for the given record. These are therefore structurally missing values that arise naturally when combining heterogeneous data sources into a unified representation.

This type of NaN value must be interpreted differently from classic missing data caused, for example, by the unavailability or loss of information. In our case, NaN explicitly signals that the attribute in question is not defined for the specific artefact and should therefore not be treated as a zero value in further processing (e.g. aggregation or modelling).

The correct interpretation of these values is crucial, particularly when designing aggregation functions, where it is necessary to decide whether NaN should be ignored or whether it should have a specific meaning within the calculation. This aspect directly influences the resulting

data representation and can have a significant impact on the subsequent analysis and interpretation of results.

7 Bibliography

[1] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, and L. Chang, “A novel anomaly detection scheme based on principal component classifier,” in Proc. IEEE Foundations and New Directions of Data Mining Workshop, 2003.

[2] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho, “A survey of network-based intrusion detection data sets,” Computers & Security, vol. 86, pp. 147–167, 2019.

[3] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, “Anomaly-based intrusion detection system through feature selection analysis and building a hybrid efficient model,” Journal of Computational Science, vol. 25, pp. 152–160, 2018.

[4] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” ACM Computing Surveys, vol. 41, no. 3, pp. 1–58, 2009.

[5] S. Wold, K. Esbensen, and P. Geladi, “Principal component analysis,” Chemometrics and Intelligent Laboratory Systems, vol. 2, no. 1–3, pp. 37–52, 1987.

[6] I. T. Jolliffe and J. Cadima, “Principal component analysis: A review and recent developments,” Philosophical Transactions of the Royal Society A, vol. 374, no. 2065, 2016.

[7] Plaso (log2timeline), 2022, [online] Available: https://github.com/log2timeline/plaso.

8 Appendices

Table ADFIR-D15-KPB3-01-table serves as an appendix to the data analysis and contains two separate worksheets providing an overview of the structure of the source data and the properties of the extracted attributes:

- **sourcetype_source_counts_summar** – a summary table recording the number of occurrences of individual source types (sourcetype) across the analysed datasets (CTF), which serves to identify dominant, marginal and less suitable data sources for further processing.
- **summary_table** – a summary table of binary attributes following pre-processing, which contains the counts of values 0, 1 and NaN for each attribute and dataset, including a description of the attributes and their interpretation from the perspective of digital forensic analysis.