Formal concept analysis and attribute dependencies of NIST data leakage case*

Ľubomír Antoni[1,*], Pavol Sokol[1,*], Sophia Petra Krišáková[1,*], Dominika Kotlárová[1], Ondrej Krídlo[1] and Stanislav Krajči[1]

[1]Institute of Computer Science, Faculty of Science, Pavol Jozef Šafárik University in Košice, Jesenná 5, 040 01 Košice, Slovakia

Abstract

Security incidents involving unauthorized access and data leakage remain one of the most critical challenges in modern cybersecurity, demanding transparent and interpretable methods for behavioral analysis. This paper presents an application of Formal concept analysis and association rule mining to the NIST Data Leakage Case EVT dataset, which simulates a security incident involving potential data exfiltration. Our objective is to uncover and interpret logical dependencies among binary audit attributes and time-based features extracted from system event logs. Using Formal concept analysis, we construct a concept lattice that visualizes the hierarchical structure of attribute co-occurrences and reveals frequent as well as rare behavioral patterns. From the underlying formal context, we derive a canonical base of attribute implications with 100% confidence and extend this with approximate association rules to capture near-deterministic relationships. The results highlight both typical access configurations and anomalous combinations involving weekend and night-time activity. Our findings demonstrate that Formal concept analysis provides a transparent and systematic framework for reasoning over event data, offering interpretable support for anomaly detection and forensic analysis in cybersecurity contexts.

Keywords

Formal concept analysis, cybersecurity, event analysis, concept lattice, attribute implication

1. Introduction

Insider threats, particularly those involving intentional data exfiltration by trusted employees, represent a persistent and complex challenge for cybersecurity systems. Traditional signature-based detection methods often fail to identify subtle behavioral patterns that precede such incidents. To address this gap, high-quality, realistic datasets are crucial for developing and evaluating behavioral and anomaly-based detection models. Behavioral modeling approaches, such as those based on access logs, user activity profiling, and context-aware policies have been proposed as more effective strategies for identifying insider misuse [1, 2, 3]. Based on this dataset, a specific dataset [4] suitable for various machine learning and data analysis methods was created. This dataset is also applicable in the area of formal concept analysis. A more detailed description of the dataset is provided in Section 3.

However, progress in this area strongly depends on the availability of comprehensive and semantically rich datasets that simulate real-world security contexts. The NIST Data Leakage Test Case is a publicly available dataset designed by the National Institute of Standards and Technology to support research in the detection of insider threats and unauthorized data exfiltration. It simulates user behavior within a fictional organization, incorporating realistic event logs, file access traces, and security-relevant metadata that culminate in a staged data leakage scenario. The dataset was created to evaluate behavioral detection methods, policy enforcement mechanisms, and forensic analysis tools in cybersecurity [5].

Formal concept analysis is a set of mathematical methods grounded in lattice theory and propositional logic, designed to identify and analyze relationships within structured data. As an unsupervised bi-

ITAT'25: Information technologies – Applications and Theory, September 26–30, 2025, Telgárt, Slovakia

*Corresponding author.

lubomir.antoni@upjs.sk (Ľ. Antoni); pavol.sokol@upjs.sk (P. Sokol); sophia.petra.krisakova@student.upjs.sk (S. P. Krišáková); dominika.kotlarova@student.upjs.sk (D. Kotlárová); ondrej.kridlo@upjs.sk (O. Krídlo); stanislav.krajci@upjs.sk (S. Krajči)

0000-0002-7526-8146 (Ľ. Antoni); 0000-0002-1967-8802 (P. Sokol); 0009-0007-9325-071X (D. Kotlárová); 0000-0001-8166-6901 (O. Krídlo); 0000-0001-5612-3534 (S. Krajči)

clustering approach, Formal concept analysis simultaneously groups both objects and their attributes, forming formal concepts, i.e., pairs consisting of a set of objects (extent) and a set of shared attributes (intent). These concepts are organized into a hierarchical structure called a concept lattice, which reveals inherent patterns, dependencies, and generalization-specialization relationships within the dataset [6, 7].

Formal concept analysis provides a robust framework for exploratory data analysis, knowledge discovery, and fuzzy extensions, particularly in domains where the interpretation of attribute co-occurrence is essential. By uncovering hidden regularities and highlighting meaningful associations, Formal concept analysis supports informed decision-making and deepens the understanding of complex relational structures [8, 9, 10, 11].

The remainder of this paper is organized as follows. After the introduction, we present the related research and papers in Section 2. Section 3 introduces the dataset and outlines the preprocessing steps applied. In Section 4, we describe the proposed methodology, which integrates Formal concept analysis and the extraction of attribute dependencies. Section 5 presents the experimental results along with their interpretation and discussion.

2. Related works

In this paper, we focus on the use of Formal concept analysis for solving cybersecurity incidents and conducting digital forensic analysis. In this sense, the paper complements the papers [12, 13]. In the paper [12], the authors focus on analysing meaningful groups of digital objects based on common attributes and visualising the hierarchy of concepts. They describe the formal context derived from digital traces collected from NTFS file system and present several concept lattices enriched with association rules. Their research is further expanded in the paper [13], where the authors generate four concept lattices for different subsets of attributes (timestamps, file types, etc.) and compare several association rule mining methods, while also interpreting fuzzy implications in the context of Formal Concept Analysis. Digital forensic analysis of cybersecurity incidents involving social engineering is also addressed in the paper [14].

Cybersecurity represents a broad area of different topics, as reflected in the related work. Formal Concept Analysis can serve as a handy tool for detecting and retrieving various criminal activities committed in cyberspace. In the paper [15], the authors develop an explicitly defined and conceptual system for analysing e-fraud data in cyberspace.

The use of Formal concept analysis for malware analysis is also of interest. In the paper [16], the authors propose F-FCA (Feature-driven Formal concept analysis), in which each object and concept is associated with a temporal logic formula. They also introduce the FOCA algorithm to generate the concept hierarchy using an object-joining operator. Experiments on a real dataset of 3,000 malware samples demonstrate the effectiveness of the proposed approach compared to traditional Formal concept analysis. Malware analysis using Formal concept analysis is also discussed in the paper [17], where the authors argue that creating a standard naming convention and hierarchy for malware is important to improve collaboration and information sharing in this field.

Security governance is an important part of cybersecurity, and examples of the use of FCA can also be found in this area. In the paper [18], the authors present a systematic synthesis of the General Data Protection Regulation (GDPR) using Formal concept analysis. Based on its principles, the GDPR is synthesised into a concept lattice containing 144.372 records. This can be used, for example, to identify implicit logical relations within the regulation and their intensity. These results can support (re)design, development, operation, or refactoring of information systems towards a higher level of GDPR compliance. Threat and threat-agent analysis is also part of cybersecurity management. In the paper [19], Formal concept analysis was used to uncover deeper relationships in the MITRE taxonomic framework. The results of the exploratory analysis were then encoded into an ontology using the OWL language, allowing logical reasoning over the relationships between cyber techniques and procedures.

Finally, asset protection and the implementation of security measures are also part of cybersecurity.

In this respect, the paper [20] proposes a new heuristic approach based on Formal concept analysis for improving the security and privacy protection of sensitive e-Health information through item-hiding techniques. The proposed FACHS method minimises side effects and distortion of the original database while not requiring preliminary frequent itemset mining.

The above shows that Formal concept analysis represents one of the possible approaches to solving various cybersecurity problems. The research presented in this paper builds on the work reported in the papers [12, 13]. In contrast to those papers, the present work focuses on a different type of forensic artefacts. In addition, the presented research also focuses on temporal data, as also explored in the paper [16].

3. Dataset and its preprocessing

The NIST Data Leakage Case [5] offers a unique resource for researchers and practitioners in this area. Published by the National Institute of Standards and Technology (NIST) as part of its Data Exfiltration Test Cases, the dataset simulates user behavior in a fictional organization where a data leakage incident gradually unfolds. It includes detailed Windows event logs, file system activity, access records, and metadata annotations. The NIST Data Leakage Case EVT dataset [4] created from captures both normal and malicious activity, making it suitable for supervised and unsupervised machine learning, as well as exploratory data analysis.

This dataset enables the study of temporal patterns, user behavior profiling, privilege escalation, and unauthorized access attempts. Its structure supports tasks such as clustering, anomaly detection, and association rule mining, particularly using its rich set of binary audit attributes and event metadata.

To investigate behavioral patterns and detect potential anomalies in system activity, we selected a set of binary attributes from the NIST Data Leakage Case EVT dataset. From the full event log, we extracted nine binary features representing audit categories and log metadata (Table 1). These were complemented by two newly engineered time-based features.

Table 1
Selected binary attributes from the EVT dataset

Attribute	Description
`audit_account_management`	Indicates events related to account management operations, such as user creation, deletion, or password changes.
`audit_logon_logoff`	Captures user logon and logoff events, both interactive and remote.
`audit_policy_change`	Marks changes in system security policies, e.g., audit settings or access controls.
`audit_system`	Signals general system-level events such as startup, shutdown, or driver loading.
`keywords_audit_failure`	Indicates that the logged event corresponds to a failed audit condition (e.g., failed login).
`keywords_audit_success`	Represents events that were successfully audited, such as successful authentication.
`keywords_correlation_hint`	Used for correlating related events across subsystems or processes.
`keywords_event_log_classic`	Identifies traditional (legacy format) Windows event log entries.
`keywords_sqm`	Refers to Software Quality Metrics events related to system telemetry.

In addition to these system-generated binary indicators, two custom binary time-based attributes were created:

- `is_night_activity`: equals 1 if the event occurred between 00:00 and 05:59 UTC (nighttime hours), and 0 otherwise;
- `is_weekend`: equals 1 if the event occurred on a Saturday or Sunday, and 0 otherwise.

The final dataset used for analysis consists of 10,306 rows and 11 binary columns, forming a structured event-log matrix suitable for unsupervised learning techniques such as bi-clustering and the extraction of attribute implications, which we describe in the following section. For a more detailed description of the dataset creation process and attribute specification, see the paper [4].

4. Methodology

In this section, we introduce the basic notions of Formal concept analysis, a mathematical framework for extracting and representing conceptual structures in data. Formal concept analysis is appropriate for analyzing binary relations between objects and their attributes, which aligns with the structure of our dataset. The fundamental building block of Formal concept analysis is the formal context, which encodes the presence or absence of relationships between elements [6, 7]. We begin with the following formal definition:

Definition 1. *Let B and A be non-empty sets representing, respectively, a collection of* objects *and* attributes*, and let $I \subseteq B \times A$ be a crisp binary relation indicating which objects are associated with which attributes. The triple $\langle B, A, I \rangle$ is termed a* formal context*. The relation I is referred to as the* incidence relation*, capturing the presence of attributes in objects.*

	i	ii	iii
a	×		
b	×	×	
c		×	

Figure 1: Illustrative example of a formal context comprising three objects and three attributes.

We may interpret a formal context as a binary incidence table, where each entry indicates whether a given object possesses a particular attribute. To formally capture the structure embedded in such a context, we now introduce two dual operators that act on subsets of objects and attributes. These operators serve as the foundation for constructing formal concepts.

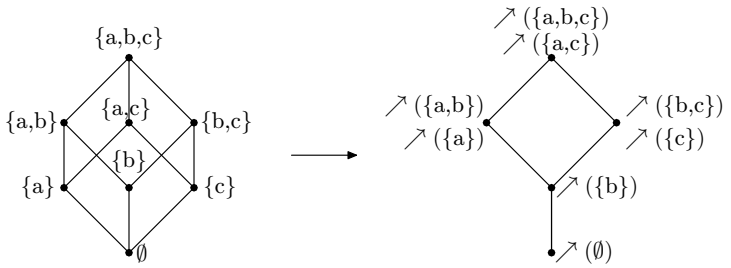

Figure 2: Illustrative example of the concept-forming operators derived from the formal context

Definition 2. *Let $\langle B, A, I \rangle$ be a formal context, and let $X \in \mathcal{P}(B), Y \in \mathcal{P}(A)$ denote subsets of the sets of objects and attributes, respectively. We define two mappings:*

$$\nearrow : \mathcal{P}(B) \to \mathcal{P}(A), \quad X^{\nearrow} = \{y \in A \mid \forall x \in X,\ \langle x, y \rangle \in I\},$$

$$\swarrow : \mathcal{P}(A) \to \mathcal{P}(B), \quad Y^{\swarrow} = \{x \in B \mid \forall y \in Y,\ \langle x, y \rangle \in I\}.$$

These mappings are referred to as the concept-forming operators *of the formal context $\langle B, A, I \rangle$. The operator $\nearrow$ yields the set of all attributes common to a given set of objects, while $\swarrow$ returns the set of all objects that share a given set of attributes.*

The concept-forming operators introduced above provide the foundation for defining structured clusters of data known as *formal concepts*.

Definition 3. *Let $\langle B, A, I\rangle$ be a formal context and let $\nearrow, \swarrow$ be the associated concept-forming operators. For any subsets $X \in \mathcal{P}(B)$ and $Y \in \mathcal{P}(A)$, a pair $\langle X, Y\rangle$ is called a* formal concept *of the context $\langle B, A, I\rangle$ if and only if it satisfies the conditions:*

$$X^{\nearrow} = Y \quad \textit{and} \quad Y^{\swarrow} = X.$$

In this case, the set X is referred to as the extent *of the formal concept, representing the collection of objects, while the set Y is called the* intent*, representing the set of attributes shared by all objects in X.*

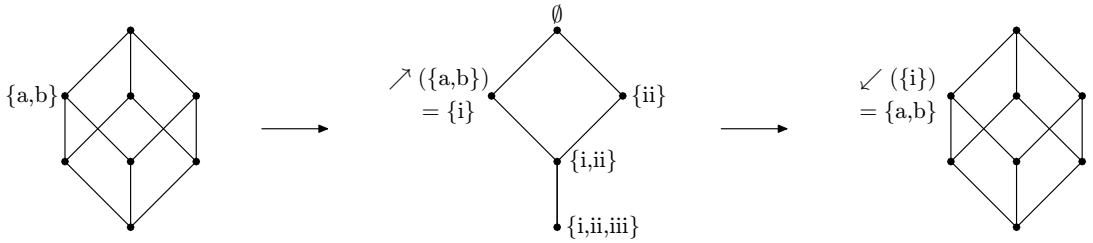

Figure 3: Illustrative example of a formal concept consisting of two objects and one shared attribute

The collection of all formal concepts derived from a given formal context $\langle B, A, I\rangle$ is denoted as:

$$\mathrm{C}(B, A, I) = \{\langle X, Y\rangle \in \mathcal{P}(B) \times \mathcal{P}(A) \mid X^{\nearrow} = Y,\ Y^{\swarrow} = X\}.$$

The set of all formal concepts derived from a given context can be naturally equipped with a partial order based on the inclusion of extents (or, equivalently, reverse inclusion of intents). Under this order, the set of formal concepts forms a complete lattice structure, known as the *concept lattice*.

Definition 4. *Let $\langle X_1, Y_1\rangle, \langle X_2, Y_2\rangle \in \mathrm{C}(B, A, I)$ be two formal concepts of the context $\langle B, A, I\rangle$. Define a partial order $\preceq$ on $\mathrm{C}(B, A, I)$ by:*

$$\langle X_1, Y_1\rangle \preceq \langle X_2, Y_2\rangle \quad \textit{if and only if} \quad X_1 \subseteq X_2 \quad (\textit{equivalently, } Y_2 \subseteq Y_1).$$

The partially ordered set $\langle \mathrm{C}(B, A, I), \preceq\rangle$ is called the concept lattice *of the context $\langle B, A, I\rangle$. For convenience, it is typically denoted by $\mathrm{CL}(B, A, I)$.*

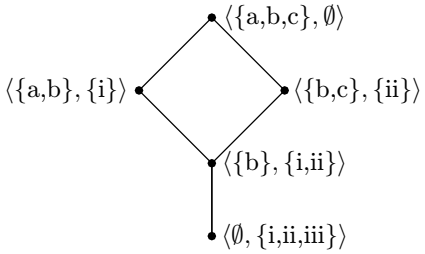

Figure 4: Illustrative example of a concept lattice comprising five formal concepts

The formal context and its associated concept lattice provide a foundation for discovering *attribute dependencies*, also known as *attribute implications*, that hold within a given dataset [6, 7]. These dependencies capture regularities in how attributes co-occur across the object set. As demonstrated by Ganter and Wille [6], Formal concept analysis can, for instance, be applied to the analysis of feasible configurations of computer hardware components, where only certain combinations of features are

considered valid. This exemplifies how one may study admissible attribute combinations and the logical relationships between them.

Such an approach is particularly beneficial in classification tasks where a large set of objects is characterized by a relatively small number of attributes. In these cases, it becomes advantageous to represent domain knowledge through *attribute implications*, i.e., logical formulas stating that the presence of one subset of attributes guarantees the presence of another.

Formally, we define attribute implications as follows:

Definition 5. *Let A be a non-empty set of attributes and let $Z, W \subseteq A$. An* attribute implication *over A is a formal expression of the form $Z \Rightarrow W$, interpreted as: "every object possessing all attributes in Z also possesses all attributes in W." The implication $Z \Rightarrow W$ is said to be* valid *in a set $Y \subseteq A$ if $Z \subseteq Y$ implies $W \subseteq Y$. Equivalently, the implication is valid in Y if either $Z \not\subseteq Y$ or $W \subseteq Y$.*

We now provide a simple example to illustrate the validity of an attribute implication:

Example 1. *Let $A = \{a_1, a_2, a_3, a_4\}$ be a set of attributes, and let $B = \{a_1, a_4\} \subseteq A$. The implication $\{a_1, a_3\} \Rightarrow \{a_4\}$ is a valid attribute implication over A. Moreover, it is* valid *in the subset B, since $\{a_1, a_3\} \not\subseteq B$ and the premise of the implication does not hold in B.*

We now extend the notion of implication validity from individual subsets of attributes to collections of such subsets. This generalization enables us to formalize the notion of an attribute implication being valid across a set of observations or contexts.

Definition 6. *Let A be a non-empty set of attributes, and let $Z, W \subseteq A$. Furthermore, let $\mathcal{Y} \subseteq \mathcal{P}(A)$ denote a collection of subsets of attributes. An attribute implication $Z \Rightarrow W$ is said to be* valid *in $\mathcal{Y}$ if it is valid in every set $Y \in \mathcal{Y}$, i.e., for all $Y \in \mathcal{Y}$, the condition $Z \subseteq Y \Rightarrow W \subseteq Y$ holds.*

This generalized notion of validity allows us to define the validity of attribute implications within a formal context by interpreting the context as a collection of attribute sets derived from objects.

Definition 7. *Let $\langle B, A, I \rangle$ be a formal context, and let $Z, W \subseteq A$. An attribute implication $Z \Rightarrow W$ is said to be* valid in the formal context *$\langle B, A, I \rangle$ if it is valid in the collection*

$$\mathcal{Y} = \{\{x\}^{\nearrow} \mid x \in B\},$$

where $\nearrow$ denotes the concept-forming operator that maps an object to the set of attributes it possesses.

The set $\mathcal{Y} = \{\{x\}^{\nearrow} \mid x \in B\}$ thus represents all attribute sets associated with individual objects in the context. This construction provides a basis for evaluating the validity of any attribute implication with respect to the entire formal context.

While this approach allows for checking the validity of selected attribute implications, our objective is often to compute a *complete* and *non-redundant* set of all valid implications that fully characterizes the data. For this purpose, the *Guigues–Duquenne basis* (also known as the *stem base*) offers a canonical and minimal representation of all valid attribute implications in a formal context [21]. Efficient algorithms for its computation are described in [6, 22], and it plays a central role in the logical analysis of data dependencies.

While attribute implications capture only those relationships that are logically valid in the context (i.e., hold in 100% of the objects), a more flexible framework is needed to describe *approximate* dependencies that may hold with high but not perfect reliability. This is where the notion of *confidence* becomes central [23].

Definition 8. *Let $\langle B, A, I\rangle$ be a formal context, and let $Z, W \subseteq A$. The* confidence *of an implication $Z \Rightarrow W$ in this context is defined as:*

$$conf(Z \Rightarrow W) = \frac{|\{x \in B \mid Z \subseteq x^{\nearrow} \text{ and } W \subseteq x^{\nearrow}\}|}{|\{x \in B \mid Z \subseteq x^{\nearrow}\}|}.$$

That is, confidence is the proportion of objects possessing all attributes in Z that also possess all attributes in W.

An attribute implication is thus a *special case* of an *association rule* with confidence equal to 100%. More generally, association rules allow for modeling relationships that are statistically strong but not universally valid. For example, the rule

$$Z \Rightarrow W \quad \text{with conf}(Z \Rightarrow W) = 97\%$$

expresses that in 97% of all objects where Z holds, the attributes in W also hold. This framework is particularly useful in data analysis tasks involving noise, exceptions, or incomplete patterns.

In our analysis, we first computed all attribute implications with confidence 100% using the canonical Guigues–Duquenne basis. We then extended the analysis by relaxing the confidence threshold, thereby deriving a broader set of association rules that capture near-deterministic behavioral patterns in the data.

5. Results and interpretation

To analyze the structure of dependencies and co-occurrences among binary attributes in the dataset, we applied *Formal concept analysis* described in the previous section. This technique allowed us to extract a hierarchy of formal concepts based on shared attribute subsets, resulting in a **concept lattice** consisting of 26 distinct concepts.

Figure 5 shows the resulting concept lattice diagram. Each node in the diagram represents a *formal concept*, characterized by an extent (the number of objects/rows) and an intent (the set of common attributes). The extent size and its proportion with respect to the full dataset (10,306 rows) are shown in each node, e.g., "4862 / 47%" indicates that 47% of the objects share the attributes represented by that node.

The lattice is ordered by set inclusion: higher nodes have more general attribute sets (smaller intent, larger extent), while lower nodes represent more specific combinations (larger intent, smaller extent). Edges between concepts indicate subset relationships among attribute sets.

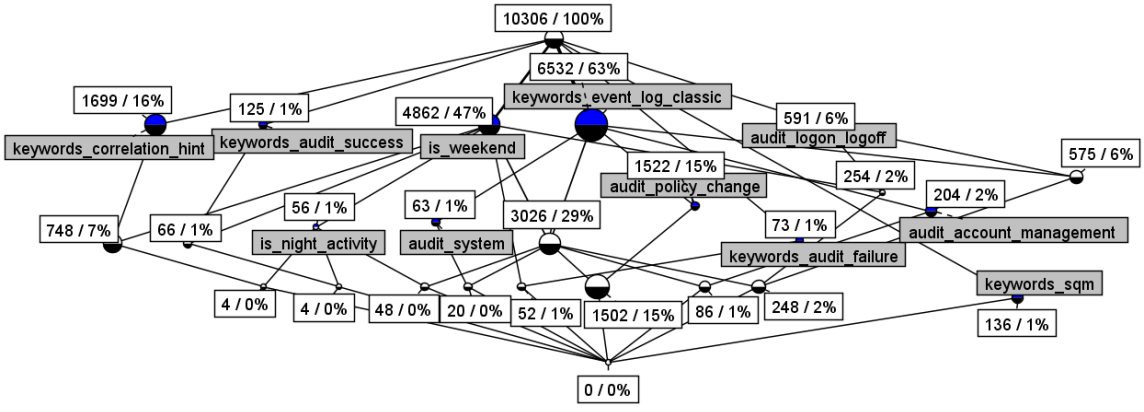

Figure 5: Concept lattice of binary attributes from the EVT dataset. Nodes are labeled with extent size and percentage.

Several important observations emerge from the structure of the concept lattice. These insights reflect not only the frequency and co-occurrence of attributes, but also reveal potentially anomalous or rare behavioral patterns in the dataset:

- The most frequent attribute is `keywords_event_log_classic`, present in 63% of the events, followed by `is_weekend` (47%) and `keywords_correlation_hint` (16%).
- The time-based attribute `is_night_activity` appears in only 1% of the events, reflecting the fact that the dataset contains only a small fraction of activity recorded during nighttime hours. This low support limits its general influence but makes it a useful indicator for identifying potential anomalies occurring outside regular working hours.
- Concepts near the bottom of the lattice (with very small support, e.g., $< 1\%$) can represent highly specific and rare patterns. These may be of interest in anomaly detection or in identifying suspicious behavioral combinations.
- No objects were found to match the bottom-most node (extent = 0), meaning that no event shares the full set of binary attributes simultaneously, which is expected.

In summary, the concept lattice provides a hierarchical view of how event attributes co-occur, revealing frequent combinations and rare intersections. This structure can guide both rule extraction (via implications) and further filtering of unusual behavior for investigation, which is described in the following paragraphs.

In addition to the concept lattice analysis, we applied attribute implication mining and association rule extraction[1] on the same binary dataset consisting of 10,306 rows and 11 attributes. The goal was to discover logical dependencies between attribute combinations, expressed in the form of implications $A \Rightarrow B$ with associated support and confidence values. The resulting rules provide interpretable insights into frequent co-occurrences, typical system behaviors, and potential temporal or structural patterns in the event data. A selection of the most relevant rules is summarized in Table 2.

Table 2
Selected attribute implications and association rules extracted using Formal Concept Analysis

No.	Premise	Conclusion	Confidence
1	`audit_policy_change`	`keywords_event_log_classic`	100%
2	`audit_policy_change`, `keywords_event_log_classic`	`is_weekend`	99%
3	`audit_logon_logoff`	`keywords_event_log_classic`	97%
4	`audit_logon_logoff`, `is_weekend`	`keywords_event_log_classic`	98%
5	`audit_account_management`	`keywords_event_log_classic`	100%
6	`audit_system`	`keywords_event_log_classic`	100%
7	`is_night_activity`	`is_weekend`	100%
8	`is_night_activity`, `is_weekend`	`keywords_event_log_classic`	86%

The rules in Table 2 reveal strong co-occurrence patterns among audit and keyword attributes. For instance, rules 1, 3, 5, and 6 show that whenever certain system-level activities occur (such as `audit_policy_change`, `audit_logon_logoff`, or `audit_account_management`), the events are consistently logged using the legacy Windows event log format (`keywords_event_log_classic`). This highlights the dominant presence of classical logging mechanisms in administrative operations.

Rule 2 suggests that nearly all policy change events (in combination with legacy logging) occur during the weekend, which may indicate scheduled maintenance or non-standard administrative behavior.

Rule 7 implies that all events recorded during the night (`is_night_activity`) also occurred on weekends (`is_weekend`). This narrow temporal window may reflect an intentional restriction or filtering in the data source which we mentioned in the previous concept lattice, as well.

Rule 8 shows that such weekend-night events are typically logged using the classic format, but with slightly reduced confidence (86%), which might hint at occasional outliers or logging exceptions.

[1]The concept lattice, attribute implications and association rules were generated using the *Concept Explorer* tool, a software environment for Formal concept analysis, available at http://conexp.sourceforge.net.

Together, these rules help identify deterministic and near-deterministic patterns in event attributes, allowing for the creation of a behavioral baseline. Deviations from these rules could serve as indicators of anomalies or unusual system activity.

6. Conclusion

In this study, we explored the application of Formal concept analysis to a security dataset, the NIST Data Leakage Case EVT. Using a carefully selected set of binary audit attributes and engineered time-based features, we constructed a formal context with 10,306 objects and 11 attributes. This enabled us to apply Formal concept analysis methods for uncovering logical dependencies and co-occurrence patterns within system event logs.

We first analyzed the structure of the dataset using a concept lattice, which revealed 26 formal concepts organized by attribute inclusion. This hierarchical representation provided insights into the frequency and overlap of various event types, highlighting typical combinations as well as rare and potentially anomalous ones. In particular, we identified rare conjunctions involving weekend and night-time activity that may warrant further security inspection.

Building on the concept lattice, we extracted both exact attribute implications (100% confidence) and approximate association rules (with confidence less than 100%). These rules capture stable behavioral patterns within the dataset and serve as a form of interpretable knowledge discovery. For instance, we found that all system-level and account management events consistently appear with legacy event log markers, and that night-time activity is limited and correlated with weekend occurrences.

In this study, we focused on the classical framework of Formal concept analysis applied to binary (single-valued) formal contexts, where the incidence relation strictly defines whether an object possesses a given attribute. While this approach provides a solid foundation for structural analysis and dependency mining, it assumes crisp object-attribute relationships. As a direction for future work, the extension of Formal concept analysis to many-valued or fuzzy settings presents a promising research avenue. Future work could benefit from integrating fuzzy and many-valued extensions of Formal Concept Analysis, which enable reasoning over graded or uncertain information. Notably, theoretical foundations developed by Bělohlávek [24], Butka et al. [25], and Medina et al. [26] offer promising directions for adapting our framework to more complex, real-world data scenarios.

7. Acknowledgments

This research was carried out within the project "Automatization of Digital Forensics and Incident Response (ADFIR)" (project code 09-I05-03-V02-00079), funded under the The Recovery and Resilience Plan of the Slovak Republic K9 scheme: "Effective management and support of funding for science, research and innovation" approved by the Council of the European Union. The project, implemented at Pavol Jozef Šafárik University in Košice in collaboration with IstroSec s.r.o. and the European Information Society Institute, aims to develop an automated framework for the collection, normalization, and evaluation of digital traces, while ensuring their integrity and legal admissibility, to empower cybersecurity teams in responding to incidents and reducing the impact of cyber-attacks.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] W. Eberle, L. Holder, Insider threat detection using graph-based approaches, in: 2009 Cybersecurity Applications & Technology Conference for Homeland Security (CATCH), IEEE, Washington, DC, USA, 2009, pp. 237–241. doi:10.1109/CATCH.2009.7.

[2] M. Bishop, C. Gates, Defining the insider threat, in: Proceedings of the 4th Annual Workshop on Cyber Security and Information Intelligence Research (CSIIRW), CSIIRW '08, ACM, Oak Ridge, TN, USA, 2008, pp. 15:1–15:3. doi:10.1145/1413140.1413158.

[3] M. N. Al-Mhiqani, T. Alsboui, T. Al-Shehari, K. H. Abdulkareem, R. Ahmad, M. A. Mohammed, Insider threat detection in cyber-physical systems: a systematic literature review, Computers and Electrical Engineering 119 (2024) 109489. doi:10.1016/j.compeleceng.2024.109489.

[4] E. Marková, P. Sokol, S. P. Krišáková, K. Kováčová, Dataset of windows operating system forensics artefacts, Data in Brief 55 (2024) 110693.

[5] National institute of standards and technology, data leakage test case, https://cfreds-archive.nist.gov/data_leakage_case/data-leakage-case.html, 2025. CFReDS forensic reference dataset.

[6] B. Ganter, R. Wille, Formal Concept Analysis: Mathematical Foundations, Springer, Berlin, Heidelberg, 1999.

[7] C. Carpineto, G. Romano, Concept Data Analysis: Theory and Applications, John Wiley & Sons, 2004.

[8] M. E. Cornejo, J. Medina, F. J. Ocaña, Attribute implications in multi-adjoint concept lattices with hedges, Fuzzy Sets and Systems 479 (2024) 108854. doi:10.1016/j.fss.2023.108854.

[9] P. Cordero, M. Enciso, Ángel Mora, F. Pérez-Gámez, Attribute implications with unknown information based on weak heyting algebras, Fuzzy Sets and Systems 490 (2024) 109026. doi:10.1016/j.fss.2024.109026.

[10] R. Bělohlávek, M. Trnečka, Semantic explorations in factorizing boolean data via formal concepts, International Journal of Approximate Reasoning 173 (2024) 109247. doi:10.1016/j.ijar.2024.109247.

[11] M. Ojeda-Hernández, D. López-Rodríguez, Ángel Mora, A formal concept analysis approach to hierarchical description of malware threats, Forensic Science International: Digital Investigation 50 (2024) 301797. doi:10.1016/j.fsidi.2024.301797.

[12] P. Sokol, L. Antoni, O. Krídlo, E. Marková, K. Kováčová, S. Krajči, The analysis of digital evidence by formal concept analysis, in: Proceedings of the International Conference on Concept Lattices and Their Applications (CLA), 2022, pp. 147–158.

[13] P. Sokol, L. Antoni, O. Krídlo, E. Marková, K. Kováčová, S. Krajči, Formal concept analysis approach to understand digital evidence relationships, International Journal of Approximate Reasoning 159 (2023) 108940.

[14] I. B. Senkyire, Q.-A. Kester, Social engineering cybercrime evidence analysis using formal concept analysis, in: 2021 International Conference on Cyber Security and Internet of Things (ICSIoT), IEEE, 2021, pp. 26–29.

[15] V. Waziri, A. Umar, M. Olalere, E-fraud forensics investigation techniques with formal concept analysis, International Journal of Cyber-Security and Digital Forensics 3 (2014) 235–245.

[16] N. T. Binh, T. C. Doi, Q. T. Tho, N. M. Hai, Feature-driven formal concept analysis for malware hierarchy construction, in: International Workshop on Multi-disciplinary Trends in Artificial Intelligence, Springer, 2015, pp. 385–396.

[17] M. Ojeda-Hernández, D. López-Rodríguez, A. Mora, A formal concept analysis approach to hierarchical description of malware threats, Forensic Science International: Digital Investigation 50 (2024) 301797.

[18] D. A. Tamburri, Design principles for the general data protection regulation (gdpr): A formal concept analysis and its evaluation, Information Systems 91 (2020) 101469.

[19] L. Maluleke, A formal concept analysis driven ontology for ics cyberthreats, in: Proceedings of the South African Conference for Artificial Intelligence Research (SACAIR), 2020, pp. 247–257.

[20] H. Hamdi, Z. Brahmi, A. S. Alaerjan, L. Mhamdi, Enhancing security and privacy preservation of

sensitive information in e-health datasets using fca approach, IEEE Access 11 (2023) 62591–62604.
[21] J.-L. Guigues, V. Duquenne, Familles minimales d'implications informatives résultant d'un tableau de données binaires, Mathématiques et Sciences humaines 95 (1986) 5–18.
[22] G. Stumme, Conceptual Knowledge Discovery with Frequent Concept Lattices, Technical Report FB4-Preprint 2043, Fachbereich Informatik, Technische Universität Darmstadt, Darmstadt, Germany, 1999. Technical Report.
[23] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD '93, ACM, Washington, D.C., USA, 1993, pp. 207–216. doi:10.1145/170035.170072.
[24] R. Bělohlávek, Lattices of fixed points of fuzzy galois connections, Mathematical Logic Quarterly 47 (2001) 111–116.
[25] P. Butka, J. Pócs, J. Pócsová, Distributed computation of generalized one-sided concept lattices on sparse data tables, Computing and Informatics 34 (2015) 77–98.
[26] J. Medina-Moreno, M. Ojeda-Aciego, J. Pócs, E. Ramírez-Poussa, On the dedekind-macneille completion and formal concept analysis based on multilattices, Fuzzy Sets and Systems 303 (2016) 1–20. doi:10.1016/j.fss.2016.01.007.