Ontological-Relational Data Store Model for a Cloud-based SIEM System Development

Viktoriya Sydorenko[1], Oksana Zhyharevych[2], Rat Berdybaev[3], Artem Polozhentsev[1], and Andriy Fesenko[1]

[1] *National Aviation University, 1 Liubomyr Huzar Ave, Kyiv, 03058, Ukraine*
[2] *Lesya Ukrainka Volyn National University, 13 Volya Ave., Lutsk, 43025, Ukraine*
[3] *Almaty University of Energy and Communications, 126/1 Baytursynuli Str, Almaty, 050013, Kazakhstan*

Abstract

Security Information and Event Management (SIEM) systems are widely used to prevent information loss in computer systems and networks. Currently, there are different approaches to building databases (data stores) for SIEM systems. Analysis has not revealed a universal type of database, and each has its advantages and disadvantages. This paper presents the rationale for selecting the most effective databases, based on which the model of an ontological-relational data store is implemented. The proposed model uses two different types of databases with appropriate characteristics, to improve the convenience of data storage and classification, to ensure high speed of obtaining large amounts of information through preliminary indexing, as well as load balancing and data replication. These results will be useful both for critical information infrastructure protection and for building various cyber threat monitoring systems.

Keywords

Database, database management system, SIEM, ontological-relational data store, SQL, NoSQL, NewSQL, load balancing, data replication.

1. Introduction

As the number of cyber threats continues to grow, one of the most effective ways to detect them and protect information is to deploy SIEM systems. The use of SIEM in computer security incident response centers (CSIRTs) is key to ensuring effective detection, analysis, and response to security incidents. Here are some of the main ways to use SIEM in a CSIRT:

- *Centralized log collection:* an SIEM allows you to centrally collect, store, and manage logs from various sources across an organization's network, including servers, network equipment, applications, and other security systems. This gives CSIRTs the ability to quickly access important data for incident analysis.
- *Event correlation:* SIEM can automatically correlate collected logs and identify potential security incidents using various rules, attack signatures, and behavioral analysis algorithms. This helps the CSIRT team detect sophisticated attacks that may go undetected without such analysis.
- *Real-time monitoring:* SIEM provides real-time security monitoring, allowing CSIRT teams to respond to incidents quickly. Using interactive tools to visualize and analyze data can help identify unusual or suspicious activity.
- *Decision support:* The analytics and reporting tools that come with an SIEM allow CSIRT teams to analyze security trends and identify potential vulnerabilities or security gaps. This helps to better plan security strategies and make informed decisions to strengthen protection.
- *Documentation and regulatory compliance:* SIEM helps in automating the collection and

CPITS-2024: Cybersecurity Providing in Information and Telecommunication Systems, February 28, 2024, Kyiv, Ukraine
EMAIL: v.sydorenko@ukr.net (V. Sydorenko); zhyharevych.oksana@vnu.edu.ua (O. Zhyharevych); r.berdybaev@aues.kz (R. Berdybaev); artem.polozhencev@gmail.com (A. Polozhentsev); aafesenko88@gmail.com (A. Fesenko)
ORCID: 0000-0002-5910-0837 (V. Sydorenko); 0000-0002-7154-9733 (O. Zhyharevych); 0000-0002-8341-9645 (R. Berdybaev); 0000-0003-0139-0752 (A. Polozhentsev); 0000-0001-5154-5324 (A. Fesenko)

storage of logs for documentation and reporting as per regulatory and compliance requirements. This is important for meeting legal requirements and can serve as evidence in incident investigations.

- *Professional development and training:* With the use of SIEM, CSIRT teams can conduct training and simulations of real incidents, analyzing the collected data and practicing response procedures. This helps increase the team's preparedness for real-world threats and improves their ability to respond quickly and effectively to incidents.
- *Automation of response:* Some SIEM systems provide the ability to automate routine incident response tasks, such as isolating an infected device, blocking IP addresses, or sending notifications to the appropriate individuals. This reduces response time and frees up resources to focus on more complex tasks.
- *Integration with other security systems:* Integrating SIEMs with other security tools and systems, such as intrusion detection and prevention systems (IDS/IPS), antiviruses, and identity and access management systems, can provide deeper security analysis and help detect sophisticated attacks.
- *Continuous improvement:* Analyzing incidents and responses to them helps to identify weaknesses in security systems and response processes, allowing for the necessary adjustments to be made to improve performance. This is a continuous improvement process that helps to strengthen protection and reduce the risk of future incidents.
- *Collaboration and information sharing:* SIEM can facilitate collaboration between CSIRT team members and other stakeholders by providing shared access to incident information, analytics, and reports. Also, sharing threat information with other organizations and communities can help develop better defense strategies.
- *Compliance with legal and regulatory requirements:* Using an SIEM helps companies meet legal and regulatory requirements by providing the necessary reporting, auditing, and monitoring of security standards.
- *Incident prediction and prevention:* Analyzing the data collected and processed by an SIEM can help predict potential threats and develop strategies to prevent them, reducing the likelihood of future incidents [1–5].

SIEM operation is based on the use of databases (DBs), i.e. structured data stored digitally in a computer system. The DB is managed by a database management system (DBMS). The data, together with the DBMS and associated applications, form a DB system. Modern types of DBs usually store data in the form of tables, where information is presented in the form of rows and columns. This information can be easily managed, added, edited, deleted, updated, monitored, etc. Most modern DBs use a structured query language (SQL) to enter records and retrieve information.

2. Literature Review and Problem Statement

There are many modern DB types [6–8]. The choice of DB type for a particular SIEM system is determined by the specifics of how the data will be used in a particular context. Templates and structures used to organize information in DBMS are referred to as DB types [6–10].

Studies [11–13] have analyzed modern DB types used in SIEM systems to identify their strengths and weaknesses and have systematized them as follows:

2.1. The simplest DB types

Let's start by looking at three types of DBs that are still found in specialized environments but have been largely replaced by reliable and efficient alternatives.

2.1.1. Simple data structures. The first and simplest way of storing data is in text files. This method is still used today for working with small amounts of information. A special character is used to separate fields: a comma or semicolon in CSV files, and a colon or space in *nix-like systems.

2.1.2. Hierarchical DBs. Unlike text tables, the next type of DBe has relationships between objects. In hierarchical DBs, each record has an ancestor. This creates a tree structure in which records are classified according to their relationship to a lower level of the record chain, the structure of hierarchical DBs.

2.1.3. Network DBs. Network DBs extend the functionality of hierarchical DBs by allowing records to have more than one ancestor. This

means that you can model complex relationships and the structure of network DBs.

2.2. Relational DBs

2.2.1. SQL. Relational DBs are the oldest and still widely used general-purpose DBs. Data in relational DBs is structured in the form of tables, which are made up of columns and rows. Each column in the table has its own name and data type. Each row represents a separate record or item of information in the table, containing the value for each of the columns.

2.2.2. OLTP. OLTP is designed to perform business transactions performed by multiple users, the structure of the OLTP database.

Relational DBs are used by the following SIEM systems IBM QRadar, AlienVault USM, LOGRHYTHM, AlienVault OSSIM, Splunk, FortiSIEM, Wazuh, SolarWinds, ManageEngine, RuSIEM, Prelude OSS, Prelude SIEM, Sagan, Maxpatrol, EventTracker, Trustwave SIEM Enterprise, McAfee (ESM) [11–16].

2.3. NoSQL DBs

NoSQL is a group of DB types that offer approaches other than the standard relational model. NoSQL stands for "non-SQL" or "not only SQL" to indicate that SQL-like queries are sometimes allowed. A NoSQL non-relational database allows you to store and process unstructured or semi-structured data (unlike a relational DB, which defines the structure of the data it contains). The popularity of NoSQL is growing as web applications proliferate and become more complex.

2.3.1. Key-value DBs. To store information in a key-value DB, you need to specify a key and a data object to store. For example, a JSON object, an image, or text. To retrieve data, the key is sent and a blob, a NoSQL structure, is received.

2.3.2. Document-oriented DBs. Document-oriented DBs (document DBs or document repositories) share the basic semantics of accessing and searching key and value stores. Such DBs also use a key to uniquely identify data. The difference between key-value stores and document DBs is that document DBs store data in structured formats: JSON, BSON, or XML, the structure of a document DB – rather than in blocks [17, 18].

2.3.3 Graph DBs. Instead of representing relationships using tables and foreign keys, graph DBs establish relationships using nodes, edges, and properties. Graph DBs represent data in terms of individual nodes, which can have any number of properties associated with them. Graph DBs store data in the context of entities and relationships between entities.

2.3.4. Columnar DBs. Columnar DBs, also known as non-relational columnar storage or wide column DBs, belong to the category of NoSQL systems but look like relational DBs. Similar to relational DBs, columnar DBs store data in the form of rows and columns, but have a different structure of relationships between elements. In relational databases, all rows must follow a fixed schema. The schema determines which columns will be in the table, their data types, and other characteristics. Columnar DBs, on the other hand, have structures called "column families" instead of tables. Column families contain rows, each of which can have its format. Each row consists of a unique identifier used for searching, followed by a set of column names and values.

2.3.5. Time series DBs. Such DBs are designed to collect and manage items that change over time. Most such DBs are organized into structures that record values for an element. For example, you might create a table to track the temperature of a processor. Inside, the values consist of a time stamp and a temperature value.

NoSQL is used by the following SIEMs: AlienVault USM, AlienVault OSSIM, MozDef, Maxpatrol, and SearchInform SIEM.

2.4. Combined DBs

NewSQL and multi-model DBs are different types of databases, but they aim to solve a common set of problems that arise from using opposing SQL or NoSQL strategies.

***2.4.1. NewSQL* DBs**. NewSQL inherits the relational structure and semantics but is built using more modern, scalable designs. The goal is to provide greater scalability than relational DBs and higher consistency guarantees than NoSQL. The trade-off between consistency and availability is a fundamental problem in distributed DBs, described by the CAP theorem [19, 20].

2.4.2. Multi-model DBs. Such DBs combine the functionality of several types of DBs. The advantages of this approach are the following. The same system can use different representations for different types of data. Combining data from different types of DBs in

one system allows new operations that would otherwise be difficult or impossible [21].

2.5. Object-oriented DBs

Information in an object-oriented database (OODB) is represented as an object, as in object-oriented programming.

2.6. Cloud DBs

A cloud DB is a set of structured or unstructured data hosted on a private, public, or hybrid cloud computing platform [11–15, 22–24]. There are two types of cloud DB models: traditional DB and DB as a service (DBaaS). In the DBaaS model, administrative tasks and maintenance are performed by a cloud provider, the structure of cloud DBs [25, 26].

Cloud DB types use the following SIEM: HPE ArcSight Splunk Ixia ThreatARMOR, Micro Focus ArcSight, and Trustwave SIEM Enterprise.

The results of the analysis of DBMSs used in different SIEM systems according to [11–13] are shown in Table 1.

Table 1: SIEM and corresponding DBMS platforms

SIEM	DBMS
IBM QRadar	Ariel database, PostgreSQL, SQLite
LogRhythm	Oracle, SQL Server, MySQL
Splunk	DB2 / Linux, Informix, MemSQL, MySQL, AWS Aurora, Microsoft SQL Server, Oracle, PostgreSQL, AWS RedShift, SAP SQL Anywhere, Sybase ASE, Sybase IQ, and Teradata
McAfee (ESM)	MSSQL, Oracle, MySQL, Data Access Server (DAS), DB2 / UDB
AlienVault USM	RedisDB, MySQL
AlienVault OSSIM	RedisDB, MySQL
FortiSIEM	PostgreSQL
Ixia ThreatARMOR	Rap Sheet
MozDef	RabbitMQ, MongoDB, Elasticsearch, Kibana
Wazuh	MySQL, PostgreSQL
Prelude OSS	MySQL, PostgreSQL
Prelude SIEM	MySQL, PostgreSQL
Sagan	MySQL, PostgreSQL
Maxpatrol	ElasticSearch, MongoDB, MS SQL Express
SolarWinds	MSSQL, Oracle, MySQL, MariaDB.
ManageEngine	Oracle, SQL, DB2, MySQL
EventTracker	Microsoft SQL Server
Micro Focus ArcSight	Own development CORR-E
Trustwave SIEM Enterprise	Microsoft SQL Server, Microsoft SQL Azure, ORACLE, SYBASE, MySQL, IBM, DB2, Hadoop
BlackStratus SIEMStorm	Own development

According to the analysis in Table 1, each specific type of DBMS remains relevant in its own area, where the relationships between data are determined by the specific structure of the DB. Attention should be paid to the possibility of using hybrid DBs that combine different types of DBMS, such as SQL and NoSQL, which will allow maintaining convenience in storing and classifying data, as well as ensuring high speed in obtaining large amounts of information due to preliminary indexing. In addition, the need to develop a new data store model has been substantiated, which, on the one hand, should store, process and search events in logs at the highest possible speed, and, on the other hand, should store service data about users, metadata, configuration settings, hashed thread counters and an alert archive in a reliable and structured manner.

The aim of the study is therefore to develop a model of an ontological-relational data store for use in a cloud-based SIEM system. To achieve the mentioned aim the correlation-regression multifactor analysis will be used.

3. Model Development: DBs Correlation Analysis and Defining

Ontological-Relational Data Store (Fig. 1) is a data management system that combines the concepts of ontologies and relational DBs to store and manage information. This paradigm combines two main ideas [27]:

Ontology: An ontology is a formally defined knowledge model or semantic framework used to describe objects, concepts, and their interactions in a particular domain. Ontologies define the semantics of data and help to understand how data is related to each other.

Relational data: Relational Databases (RDBMS) are powerful systems for storing data in the form of tables with relational relationships between them. They use the SQL query language to access data.

In an ontological-relational data store, information is stored in the form of tables (as in relational DB) but is additionally accompanied by an ontology that provides a semantic context for the data. The ontology helps to understand the meaning of data, its relationships, and context in a broader sense.

This makes it easier to search, filter, and understand data. Ontology-relational data stores are used in various fields, including the semantic web, biology, medicine, geography, and other areas where it is important to understand the semantic context of data, as well as to efficiently access large amounts of data.

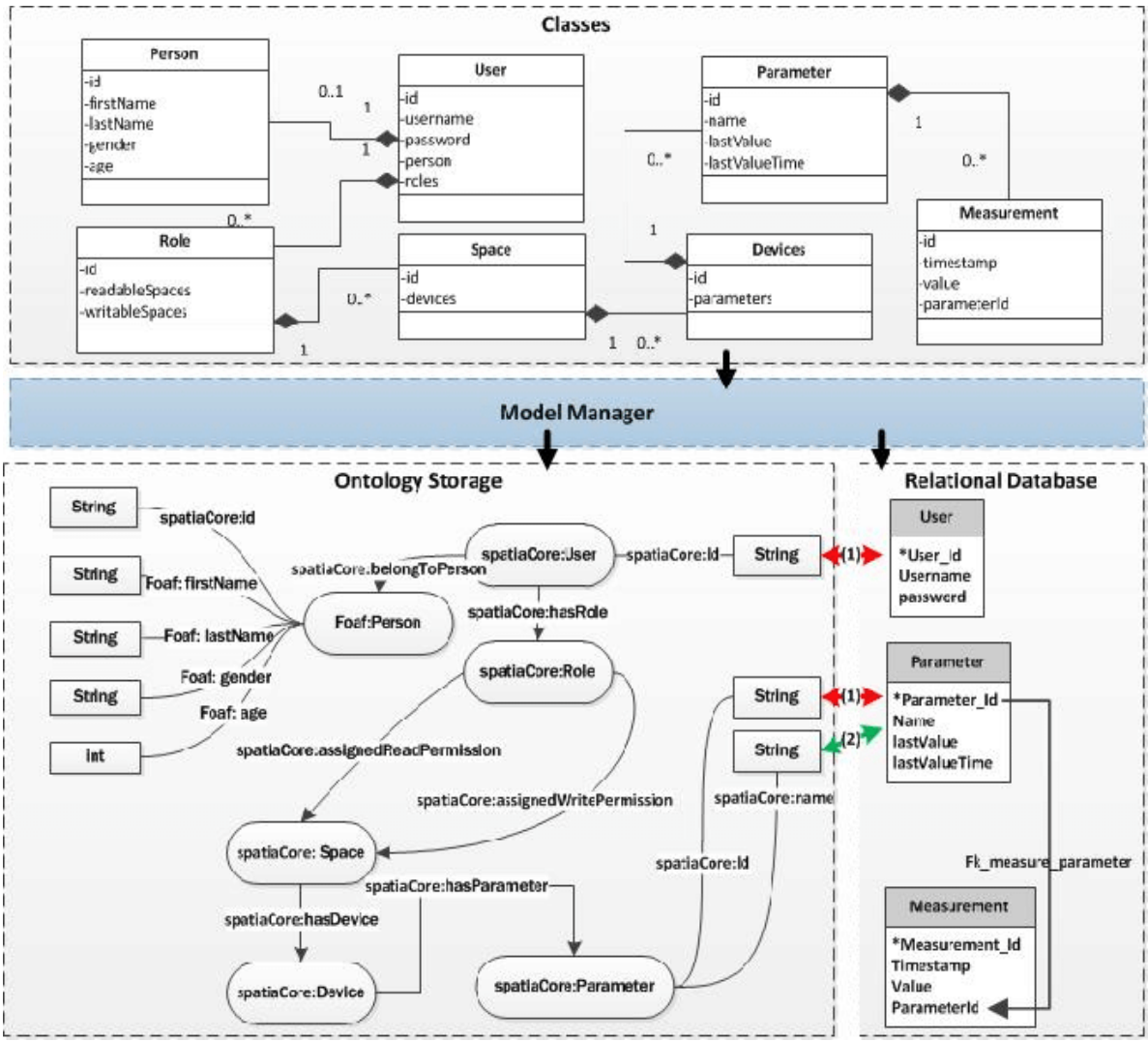

Figure 1: Scheme of ontological-relational paradigm

To support the selection of the most effective DBs used in modern SIEM systems, let's use the procedure of correlation and regression multifactor analysis, which includes the following stages:

Stage 1: Analysis of existing factors and justification of the type of regression model. A general list of factors is made and their possible numerical characteristics for quantitative and qualitative representation are determined. An analytical expression is constructed to reflect the relationship between the factor and the resulting characteristics of the function:

$$\hat{Y} = f(x_1, x_2, x_3, ..., x_d), \qquad (1)$$

where $\hat{Y}$ is an effective feature function; $x_1, x_2, x_3, ..., x_d$ are factor attributes of **DB**.

In addition, the multiple regression equation can be represented in a linear form:

$$\hat{Y} = a_0 + a_1 x_1 + a_2 x_2 + ... + a_d x_d, \qquad (2)$$

where $a_0, a_1, ..., a_d$ are the parameters of the equation to be determined.

If d values of $y_h, x_{1h}, x_{2h}, \ldots, x_{dh}$, are known for each **DB** factor and the resulting trait, $h = 1, 2, \ldots, m$, then using the standard least squares method a system of linear algebraic equations is obtained to estimate the parameters of the regression equation:

$$\begin{cases} a_0 m + a_1 \sum_{j=1}^{m} x_{1j} + a_2 \sum_{j=1}^{m} x_{2j} + ... + a_d \sum_{j=1}^{m} x_{dj} = \sum_{j=1}^{m} y_j; \\ a_0 \sum_{j=1}^{m} x_{1j} + a_1 \sum_{j=1}^{m} x_{1j}^2 + a_2 \sum_{j=1}^{m} x_{1j} x_{2j} + ... + a_d \sum_{j=1}^{m} x_{1j} x_{dj} = \sum_{j=1}^{m} x_{1j} y_j; \\ \ldots \\ a_0 \sum_{j=1}^{m} x_{dj} + a_1 \sum_{j=1}^{m} x_{dj} x_{1j} + a_2 \sum_{j=1}^{m} x_{dj} x_{2j} + ... + a_d \sum_{j=1}^{m} x_{dj}^2 = \sum_{j=1}^{m} x_{dj} y_j. \end{cases} \qquad (3)$$

The resulting system of $d+1$ equations with $a_0, a_1, \ldots, a_d$ of unknowns can be solved by means of linear algebra. For many equations, it is best to use the Gaussian method with the choice of the main element. Since the matrix of this system of linear algebraic equations is symmetric, its solution always exists and is unique. If the number of equations is small, the inverse matrix method can be successfully used to solve the problem.

Stage 2. Verification of the adequacy of the model obtained. For this purpose, it will be necessary to do the calculation of

- the model residuals, i.e. the differences between the observed and calculated values:

$$u_h = y_h - \hat{y}h = y_h - (a_0 + a_1 x_{1h} + a_2 x_{2h} + \ldots + a_d x_{dh}), \; h = 1, 2, \ldots, m; \qquad (4)$$

- the relative error of the residuals and their mean:

$$\delta_h = \frac{u_h}{y_h} \cdot 100\%, \; \delta = \frac{\sum_{h=1}^{m} \delta_h}{m}; \qquad (5)$$

- root mean square error of the disturbance variance:

$$\sigma_u = \sqrt{\frac{\sum_{h=1}^{m} u_h^2}{m - d - 1}}; \qquad (6)$$

- coefficient of determination:

$$R^2 = 1 - \frac{\sum_{h=1}^{m} u_h^2}{\sum_{h=1}^{m} (y_h - \bar{y})^2} \; або \; R^2 = 1 - \frac{\sum_{h=1}^{m} (y_h - \hat{y})^2}{\sum_{h=1}^{m} (y_h - \bar{y})^2}; \qquad (7)$$

- the multiple correlation coefficient R, which is the main indicator of the density of the correlation between the generalized indicator and the factors:

$$R = \sqrt{1 - \frac{\sum_{h=1}^{m} (y_h - \hat{y}_h)^2}{\sum_{h=1}^{m} (y_h - \bar{y})^2}}. \qquad (8)$$

All values of the correlation coefficient R are in the interval from -1 to 1. The sign of the coefficient indicates the "direction" of the relationship: a positive value indicates a "direct" relationship, a negative value indicates an "inverse" relationship, and a value of "0" indicates the absence of a linear correlation. When $R=1$ or $R=-1$, we have a functional relationship between the features. The multiple correlation coefficient R is the main measure of the closeness of the relationship between the resultant trait and the set of factor traits.

Stage 3. Verification of the statistical significance of the results. The experiment is performed using Fisher's statistics with d and $(m-d-1)$ degrees of freedom:

$$F=\frac{\frac{\sum_{h=1}^{m}(\hat{y}_h-\overline{y})^2}{d}}{\frac{\sum_{h=1}^{m}(y_h-\hat{y}_h)^2}{m-d-1}} \text{ or } F=\frac{R^2}{1-R^2}\cdot\frac{m-d-1}{d}, \quad (9)$$

where d is several **DB** factors, included in the model; m is a total number of observations; $\hat{y}_h$ is the estimated value of the dependent variable at the h-th observation; $\overline{y}$ is an average value of the dependent variable; y_h is the value of the dependent variable at the h-th observation; R is a multiple correlation coefficient.

Fisher's tables are used to find the critical value of $F_{кр}$ with d and $(m-d-1)$ degrees of freedom. If $F>F_{кр}$, this indicates that the model is adequate. If the model is inadequate, it is necessary to return to the model-building stage and possibly introduce additional factors or move to a non-linear model.

Stage 4. Determining the regression coefficients, the elasticity coefficient, and the confidence intervals for the regression parameters. It is necessary to verify the significance of the coefficients of the regression equation. The test is performed using the t-statistic, which has the form for multivariate regression parameters:

$$t_h=\frac{a_h}{\sigma^2_{a_h}}, \quad 10)$$

where σ_{a_h} is the standard deviation of the estimate of the h^{th} parameter.

If the t_h value exceeds the critical value found in the Student's t-test tables, the corresponding parameter is statistically significant and has a significant impact on the generalizing indicator.

Differences in the units of measurement of the **DB** factors are eliminated by using partial elasticity coefficients given by the ratio:

$$\varepsilon_h=\frac{\partial\hat{y}}{\partial x_h}\cdot\frac{\overline{x}_h}{\overline{y}}, \quad (11)$$

where x_h is the average value of the h-th parameter; $\overline{y}$ is the average value of the resultant trait.

The partial elasticity coefficient ε_h indicates how much the outcome variable changes on average for a 1% change in factor x_h ,while holding other parameters constant.

The confidence interval at the level of reliability (1-α) is an interval with randomly determined boundaries that covers the true value of the coefficient of the regression equation a_h with the level of confidence (1-α) and is specified by the dependencies:

$$(a_h-t_{\alpha/2,z}\sigma^2_{ah};a_h+t_{\alpha/2,z}\sigma^2_{ah}), \quad (12)$$

where $t_{\alpha/2,z}$ is Student's statistic with $z=m-d-1$ degrees of freedom and α as significance level; σ^2_{ah} is a standard deviation of the a_h parameter estimate.

Therefore, consider S random variables of $x_1,x_2,\ldots,x_r,\ldots,x_s$ (the parameters under study) represented by samples of v values $x_r=\{x_{r1},x_{r2},\ldots,x_{rz},\ldots,x_{rv}\}$. For each pair of random variables x_r and x_w, the empirical linear correlation coefficient r_{rw} can be estimated from the equation. The values of the coefficients obtained are written in a matrix of size $s\times s$:

$$\begin{pmatrix} 1 & r_{12} & \dots & r_{1w} & \dots & r_{1s} \\ r_{21} & 1 & \dots & r_{2w} & \dots & r_{2s} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ r_{r1} & r_{r2} & \dots & 1 & \dots & r_{rs} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ r_{s1} & r_{s2} & \dots & r_{sw} & \dots & 1 \end{pmatrix}. \quad (13)$$

All values of the correlation coefficient range from -1 to 1. The sign of the coefficient indicates the 'direction' of the relationship: a positive value indicates a 'direct' relationship, a negative value indicates an 'inverse' relationship, and a value of '0' indicates the absence of a linear correlation. When $r=1$ or $r=-1$, when there is a functional relationship between the attributes.

The multiple correlation coefficient is the main indicator of the closeness of the relationship between a set of databases **DB** and a set of factor attributes (main selection criteria) **EC**. The Chaddock scale is used to assess the strength of the relationship.

Thus, using the above calculation procedure of correlation and regression analysis, we have examined the set of **DB** used in modern SIEM systems according to the main **EC** criteria.

In Table 1, the set of DB defined in the works [11–13, 28] is presented in the following form:

$$\mathbf{DB} = \{\bigcup_{i=1}^{n} DB_i\} = \{DB_1, DB_2, \dots, DB_n\}, \quad (14)$$

where $DB_i \subseteq \mathbf{DB}\ (i=\overline{1,n})$ are types of DBMS used in certain SIEM systems, n is a total number of databases.

According to the analyzed systems, with $n=34$, considering (14), let's define the set of databases as follows:

$$\mathbf{DB} = \{\bigcup_{i=1}^{34} DB_i\} = \{DB_1, DB_2, \dots, DB_{34}\},$$

where DB_1 – Ariel database, DB_2 – PostgreSQL, DB_3 – SQLite, DB_4 – Oracle, DB_5 – SQL Server, DB_6 – MySQL, DB_7 – DB2/Linux, DB_8 – Informix, DB_9 – MemSQL, DB_{10} – AWS Aurora, DB_{11} – Microsoft SQL Server, DB_{12} – AWS RedShift, DB_{13} – SAP SQL Anywhere, DB_{14} – Sybase ASE, DB_{15} – Sybase IQ, DB_{16} – Teradata, DB_{17} – MSSQL, DB_{18} – Data Access Server (DAS), DB_{19} – DB2/UDB, DB_{20} – RedisDB, DB_{21} – Rap Sheet, DB_{22} – RabbitMQ, DB_{23} – MongoDB, DB_{24} – ElasticSearch, DB_{25} – Kibana, DB_{26} – MS SQL Express, DB_{27} – MariaDB, DB_{28} – SQL, DB_{29} – DB2, DB_{30} – Own development CORR-E, DB_{31} – Microsoft SQL Azure, DB_{32} – SYBASE, DB_{33} – IBM, DB_{34} – Hadoop as it advised in [12].

The **DB** set of databases was studied according to the main criteria introduced by the **EC** set:

$$\mathbf{EC} = \{\bigcup_{j=1}^{q} EC_j\} = \{EC_1, EC_2, \dots, EC_q\}, \quad (15)$$

where $EC_j \subseteq \mathbf{EC}\ (j=\overline{1,q})$ is a category of criteria for evaluating the most efficient DBMSs, q is a total number of criteria.

Therefore, for $q=7$, considering (15), let's define the set of proposed criteria **EC**:

$$\mathbf{EC} = \{\bigcup_{j=1}^{7} EC_q\} = \{EC_1, EC_2, EC_3, EC_4, EC_5, EC_6, EC_7\} =$$

$$= \{EC_{HOS}, EC_{FL}, EC_{FS}, EC_{STD}, EC_{SCD}, EC_{SSQL}, EC_{DBAAS}\},$$

where EC_1 is a highly organized structure, EC_2 is flexible, EC_3 is quick access, EC_4 is a support for different types of data, EC_5 is saving configuration data possibility, EC_6 is a structured query language support, EC_7 is the DBaaS (supported for cloud technologies).

As a result, the two most effective databases were studied and reasonably identified by the correlation-regression multifactor analysis (scheme presented at Fig.2):

1) DB_{23} which is MongoDB;

2) DB_{24}, which is ElasticSearch.

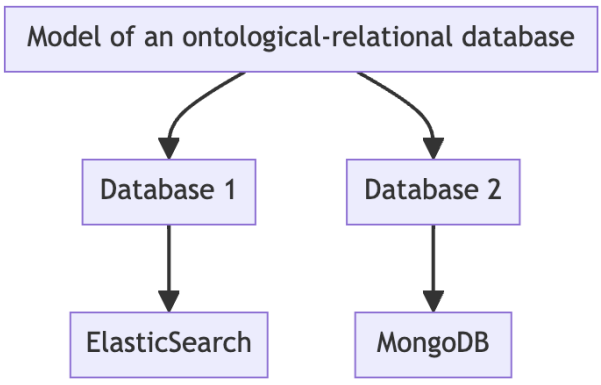

Figure 2: Scheme of implementation of the ontological-relational data store model

4. Ontology-Relational Data Store Model Development

The ontology-relational data store model consists of two types of DBs:

1) DB Type 1

Purpose: fast processing of logs.

To solve this problem, we chose the open-source Elasticsearch technology. Elasticsearch is perfectly designed to work with logs. After indexing, it is possible to search, sort, and filter data, not just columns of data. This again demonstrates a different approach to data retrieval and shows that Elasticsearch can perform complex full-text searches.

Documents are represented as JSON objects. JSON serialization (the process of translating any data structure into a sequence of bits) is supported by most programming languages and is a standard format for NoSQL.

Elasticsearch is an open-source full-text search platform based on the Lucene library and written in Java. It is designed to perform complex document/file-based searches. In the Elasticsearch tables are called indexes and the process of loading documents is called indexing. It can be thought of as both a non-relational document store in JSON format and a search engine based on Lucene's full-text search. The official clients are available in Java, NET (C#), Python, Groovy, JavaScript, PHP, Perl and Ruby. Elasticsearch is developed by Elastic and distributed under an open license. The Java code has been modified for the current model.

2) DB Type 2

Purpose: reliable storage of proprietary information.

To achieve this, we chose the open-source MongoDB technology. MongoDB is a document-oriented DBMS that does not require a table schema description. It is considered one of the classic examples of a NoSQL system that uses JSON-like documents and a DB schema. Written in C++, it is used in web development, particularly as part of the JavaScript-oriented MEAN stack.

The system can work with a set of replicas, i.e. two or more copies of data on different nodes. Each instance of the replica set can act as a primary or secondary replica at any time. By default, all write and read operations are performed on the primary replica. Auxiliary replicas keep the data copy up to date. If the primary replica fails, the replica set selects which replica should become the primary. Secondary replicas can also be a source for read operations.

The system is scaled horizontally using the technique of segmenting DB objects - distributing their parts to different cluster nodes. The administrator chooses the segmentation key, which determines the criteria by which data is distributed to the nodes (depending on the hash values of the segmentation key). The fact that any node in the cluster can receive requests ensures load balancing. The system can be used as a file storage with load balancing and data replication (Grid File System function). In addition, software tools are provided for working with files and their contents. GridFS is used in plugins for Nginx and lighttpd. GridFS splits a file into chunks and stores each chunk as a separate document. It is released under the AGPL open-source license.

5. Implementation of the Proposed Model

The model proposed in this paper (Figure 1) can be implemented as part of an event correlation and cybersecurity incident management system [13, 29]. When scaling resources in the developed SIEM, there are several practical rules:

- SIEM nodes focus on processor performance. They also serve as the user interface for the browser;
- Elasticsearch nodes should have as much RAM as possible and the fastest discs that can be used. It all comes down to I/O speed;
- MongoDB stores meta-information and configuration data and is not resource-intensive;
- received messages are only stored in Elasticsearch.

The main task of an ontological-relational data store for SIEM is to combine the operation of two types of DBs while preserving the possibility of clustering DBs of both types.

The proposed approach to organizing the operation of the ontological-relational data store model for an SIEM system allows the indexing service to access external data stores (with the data being correctly indexed and correctly displayed during searches), to scale

(cluster) with the growth of data volume, to support work with different queries (simple, complex, structured) and with different types of data, to allow aggregation, analysis, collection of entities, patterns, simplification of searches and high search speed.

In addition, a SIEM based on this model can work with a set of replicas (i.e. contain 2 or more copies of data on different nodes), scale horizontally using the technique of segmenting DB objects, and be used as file storage with load balancing and data replication (Grid File System function).

6. Conclusions

This paper has analyzed the modern types of DBs used in SIEM systems and shown that each type of DB remains relevant in its area, where the relationships between data are determined by the specific structure of the DBMS. When choosing a DB to build an SIEM system, it is important to consider factors such as ease of data storage, speed of data retrieval, and ease of use. It is also worth considering the possibility of integration with other system modules and external APIs to support a variety of DB for most DPI systems (comprehensive deep inspection content analyzers), both software and hardware. In addition, the possibility of using hybrid DBs that combine different types, such as SQL and NoSQL, should be considered.

A model of an ontological-relational data store has been developed. It uses two different DBs, Elasticsearch and MongoDB, with appropriate characteristics, and allows to improve the convenience of storing and classifying data, as well as to ensure high speed of obtaining large amounts of information through pre-indexing, horizontal scaling by segmenting DB objects, as well as load balancing and data replication [30].

References

[1] P. Anakhov, et al., Increasing the Functional Network Stability in the Depression Zone of the Hydroelectric Power Station Reservoir, in: Workshop on Emerging Technology Trends on the Smart Industry and the Internet of Things, vol. 3149 (2022) 169–176.

[2] P. Anakhov, et al., Evaluation Method of the Physical Compatibility of Equipment in a Hybrid Information Transmission Network, Journal of Theoretical and Applied Information Technology 100(22) (2022) 6635–6644.

[3] V. Sokolov, et al., Method for Increasing the Various Sources Data Consistency for IoT Sensors, in: IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (2023) 522–526. doi: 10.1109/PICST57299.2022.10238518.

[4] H. Hulak, et al., Dynamic Model of Guarantee Capacity and Cyber Security Management in the Critical Automated Systems, in: 2nd International Conference on Conflict Management in Global Information Networks, vol. 3530 (2022) 102–111.

[5] V. Astapenya, et al., Analysis of Ways and Methods of Increasing the Availability of Information in Distributed Information Systems, in: IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (2021). doi: 10.1109/picst54195.2021.9772161.

[6] M. Vielberth, G. Pernul, A Security Information and Event Management Pattern, in: 12th Latin American Conference on Pattern Languages of Programs (2018).

[7] K. Agrawal, H. Makwana, A Study on Critical Capabilities for Security Information and Event Management, Int. J. Sci. Res. 4(7) (2015) 1893–1896.

[8] H. Karlzén, An Analysis of Security Information and Event Management Systems. Department of Computer Science and Engineering Chalmers University of Technology University of Gothenburg, Göteborg, Sweden (2009). http://publications.lib.chalmers.se/records/fulltext/89572.pdf

[9] D. Ribolovlev, S. Karasov, S. Polyakov, Classification of Emergency Management Systems for Incidents without Baking, Food Cyber Secur. 3(27) (2018) 47–53.

[10] Ariel Query Language Guide, IBM QRadar 7.3.3. https://www.ibm.com/docs/en/SS42VS_7.3.3/com.ibm.qradar.doc/b_qradar_aql.pdf

[11] S. Gnatyuk, et al., Modern Types of Databases for SIEM System Development, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3187 (2021) 127–138.
[12] S. Gnatyuk, et al., Model of Information Technology for Efficient Data Processing in Cloud-based Malware Detection Systems of Critical Information Infrastructure, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3421 (2023) 206–213.
[13] S. Gnatyuk, et al., Event Correlation and Incident Management System for Cybersecurity of Critical Infrastructure Objects, Cybersecur. Edu. Sci. Technol. 3(19) (2023) 176–196.
[14] S. Sekharan, K. Kandasamy, Profiling SIEM Tools and Correlation Engines for Security Analytics, in: International Conference on Wireless Communications, Signal Processing and Networking, Chennai, India (2017) 717–721. doi: 10.1109/WiSPNET.2017.8299855.
[15] J. Lee, et al., Toward the SIEM Architecture for Cloud-based Security Services, in: IEEE Conference on Communications and Network Security (CNS), Las Vegas, NV (2017) 398–399. doi: 10.1109/CNS.2017.8228696.
[16] I. Bachane, Y. I. K. Adsi, H. C. Adsi, Real Time Monitoring of Security Events for Forensic Purposes in Cloud Environments using SIEM, in: 3rd International Conference on Systems of Collaboration (SysCo) (2016) 1–3, doi: 10.1109/SYSCO.2016.7831327.
[17] B. AlSabbagh, S. Kowalski, A Framework and Prototype for A Socio-Technical Security Information and Event Management System (ST-SIEM), in: European Intelligence and Security Informatics Conference (EISIC) (2016) 192–195, doi: 10.1109/EISIC.2016.049.
[18] A. Serckumecka, I. Medeiros, A. Bessani, Low-Cost Serverless SIEM in the Cloud, in: 38th Symposium on Reliable Distributed Systems (SRDS) (2019). doi: 10.1109/SRDS47363.2019.00057.
[19] M. Nabil, et al., SIEM Selection Criteria for an Efficient Contextual Security, in: International Symposium on Networks, Computers and Communications (ISNCC) (2017) 1–6, doi: 10.1109/ISNCC.2017.8072035.
[20] R.-V. Mahmoud, et al., DefAtt—Architecture of Virtual Cyber Labs for Research and Education, in: International Conference on Cyber Situational Awareness Data Analytics and Assessment (CyberSA), pp. 1–7, 2021.
[21] Y. Danik, R. Hryschuk, S. Gnatyuk, Synergistic Effects of Information and Cybernetic Interaction in Civil Aviation, Aviat. 20(3) (2016) 137–144.
[22] R. Berdibayev, et al., A Concept of the Architecture and Creation for SIEM System in Critical Infrastructure, Stud. Syst. Decis. Control 346 (2021) 221–242.
[23] O. Oksiiuk, V. Chaikovska, A. Fesenko, Security Technique for Authentication Process in the Cloud Environment, in: IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (2019) 379–382, doi: 10.1109/PICST47496.2019.9061248.
[24] S. Gnatyuk, et al., Studies on Cloud-based Cyber Incidents Detection and Identification in Critical Infrastructure, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 2923 (2021) 68–80.
[25] N. Lukova-Chuiko, et al., Threat Hunting as a Method of Protection Against Cyber Threats, in: Information Technology and Interactions, vol. 2833 (2021) 103–113.
[26] A. Yushko, et al., Shielding Web Application against Cyber-Attacks using SIEM, in: 13th International Conference on Advanced Computer Information Technologies (ACIT), Wrocław, Poland (2023) 393–396, doi: 10.1109/ACIT58437.2023.10275630.
[27] J. Song, et al., Data Consistency Management in an Open Smart Home Management Platform (2014). doi: 10.1109/EMS.2014.51.
[28] A. Polozhentsev, et al., Novel Cyber Incident Management System for 5G-based Critical Infrastructures, in: IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and

Applications, IDAACS (2023) 1037–1041.
[29] A. Tikhomirov, et al., Network Society: Aggregate Topological Models, Communications in Computer and Information Science, Verlag: Springer International Publ, vol. 487 (2014) 415–421.
[30] M. Nabil, et al., SIEM Selection Criteria for an Efficient Contextual Security, in: International Symposium on Networks, Computers and Communications, Marrakech, Morocco (2017) 1–6, doi: 10.1109/ISNCC.2017.8072035.