Hybrid method for detecting cyber threats in the network traffic*

Valeriy Lakhno[1,†], Sergiy Mamchenko[1,†], Alona Desiatko[2,*,†], Bohdan Bebeshko[3,†] and Ihor Mirko[2,†]

[1] National University of Life and Environmental Sciences of Ukraine, 15 Heroes of Defense str., 03041 Kyiv, Ukraine
[2] State University of Trade and Economics, 19 Kyoto str., 02156 Kyiv, Ukraine
[3] Borys Grinchenko Kyiv Metropolitan University, 18/2 Bulvarno-Kudryavska str., 04053 Kyiv, Ukraine

Abstract

Modern cyber threats are characterized by a high degree of adaptability, secrecy and variability. This fact makes the task of their timely detection in network traffic one of the key problems in the field of cyber security of informatization objects. Traditional methods based on signatures and rigidly defined rules do not provide sufficient flexibility to detect previously unknown or modified attacks. Consequently, the relevance of developing new hybrid intelligent systems capable of taking into account the behavioral characteristics of traffic and adapt to its dynamics is increasing. The paper proposes a hybrid method for detecting cyber threats that combines the advantages of ensemble clustering and Bayesian probabilistic modeling. In the first stage, a machine learning model extracts the hidden behavioral features of network connections using multiple clustering algorithms. And the obtained behavioral embeddings are further used as input variables to construct a Bayesian network that models the probabilistic dependencies between behavioral attributes and anomaly attributes. The outlined approach will allow not only to detect abnormalities in traffic, but also to ensure the interpretability of the adopted security decisions. The practical significance of the proposed method lies in the potential of its integration into traffic monitoring systems in corporate and distributed network infrastructures.

Keywords

cyber threats, network traffic, behavioral analysis, Bayesian network, clustering, machine learning, anomaly, hybrid method, cybersecurity, attack detection

1. Introduction

Computer networks of informatization objects, being in a state of constant development. And as information technologies evolve and business processes become more dependent on network operations, they become increasingly vulnerable to a wide range of cyber threats. Such cyber threats manifest themselves in the form of attacks of different nature, ranging from unauthorized access to complex multi-stage intrusions [1, 2]. With increasing volumes of network traffic and increasing complexity of user behavioral patterns, traditional methods of threat and anomaly detection based on signatures and static heuristics described in [2–6], as shown in [6–11], are losing their effectiveness. This circumstance is caused not only by the high dynamics of cyber threats, but also by the need for rapid adaptation to new types of attacks. And such new attacks often do not have predetermined templates [11, 12].

This paper considers a new hybrid approach to network traffic analysis combining machine learning (ML) and probabilistic modeling methods based on Bayesian networks (BN). The proposed method is based on multi-stage processing of observed network features. First, latent behavioral representations reflecting the structure of interactions and behavioral heterogeneity of network activity are extracted using ensemble clustering. And then a Bayesian model is built based on them,

* CSDP'2025: Cyber Security and Data Protection, July 31, 2025, Lviv, Ukraine
* Corresponding author.
† These authors contributed equally.
lva964@nubip.edu.ua (V. Lakhno); s.mamchenko@nubip.edu.ua (S. Mamchenko); desyatko@gmail.com (A. Desiatko); b.bebeshko@kubg.edu.ua (B. Bebeshko); igorkmi51@ukr.net (I. Mirko)
0000-0001-9695-4543 (V. Lakhno); 0009-0006-8743-5606 (S. Mamchenko); 0000-0002-2284-3418 (A. Desiatko); 0000-0001-6599-0808 (B. Bebeshko); 0009-0000-9242-5572 (I. Mirko)

allowing probabilistic inferences about whether the traffic belongs to normal or abnormal category. The method proposed in this paper integrates the advantages of trained behavioral models and explainability of probabilistic structures, providing high adaptability to the variety of cyber threats while maintaining the interpretability of the results obtained.

Modern cybersecurity challenges require comprehensive approaches that integrate secure system design, effective data protection, and adaptive threat response mechanisms. For example, methods of secure digital system design, including protection against SQL injection and related attacks, demonstrate the importance of embedding security at all stages of system development [13, 14]. Furthermore, establishing an Information Security Management System (ISMS) tailored to counter evolving cyber threats provides a structured framework for proactive risk management and resilience [15–21]. In this context, designing secured services for authentication, authorization, and accounting (AAA) plays a critical role in strengthening network security and supporting the detection of abnormal user behaviors [22].

2. Problem statement

The problem of detecting cyber threats in network traffic is formulated as the problem of detecting anomalous behavior that characterizes potentially dangerous or malicious activities in the flow of network connections. The main difficulty is that attacker behavior may vary from case to case. Consequently, such data will not be pre-represented in training samples (which is the basis for many researchers, e.g., in [2–10]). In addition, anomalies are often behavioral in nature and manifest themselves not in the values of individual traits, but in deviations from typical activity patterns.

Existing methods [8, 9, 11] either use static rules and signatures that are unable to cope with unknown attacks, or employ ML methods that, although capable of detecting complex dependencies, often suffer from a lack of interpretability and an inability to account for causal relationships. In environments where detection accuracy, robustness to false positives, and the ability to explain results simultaneously are required, there is a need for complex models that combine empirical adaptation with probabilistic interpretation.

The objective is to develop a method capable of detecting cyber threats based on behavioral analysis of network traffic, relying on latent patterns detected by ML and a Bayesian network (BN) adept at modeling probabilistic relationships between behavioral characteristics and anomalies.

3. Methods and models

Step 1: Extraction of behavioral features using ensemble clustering.

Let there is a sample of network traffic:

$$\chi = \left\{x^{(1)}, x^{(2)}, \dots, x^{(n)}\right\}, x^{(i)} \in R^d, \tag{1}$$

where $x^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \dots, x_d^{(i)}\right)$ is the vector of features for i^{th} network connection (connection duration, bytes, protocol, etc.).

Next, we select a set of M clustering algorithms:

$$C = \left\{C_1, C_2, \dots, C_M\right\}.$$

Here, each C_j is a function $C_j : R^d \longrightarrow \left\{1,2,\dots,K_j\right\}$ that mps a cluster label $x^{(i)}$ to each vector $z_j^{(i)} = C_j\left(x^{(i)}\right)$.

Then we obtain the following intermediate result

$$Z^{(i)} = \left(z_1^{(i)}, z_2^{(i)}, \dots, z_M^{(i)}\right) \in \prod_{j=1}^{M} \left\{1, \dots, K_j\right\}. \tag{2}$$

At this stage, the goal is to identify hidden behavioral patterns in network traffic by applying several clustering algorithms to the data. This allows us to form a generalized view of possible traffic patterns. For example, suppose that each object in the sample is a distinct network connection described by a vector of features. As we mentioned earlier, e.g., connection duration, number of bytes transferred, protocol type, etc. This data is input to several clustering algorithms that make up the ensemble. Such well-known methods as K-means, DBSCAN, hierarchical clustering, spectral clustering and others can be chosen as algorithms. Note that the choice of specific algorithms for an ensemble is determined by considering several factors. Among these factors we consider differences in density and distance approximation methods (density methods or metric methods), sensitivity to the shape and size of clusters, robustness to noise and outliers, and scalability when dealing with large volumes of traffic. Therefore, by applying each of the algorithms to the raw data, we obtain that each network connection is assigned to a cluster. Thus, each connection is assigned a label, i.e., the cluster identifier to which it is assigned by the algorithm. Since multiple algorithms are used, a vector of labels is generated for each connection, where each component corresponds to the result of clustering by one of the methods. The intermediate result of this step is then the so-called "cluster representation" of the connection behavior. This is the set of cluster labels obtained from the results of all ensemble methods. This set of labels should not be interpreted directly as features in the classical sense. Instead, it serves as a basis for further construction of numerical behavioral features reflecting the consistency or divergence in the estimates of the different clustering methods. These features will be used as input variables in the probabilistic model of the Bayesian network at the next stages of the method, allowing it to take into account and interpret the behavioral patterns revealed by clustering.

To move to a numerical vector of behavioral traits, a mapping is used:

$$\Phi : \prod_{j=1}^{M} \{1, \dots, K_j\} \longrightarrow R^k, \tag{3}$$

where Φ is a mapping (function) that transforms the original cluster labels into a vector of behavioral features. I.e., Φ performs the role of embedding, taking the clustering results (cluster labels from each algorithm) and transforming them into a numerical vector of fixed length; K_j is the result of the j^{th} clustering method (from an ensemble of M methods) over object x_i. Or $K_j(x_j)$ is the cluster label assigned by the j^{th} clustering method for i^{th} network connection; R^k is a vector of k real numbers. The cluster embedding vector for the i-th network connection is a feature vector of dimension d, suitable to be fed to the BN input. The dimensionality d depends on the number of methods in the ensemble (M), and the number of clusters output by each method.

Then, after a vector of cluster labels obtained from several clustering algorithms has been generated for each network connection, there is a need to convert this vector into a numerical representation. This representation should be suitable for further analysis within a Bayesian model. This transformation is called cluster embedding [23, 24]. The cluster labels obtained in the previous step are categorical values (e.g., "cluster 1", "cluster 3", etc.) that do not carry quantitative meaning and cannot be directly used in a Bayesian network (BN), where variables require either numerical or strictly probabilistic representation. In addition, it is fundamental to obtain noise robust features. Then these features will reflect the data structure revealed on the basis of several clustering methods rather than the results of a single method. This is done by mapping the label vector into a new feature space. Or a behavioral feature space where each component reflects the object's participation in certain clusters. This transformation can be realized by different methods. Let us consider a concrete example. Suppose the following labels from three algorithms are obtained for a single connection: K-means: categorized the connection as cluster 2 out of 4 possible clusters. DBSCAN identified the connection as "noise" (did not assign it to any cluster). Hierarchical clustering placed the connection in cluster 3 out of 5 possible clusters. Hence, we obtain the label vector:

[Kmeans=2, DBSCAN=Noise, Hierarchical=3]

To turn this set into a numerical vector (embedding), one-hot encoding can be applied, for example. Then K-means → [0, 1, 0, 0, 0] (cluster 2 of 4 is active). DBSCAN → [0, 0] (assume two valid clusters, and the "noise" is encoded with zeros). And Hierarchical → [0, 0, 0, 1, 0, 0] (cluster 3 of 5 is active).

Combining everything, we get an embedding of length 11. That is, [0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0]. This vector can already be used in subsequent probabilistic models because it is numerically interpretable and reflects the characteristics of the object. Moreover, this vector aggregates information from several clusters at once and does not depend on the initial numerical features.

The main goal is to provide the BN with "clean" and high-level attributes that reflect the structure of behavior, not just technical traffic parameters. Accordingly, the BN does not work with raw data, but with stable and interpretable behavioral characteristics. This is essential for the task of detecting anomalies in the network, where "anomaly" itself is most often manifested as a behavioral deviation from the norm.

We end up with a behavioral profile of an object $x^{(i)}$, extracted based on cluster membership.

Then the next step is to build a Bayesian network on top of the extracted features. A Bayesian network is a probabilistic model describing the relationship between random variables [25, 26].

Stage 2: Formalize the composition of BN nodes.

Let us let $Z=\left(Z_1, Z_2, \ldots, Z_k\right)$ is a variables corresponding to the components $h^{(i)}$; $A\in\{0,1\}$ is a binary random variable reflecting anomaly (A = 1 is an anomaly, A = 0 is a standard).

Here $h^{(i)}$ is the hidden (latent) variable characterizing the anomaly of the i^{th} object. For example, a network connection. This is not a directly observable characteristic, but a latent hypothesis about whether the object belongs to the class of "normal traffic" or "anomaly". In fact, it is a target node in the BN modeling the hypothesis of whether an object is a potential cyber threat.

We combine all variables into a set $v=\left(Z_1, \ldots, Z_k, A\right)$. The set v represents the complete set of variables used in the BN. We combine them to define the structure of the BN. Or, in other words, to establish between which variables probabilistic dependencies can exist. Also, v is needed to specify the factorization domain, since the BN is constructed as a factorization of the joint distribution of all variables from v, using a directed acyclic graph (DAG) [8]. Or in formalized form G= $=\left(V, E\right), V=v, E=V\times V$. Each edge $\left(Z_i, A\right)\in E$ s interpreted as "a feature of behavior Z_i affects the probability that the connection is an anomaly".

The network structure, represented as an oriented acyclic graph, serves as a "framework" for modeling probabilistic relationships between variables. By determining which nodes (variables such as the latent variable characterizing abnormality and extracted behavioral traits) are directly connected, we form the basis for factoring the joint probability distribution. In other words, this means that each variable in the network is considered together with the set of its immediate antecedents, thus allowing an adequate description of its probabilistic behavior. The transition to a joint distribution is made by decomposing the full probability measure into a product of conditional distributions, where each component corresponds to a node in the graph and depends only on the variables to which it is directly related in the structure (in other words, on its parents in the graph).

The joint distribution in the BN is defined as follows:

$$P\left(Z_1, \ldots, Z_k, A\right)=\prod_{v\in V} P\left(v\middle| pa(v)\right), \tag{4}$$

where $pa\left(v\right)|V$ is a set of node parents v.

The main goal is to compute the posterior distribution of the:

$$P\left(A=1\middle| Z_1=z_1, \ldots, Z_k=z_k\right), \tag{5}$$

which is the prediction of the anomaly probability for the object $x^{(i)}$

The joint distribution expresses the complete set of dependencies extracted during graph construction and then allows for the conditional probability calculations required for cyber threat

detection. Consequently, combining the concepts of the method steps of deriving the BN structure and the joint distribution reflects the fundamental idea of the Bayesian approach, in which a predefined dependency structure defines the rules by which the complex probability distribution underlying the network traffic anomaly detection model can be decomposed and described.

After formalizing the joint distribution of all variables included in the BN structure, the next logical step is the model training procedure. BN training is a process of determining numerical parameters corresponding to conditional probabilities in the nodes of the graph. For each variable included in the network, it is required to estimate its conditional distribution given by its parents in the dependency graph. If the network structure is fixed in advance (e.g., relying on a priori knowledge of experts), the learning task is reduced to estimating the parameters of the distributions. In the case of discrete variables, which include both the latent variable reflecting anomaly and the features obtained from cluster embedding, training is performed by calculating the relative frequencies of the training sample. Then, likelihood maximization, i.e., parametric tuning of the model so that it best explains the observed network traffic data, is provided. Actually, BN training represents the central stage of our method. At this stage, the probabilistic model acquires a concrete numerical content that reflects the statistical relationship between traffic behavioral attributes and the anomaly hypothesis. The trained model will later serve as a basis for probabilistic inference, allowing us to assess the degree to which new network connections belong to potentially dangerous ones based on their behavioral profile.

4. Results of the study

The proposed method for analyzing network traffic to detect cyber threats can be conceptualized in the form of expression (6) or a conceptual diagram of a formalized algorithm, see Figure 1.

In other words, the method can be conceptualized as a composition of sequentially applied mappings, see expression (6). And each of them implements a certain functional transformation over the data, bringing us closer to the formation of a probabilistic model of behavior.

$$x^{(i)} \underset{\rightarrow}{C} Z^{(i)} \underset{\rightarrow}{\Phi} h^{(i)} BN \underset{\rightarrow}{inference} P\left(A=1 \middle| h^{(i)}\right). \tag{6}$$

The input is the observed characteristics of network connections, representing low-level network features $x^{(i)}$. These data pass through a machine learning block, in particular ensemble clustering, which results in the transformation of the original observations into a behavioral representation. This stage can be interpreted as the first mapping. This first mapping identifies hidden behavioral patterns characteristic of different types of activity in the network and encodes them as embeddings.

The next step is the second mapping, the construction of a Bayesian network on the obtained embeddings, i.e., $h^{(i)} BN \underset{\rightarrow}{inference} P\left(A=1 \middle| h^{(i)}\right)$. This mapping establishes probabilistic relationships between behavioral attributes and the connection anomaly hypothesis, allowing us to combine the behaviors captured by clusters with a probabilistic model adapted to account for uncertainty, causality, and partially observed data.

The final model is thus a composite of transformations sequentially moving from the original network attributes to a probabilistic risk assessment.

In Figure 1, $a^{(i)}$ is a binary (or categorical variable) indicating which class the network connection actually belongs to. And the parameter $\hat{A}^{(i)}$ is the model prediction, i.e., the result of probabilistic inference in BN. Usually $\hat{A}^{(i)}$ means

$$\hat{A}^{(i)} = \begin{cases} 1, \text{if } P\left(h^{(i)}=1 \middle| z^{(i)}\right) > \tau, \\ 0, else, \end{cases} \tag{6}$$

where τ is the preselected threshold.

For example, $a^{(i)}$ is a true label from the dataset used for quality assessment. Then $\hat{A}^{(i)}$ is the prediction of the model, i.e., the result of Bayesian inference about traffic anomalies.

The proposed approach is a development of existing methods for detecting anomalies in traffic, and combines the ideas of behavioral analysis and probabilistic inference. The key novelty of the proposed solution is the use of ensemble clustering as a mechanism for extracting hidden behavioral features, which are then used not just for classification, but for building a dynamically adaptive BN. Such a network not only identifies threats, but is also able to account for the variability of network activity. For example, this is relevant in the face of constantly evolving cyber threats. We believe that the method outlined in this paper, extends the classical "clustering → labeling" scheme and proposes a flexible hybrid model. The model eventually combines empirical behavior and probabilistic interpretation, giving the outlined method a good.

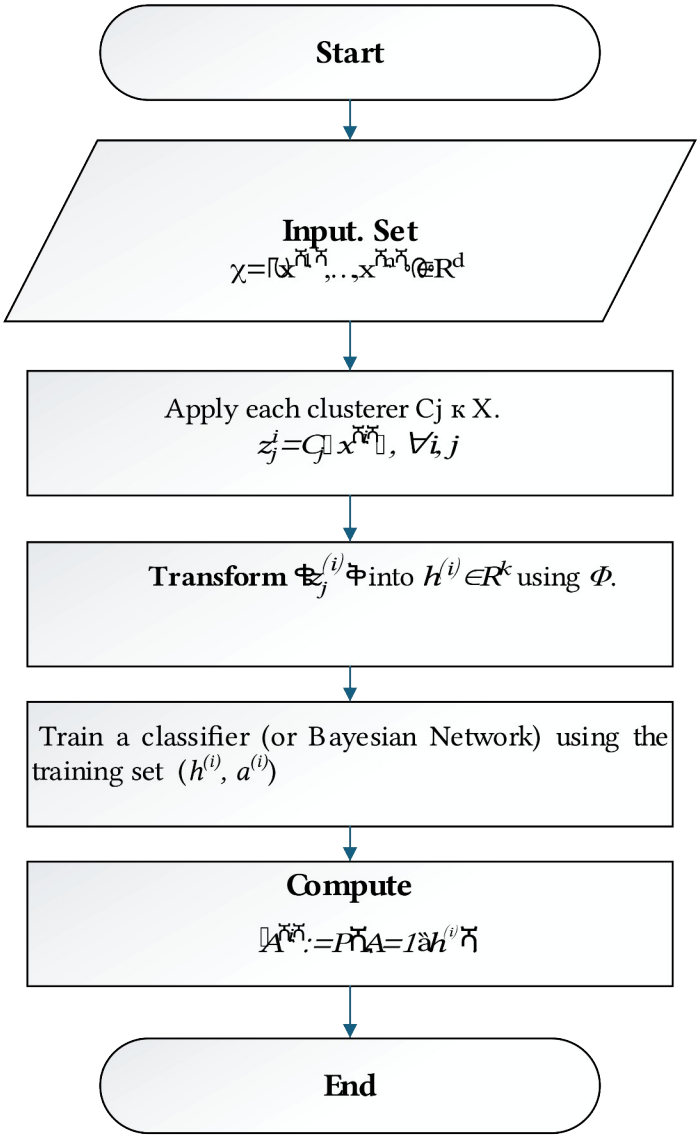

Figure 1: Conceptual diagram of a formalized algorithm for a hybrid method for detecting cyber threats in network traffic

Conclusions

The hybrid method of network cyber threat detection proposed in this paper combines the capabilities of behavioral modeling and probabilistic inference. The method forms an approach to analyzing network traffic under conditions of uncertainty of initial parameters for traffic analysis.

The use of ensemble clustering in the method at the first stage will allow to extract stable and informative representations of behavioral features. And the construction of a Bayesian network on their basis at the second stage, respectively, will ensure the feasibility of interpretable inference and consideration of causal dependencies between traffic parameters and threats to network security. The approach presented in the paper allows us to flexibly adapt the model structure to changes in user behavior and the dynamics of attacking strategies. It should be noted that the presented conceptual scheme of the formalized algorithm for the hybrid method of detecting cyber threats in network traffic opens a number of directions for further research. In our opinion, it is promising to develop methods of automatic learning of the BS structure based on the analysis of traffic dynamics. As well as the introduction of additional sources of information, such as temporal and/or contextual dependencies, into the learning model.

Declaration on Generative AI

While preparing this work, the authors used the AI programs Grammarly Pro to correct text grammar and Strike Plagiarism to search for possible plagiarism. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

References

[1] M. H. Bhuyan, D. K. Bhattacharyya, J. K. Kalita, Network Traffic Anomaly Detection and Prevention: Concepts, Techniques, and Tools, Springer, 2017. doi:10.1007/978-3-319-65188-0

[2] M. H. Bhuyan, D. K. Bhattacharyya, J. K. Kalita, Network Anomaly Detection: Methods, Systems and Tools, IEEE Communications Surveys & Tutorials, 16(1) (2013) 303–336. doi:10.1109/SURV.2013.052213.00046

[3] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, E. Vázquez, Anomaly-based Network Intrusion Detection: Techniques, Systems and Challenges, Comput. Secur., 28(1–2) (2009) 18–28. doi:10.1016/j.cose.2008.08.003

[4] M. Ahmed, A. N. Mahmood, J. Hu, A Survey of Network Anomaly Detection Techniques, J. Netw. Comput. Appl. 60 (2016) 19–31. doi:10.1016/j.jnca.2015.11.016

[5] L. F. Carvalho, T. Abrao, L. de Souza Mendes, M. L. Proença Jr, An Ecosystem for Anomaly Detection and Mitigation in Software-Defined Networking, Expert Systems with Appl. 104 (2018) 121–133. doi:10.1016/j.eswa.2018.03.027

[6] A. Patcha, J. M. Park, An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends, Comput. Netw. 51(12) (2007) 3448–3470. doi:10.1016/j.comnet.2007.02.001

[7] N. Jeffrey, Q. Tan, J. R. Villar, A Review of Anomaly Detection Strategies to Detect Threats to Cyber-Physical Systems, Electronics, 12(15) (2023) 3283. doi:10.3390/electronics12153283

[8] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, E. Akin, A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions, Electronics, 12(6) (2023) 1333. doi:10.3390/electronics12061333

[9] R. Samrin, D. Vasumathi, Review on Anomaly based Network Intrusion Detection System, in: Proc. 2017 Int. Conf. on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), IEEE, 2017, 141–147. doi:10.1109/ICEECCOT.2017.8284655

[10] Z. Yang, X. Liu, T. Li, D. Wu, J. Wang, Y. Zhao, H. Han, A Systematic Literature Review of Methods and Datasets for Anomaly-based Network Intrusion Detection, Comput. Secur. 116 (2022) 102675. doi:10.1016/j.cose.2022.102675

[11] A. Alshamrani, S. Myneni, A. Chowdhary, D. Huang, A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities, IEEE Communications Surveys & Tutorials, 21(2) (2019) 1851–1877. doi:10.1109/COMST.2019.2891891

[12] P. Bereziński, B. Jasiul, M. Szpyrka, An Entropy-based Network Anomaly Detection Method, Entropy, 17(4) (2015) 2367–2408. doi:10.3390/e17042367

[13] M. Szmajda, V. Khoma, Y. Khoma, V. Otenko, A Method of Rheographic System Design, that is based on the Wide use of Digital Components / Zastosowanie technologii cyfrowego przetwarzania sygnałów w nowoczesnych układach reograficznych, Przegląd Elektrotechniczny, 95(11) (2019) 233–239. doi:10.15199/48.2019.11.54
[14] Y. Momryk, D. Sabodashko, Numeric Fields in Database Development: From Optimal Design to Secure Coding—SQL Attacks Protection Method, in: Cybersecurity Providing in Information and Telecommunication Systems, 3991, 2025, 84–96.
[15] V. Susukailo, I. Opirsky, O. Yaremko, Methodology of ISMS Establishment Against Modern Cybersecurity Threats, Lecture Notes in Electrical Engineering, Springer, Cham (2021) 257–271. doi:10.1007/978-3-030-92435-5_15
[16] Y. Kostiuk, et al., A System for Assessing the Interdependencies of Information System Agents in Information Security Risk Management using Cognitive Maps, in: 3rd Int. Conf. on Cyber Hygiene & Conflict Management in Global Information Networks (CH&CMiGIN), Kyiv, Ukraine, vol. 3925, 2025, 249–264.
[17] Y. Kostiuk, et al., Models and Algorithms for Analyzing Information Risks during the Security Audit of Personal Data Information System, in: 3rd Int. Conf. on Cyber Hygiene & Conflict Management in Global Information Networks (CH&CMiGIN), Kyiv, Ukraine, vol. 3925, 2025, 155–171.
[18] Y. Kostiuk, et al., Integrated Protection Strategies and Adaptive Resource Distribution for Secure Video Streaming over a Bluetooth Network, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3826 (2024) 129–138.
[19] O. Milov et al., Development of Methodology for Modeling the Interaction of Antagonistic Agents in Cybersecurity Systems, Eastern-European J. Enterp. Technol. 2.9 (98) (2019) 56–66. doi:10.15587/1729-4061.2019.164730
[20] I. Hanhalo, et al., Adaptive Approach to Ensuring the Functional Stability of Corporate Educational Platforms under Dynamic Cyber Threats, in; Cybersecurity Providing in Information and Telecommunication Systems, vol. 3991 (2025) 481–491.
[21] S. Vasylyshyn, et al. A Model of Decoy System based on Dynamic Attributes for Cybercrime Investigation, Eastern-European J. Enterp. Technol. 1.9 (121) (2023) 6–20. doi:10.15587/1729-4061.2023.273363
[22] D. Shevchuk, O. Harasymchuk, A. Partyka, N. Korshun, Designing Secured Services for Authentication, Authorization, and Accounting of Users, in: Cybersecurity Providing in Information and Telecommunication Systems (CPITS-2023-II), 3550, 2023, 217–225.
[23] J. Xie, R. Girshick, A. Farhadi, Unsupervised Deep Embedding for Clustering Analysis, in: Proc. Int. Conf. on Machine Learning (ICML), PMLR, 2016, 478–487. doi:10.5555/3045390.3045442
[24] Z. Jiang, Y. Zheng, H. Tan, B. Tang, H. Zhou, Variational Deep Embedding: An Unsupervised and Generative Approach to Clustering, arXiv, 2016. doi:10.48550/arXiv.1611.05148
[25] S. H. Haji, S. Y. Ameen, Attack and Anomaly Detection in IoT Networks using Machine Learning Techniques: A Review, Asian J. Res. Comput. Sci., 9(2) (2021) 30–46.
[26] D. C. Le, N. Zincir-Heywood, Anomaly Detection for Insider Threats using Unsupervised Ensembles, IEEE Transactions on Network and Service Management, 18(2) (2021) 1152–1164. doi:10.1109/TNSM.2021.3071928