Critical Use of AI/ML in Cybersecurity, Including Generative Artificial Intelligence

Małgorzata Plichta, Damian Kostycz, Paweł Murdzek, Piotr Szewczyk
Institute of Telecommunications, Warsaw University of Technology

***Abstract*—In the face of dynamically changing standards and constantly developing technologies, consistent and effective methods enabling successful cybersecurity management are often lacking. One of the main challenges currently faced by specialists in this field is the integration of complex IT systems with security tools and frameworks. This article presents the results of a literature review and experiments conducted using artificial intelligence and machine learning (AI/ML) methods on two datasets concerning the detection of network threats and APT attacks. Particular attention was paid to the detection of suspicious Windows user behaviors, analysis of network traffic anomalies, data processing stages (preprocessing), as well as the application of generative artificial intelligence techniques.**

***Index Terms*—cybersecurity, AI, ML, IDS, APT, generative AI, anomaly detection, cyberattacks, DDoS, Windows**

I. Introduction

Nowadays, in the era of artificial intelligence development and ubiquitous automation, it is impossible to ignore the growing competencies of cybercriminals in using AI for their own purposes [1]. Machine learning algorithms are increasingly used not only in defensive but also in offensive actions – from automating phishing, through creating convincing deepfakes, to conducting APT (Advanced Persistent Threat) attacks with the support of generative models [2], [3].

On the other hand, the growing demand for AI-based security solutions causes companies and research institutions to compete in developing new techniques for protecting IT infrastructure. A special place is occupied here by classification solutions based on the analysis of network traffic anomalies, intrusion detection systems (IDS), or adaptive firewalls supported by deep learning algorithms [4], [5]. Examples of such tools can be modern EDR (Endpoint Detection and Response) systems, which use artificial intelligence to identify and neutralize threats in real time [6]. Although some of these solutions are commercial and have limited access to implementation details, their importance grows every year.

At the same time, threats related to the security of AI/ML models themselves cannot be overlooked – including adversarial attacks, model stealing, data poisoning, or inference attacks [7]–[9]. Thus, two parallel directions of research appear: using AI for protection and protecting AI as a strategic technology.

This article presents a literature review and the results of experimental research using AI/ML techniques in the area of threat detection. The main activities concerned data analysis and the construction of ML models, the aim of which was to indicate efficient methods for classifying suspicious Windows user behaviors and network traffic anomalies. Additionally, the possibilities of applying generative artificial intelligence, both as a threat and a potential tool supporting incident analysis, were discussed.

II. Literature Review

The contemporary scientific discussion on automation and attack detection is often based on searching for the best and most efficient methods and tailoring them to specific types of threats [10]. In many publications, researchers emphasize the growing importance of cloud computing aspects and the necessity of examining the security of cloud environments. In the review "A comprehensive review of AI based intrusion detection system", the authors focus on the division of detection systems (Intrusion Detection System) into network-based, host-based, and hybrid (network-based, host-based, hypervisor-based, distributed-based). Similar observations are also presented by the analysis published by GeeksforGeeks [4], where the role of artificial intelligence in increasing the effectiveness of modern security systems was emphasized – in particular due to AI's ability to process large datasets in real time, detect behavioral anomalies, and automate incident response. This article also highlights the growing importance of the predictive nature of AI tools in countering threats such as ransomware, phishing, or malware, which reinforces the necessity for further research on their effectiveness and adaptability in multi-cloud environments.

The discussed topic is important because the discussion redirects the attention of security specialists to another protection vector, while indicating the limitations of frameworks dealing with anomaly detection and relying on traditional firewalls in networks based on multi-cloud solutions [10].

Another important thread raised in the publication of researchers from the Indian Christ University is the application of artificial intelligence techniques – in particular machine learning (ML), deep learning (DL), and ensemble learning methods – for intrusion detection and attack classification. The authors indicate that AI-based methods offer greater accuracy and effectiveness compared to traditional approaches, but most research focuses on general attack detection, ignoring the aspect of accurately distinguishing individual types of threats. Attention is also drawn to challenges related to multi-class attack classification and the need for further development of methods enabling automatic learning of features from large datasets, which is particularly important in the context of the constantly growing amount of data in the cloud.

In a similar context, practical applications of AI in threat detection systems are presented in a summary published by

Analytics Insight, where selected commercial solutions such as Darktrace or CrowdStrike Falcon were described. These systems use machine learning algorithms and behavioral analysis to identify anomalies and respond to attacks in real time, also in multi-cloud environments. This shows that the direction set by academic research is already reflected in industrial implementations, which reinforces the importance of further research on the effectiveness and precision of AI in the context of network security [11].

Similar conclusions are also presented by Vidura Wijekoon [12], indicating that AI significantly increases the effectiveness of IDS systems – among others by reducing false alarms and better detection of zero-day attacks. The author also emphasizes the role of algorithms such as Random Forest or SVM in identifying anomalies, which confirms the effectiveness of approaches based on machine learning in dynamic network and cloud environments.

The analysis published on the GeeksforGeeks platform [4], in turn, indicates that the application of artificial intelligence in cybersecurity enables the analysis of huge amounts of data in real time, which allows for more effective detection of malware, ransomware attacks, and phishing. The importance of automating incident response and predicting potential threats based on behavioral patterns was also emphasized. This points to the increasing maturity and practical importance of AI tools not only in academic environments but also in real system applications.

III. Implementation of the Solution on the CIC-IDS2017 Dataset

This chapter describes the details of the implementation of a system detecting threats in network traffic using machine learning methods. This system includes four main components: a data preparation module, a training module, an evaluation layer, and a module responsible for network traffic analysis in real time.

A. Dataset

The CIC-IDS2017 (Canadian Institute for Cybersecurity – Intrusion Detection System 2017) dataset constitutes a complex set of data serving to study the issue of intrusion detection in computer networks [13]. It includes both normal traffic (benign) and data concerning various attacks, such as brute force, DoS/DDoS, web attacks (including XSS, SQL injection), port scanning, as well as the activities of advanced APT threats. The data was collected in a controlled laboratory environment over five days and contains flow-based features generated using tools such as CICFlowMeter.

Thanks to labeling samples and maintaining diverse attack scenarios, the dataset enables the application of both supervised and unsupervised methods in anomaly analysis. CIC-IDS2017 is considered one of the more representative and realistic data sources for testing threat detection algorithms in IDS systems.

B. Data Preprocessing

The first step in the system implementation was the preparation of input data for training machine learning models. The data was cleaned by removing irrelevant columns and records with missing values. Then, categorical variables (such as network protocol) were encoded into numerical form, which enabled their use in classification models.

In order to ensure a uniform scale of values, all features were scaled using standardization. A key stage was also balancing the training dataset using the SMOTE technique, which limited the impact of dominant classes on the model result. SMOTE (Synthetic Minority Over-sampling Technique) involves artificially adding new examples to less numerous classes so that the dataset contains a similar number of samples for each category. New data is created through interpolation — that is, generating examples based on existing samples of the minority class and their nearest neighbors in the feature space.

Finally, the data was divided into training and validation sets, and test data was prepared separately, which was not used during model training.

C. Training Detection Models

In order to detect anomalies and potential threats in the analyzed network traffic, two distinct modeling approaches were applied: Random Forest and a Deep Neural Network (DNN). Both models were selected due to their documented effectiveness in classification tasks, but they represent different classes of algorithms: ensemble methods and deep learning.

Random Forest is a classification algorithm using a set of independent decision trees, whose final decision is based on a majority voting mechanism. In this work, a random forest consisting of 100 trees was constructed, each trained on a random sample of input data (with replacement, i.e., so-called bootstrap sampling). Additionally, to reduce correlation between trees, a random subset of features was considered at each decision node. This model exhibits high resistance to overfitting, making it a suitable choice for datasets containing noise or features of low importance. An additional advantage is the possibility of interpreting results by analyzing the importance of individual input features. Random Forest does not require prior data scaling or their linear separation, making it a versatile tool in the classification of tabular input data [14].

The second approach applied was a deep neural network (DNN) built from several fully connected layers (dense). The network consisted of an input layer adapted to 42 input features, two hidden layers containing 128 and 64 neurons respectively with the ReLU activation function, and an output layer with a single neuron and a sigmoid activation function (for binary classification). The ReLU (*Rectified Linear Unit*) function introduces non-linearity, passing positive values without changes while cutting negative values to zero, which favors faster learning of deep models. The sigmoid function transforms the output result to the range from 0 to 1, thanks to which it can be interpreted as the probability of belonging to the positive class. To limit overfitting, a *Dropout* mechanism was applied (with a probability of 0.3–0.5), which randomly

turns off a portion of neurons during learning, forcing the model to generalize. Learning took place using the *Adam* optimization algorithm, which combines the advantages of adaptive methods and momentum, automatically adjusting the learning rate for each weight in the model. The *binary_crossentropy* loss function was applied, measuring the difference between the predicted probability and the actual class label in binary tasks. Furthermore, the *Early Stopping* technique was used, which interrupts training when the model quality on the validation set stops improving, preventing its overfitting [15].

The advantage of the applied neural network is its ability to model complex, non-linear dependencies in data, which can lead to higher classification effectiveness. However, due to greater computational requirements and interpretability difficulties, this model requires careful calibration of hyperparameters and verification of operational effectiveness.

D. Evaluation and Analysis of Results

The evaluation of model effectiveness was conducted on both the validation and test sets. For each model, standard classification metrics were calculated: accuracy (the overall percentage of correct predictions), precision (how many reported alarms were actual threats, i.e., a low number of false alarms), recall (sensitivity – how many actual threats were detected, i.e., a low number of missed attacks), and the F1-score (a compromise between precision and recall). Furthermore, for the neural network, graphs were generated showing the course of the learning process (fig. 1) – accuracy and the model's loss function.

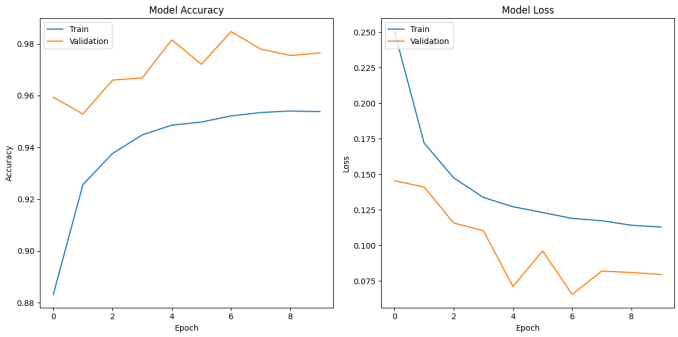

Fig. 1: Neural network learning process

Below, the results obtained by both models in binary threat classification mode are presented.

TABLE I: Results on the validation set (binary classification)

Model	Accuracy	Precision	Recall	F1-score
Random Forest	0.98	0.98	0.98	0.98
Neural Network	0.45	0.20	0.45	0.28

TABLE II: Results on the test set (binary classification)

Model	Accuracy	Precision	Recall	F1-score
Random Forest	0.90	0.91	0.90	0.90
Neural Network	0.32	0.10	0.32	0.15

The results unequivocally indicate the higher effectiveness of the Random Forest classifier compared to the neural network. The decision tree model achieved not only high accuracy but also a good balance between precision and sensitivity. In the case of the neural network, significant problems with attack class classification were noticed, which may indicate the necessity for better hyperparameter tuning and extended training.

A confusion matrix was also created for both the neural network and the Random Forest model to better understand the decisions made by the models during their operation. Both matrices were created based on the models' work on the test dataset. In fig. 2, it can be seen that the model completely fails to detect threats; all test input data representing malicious traffic were classified as non-threatening traffic. This testifies to a large problem in the dataset or improper training tuning of the model using hyperparameters.

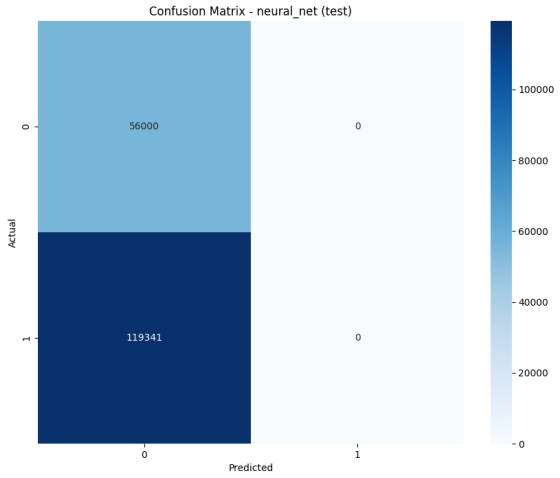

Fig. 2: Confusion matrix of the neural network-based model

Meanwhile, fig. 3 presents the confusion matrix for the Random Forest model. It can be seen that the predictions made by the model were overwhelmingly correct. As presented in table III, the Random Forest model achieves high metric values (approx. 90%) and could be used, for example, for early alerting about suspicious traffic in the network.

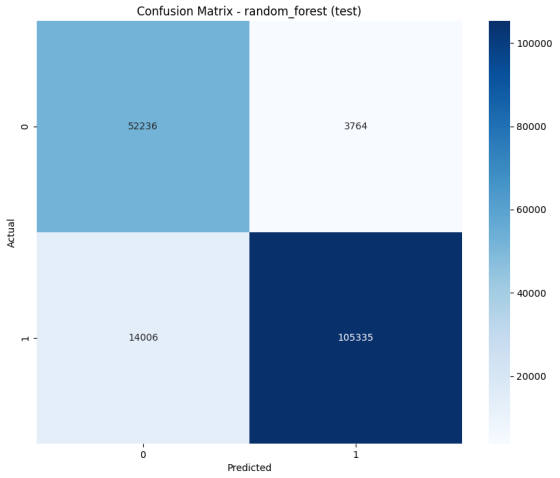

Fig. 3: Confusion matrix for the Random Forest model

E. Real-Time Threat Detection

To ensure practical application of the system, a monitoring module operating in real time was also developed. This component analyzed network traffic on a given interface, selected appropriate features from packets, processed them according to the training method, and then classified them as a threat or normal traffic.

Each detected incident was saved to a file in the form of a report containing IP addresses, protocol, packet size, classification label, and prediction confidence value. The collected data was used to create threat visualizations, including threat distributions by protocol and activity graphs over time.

Thanks to the application of models previously saved after training, real-time detection did not require retraining and could be run multiple times in test environments. The operation of this module is presented in fig. 4.

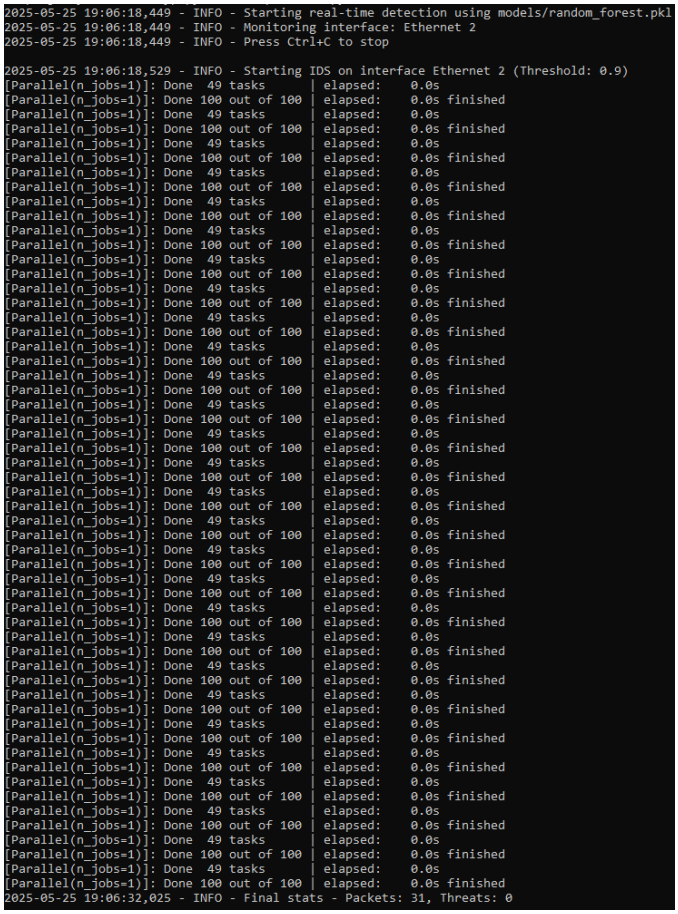

Fig. 4: Operation of the module checking model behavior in real time

F. Summary

The designed system constitutes a complete solution for the analysis and detection of threats in computer networks. Thanks to the modular structure, it was possible to quickly experiment with different approaches and scalably deploy both offline (training and evaluation) and online (real-time monitoring). The application of the CIC-IDS2017 dataset and replicable pipelines allows for a reliable assessment of model effectiveness and further development towards real applications in production environments.

IV. Experiment on the *Unraveled: A Semi-Synthetic Dataset for APT* Dataset

A. Introduction

The Unraveled dataset, and in particular its part located in the data/host-logs/windows directory, was created with the analysis of host-level activity in the context of Advanced Persistent Threats (APT) in mind [16]. These data include detailed system logs from Windows systems, containing information about launched processes, commands, and authentication events. Thanks to appropriately labeled records, they enable the study of user behavior patterns and correlating them with the actions of attackers of varying levels of advancement, including APT groups. These logs constitute research material

for assessing anomaly detection methods and machine learning models for early detection of internal and external threats in corporate environments (IDS type systems).

The first step was the normalization of datasets and aggregation of logs from different Windows stations.

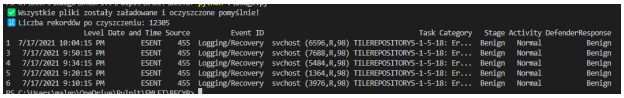

Fig. 5: Dataset after normalization

After loading and cleaning the data, the script proceeds to the stage of feature engineering and training four different anomaly detection models: Isolation Forest, MLPClassifier, Autoencoder, and a binary classifier in Keras architecture. The data is processed using numerical and categorical transformations, which enables their effective use in machine learning models.

The input dataset included over half a million records, which after filtering and cleaning was reduced to approx. 8600 examples. Despite a significantly unbalanced dataset (only single anomalies), the models achieved very high accuracy — particularly MLP and the binary classifier, which achieved 100 percent effectiveness on the test set.

The Autoencoder, in turn, demonstrated the ability to detect anomalies using reconstruction error (MSE), effectively identifying an outlier case despite very low precision. These results suggest that classical classifiers fit well to balanced data, while unsupervised models, like Isolation Forest and Autoencoder, are more resistant to the lack of labels, but their effectiveness may be limited by the low representation of anomaly classes.

The collected data after preliminary cleaning contained 8621 records, of which only 1 case was marked as an anomaly, which significantly influenced the assessment of the effectiveness of individual models. For example, the MLP classifier obtained a perfect result — 100 percent precision, recall, and f1-score — however, it should be noted that with such an extremely unbalanced set (1:1724), even a minimal number of errors can drastically lower the assessment of other models.

TABLE III: Results on the test set (binary classification)

Model	Accuracy	Precision	Recall	F1-score
Isolation Forest	0.88	0.00	0.00	0.00
MLP	1.00	1.00	1.00	1.00
Autoencoder	0.75	0.00	0.00	0.00
Keras Classifier	0.88	0.00	0.00	0.00

Isolation Forest and Autoencoder achieved high effectiveness in detecting normal cases (96–95 percent recall), but had difficulties correctly recognizing a single anomaly (f1-score for class 1 was 0.0 and 0.02 respectively). It is also worth noting that the Autoencoder, despite a very low reconstruction error (MSE 0.0003), classified the anomaly correctly — which indicates that it may be more sensitive to subtle deviations, even in the absence of a clear representation of the rare class in the training data.

Subsequently, the anomaly detection threshold was increased (especially in Isolation Forest, where parameters can be selected and the direct impact on results can be quickly seen) and it proved necessary to radically balance the data, but these actions did not bring measurable effects. In our opinion, the dataset requires structural unification and better differentiation of normal data from anomalies, because with such small differences, the calibration of detection thresholds (especially for methods based on reconstruction) is burdened with a high measurement error.

Like Isolation Forest, the Autoencoder did not detect the anomaly — the f1-score for class 1 was 0.00, and the reconstruction MSE (0.0796) suggests that the errors were not high enough to cross the decision threshold. However, the test and validation set extracted at the data processing stage was too small to enable a reliable assessment of the model's effectiveness. As a result, metrics such as accuracy can be misleading, and other results — unstable and strongly dependent on the specific data split. The results should therefore be treated as experimental and requiring verification on a larger, more balanced test set.

V. Conclusions and Future Directions

In the face of the growing complexity of cyberattacks and the increasingly frequent use of new technologies by cybercriminals, the development of effective defense methods becomes a priority. Artificial intelligence (AI) and machine learning (ML) offer huge potential in this field, but their implementation involves a number of challenges.

A. Preparation of Training Data

One of the key problems in applying AI to threat detection is the unbalanced nature of data. Typical datasets are dominated by normal network traffic, and suspicious activities constitute a small percentage of them. To counteract this phenomenon, techniques such as SMOTE are used, consisting of generating additional examples of rare events. An alternative is the ADASYN method, which focuses on creating data for harder-to-classify cases. Generative adversarial networks (GANs), enabling the creation of realistic attack scenarios for the needs of training AI models, are also arousing increasing interest.

B. Performance and Quality of Algorithms

Deep learning (DL), as an advanced form of AI, requires significant computational resources. To improve the operation of threat detection systems, specialized architectures are being developed, such as LSTM networks (time series analysis) or graph convolutional networks (GCN), which examine relationships between devices in a network. The transfer learning technique, consisting of transferring knowledge from one area to another, allows existing models (e.g., from the area of image recognition) to be used in the context of cybersecurity.

C. Integration with Security Tools

The effectiveness of AI depends not only on the quality of algorithms but also on the possibility of their integration with existing protection systems. Platforms such as SOAR (Security Orchestration, Automation and Response) allow for

the automation of incident response, e.g., blocking dangerous IP addresses. AI models can also cooperate within hybrid systems, combining simple algorithms (e.g., Random Forest) with advanced models (e.g., autoencoders), enabling the detection of previously unknown threats.

D. Generative AI: Opportunities and Threats

Generative AI models, capable of creating text, code, or images, open new possibilities but also create new risks. Tools such as ChatGPT can support analysts in interpreting login data, but at the same time can be used by criminals to create credible phishing messages. Researchers are experimenting with using GANs to simulate attacks, which allows for better preparation of defense systems for real threats.

E. Security and Ethical Aspects

The application of AI in cybersecurity entails the necessity of protecting the models themselves against attacks. An example is adversarial attacks, consisting of intentional modification of input data to confuse the system. There is also a risk of model theft, which could be used against their creators. The solution is model encryption techniques and the development of explainable AI (XAI), which allows for better understanding of decisions made by neural networks. Ethical issues are equally important, including those concerning employee privacy and responsibility for automatic decisions made by systems.

F. Verification in Practical Applications

Although many AI models show high effectiveness in test environments, their efficiency in real IT systems can be limited. Key significance has testing solutions in real corporate conditions and cooperation with institutions such as MITRE, which develops the ATT&CK framework. It allows for the classification of attack tactics and techniques, supporting the adaptation of AI models to current threats.

G. Conclusions

The development of AI in cybersecurity requires the cooperation of specialists from various fields. Engineers, security experts, lawyers, and ethicists should cooperate in creating systems that will be effective, safe, and consistent with social values. Only an integrated approach will ensure the appropriate use of the potential of artificial intelligence in defense against digital threats.

REFERENCES

[1] C. Advisory, "Artificial intelligence fuels new wave of complex cyber attacks challenging defenders," https://cybersecuritynews.com/artificial-intelligence-in-cyber-attacks/#:~:text=While%20AI-driven%20threat%20detection%20systems%20have%20advanced%2C%20cybercriminals,traditional%20defenses%2C%20creating%20a%20high-stakes%20technological%20arms%20race., 2025, dostęp: 2025-05-20.

[2] T. B. Brown and et al., "Language models are few-shot learners," *NeurIPS*, 2020. [Online]. Available: https://arxiv.org/abs/2005.14165

[3] M. Brundage and et al., "The malicious use of artificial intelligence: Forecasting, prevention, and mitigation," 2018. [Online]. Available: https://arxiv.org/abs/1802.07228

[4] GeeksforGeeks, "Ai in cybersecurity," https://www.geeksforgeeks.org/ai-in-cybersecurity/, 2025, dostęp: 2025-05-20.

[5] Avsistema, "https://avsistema.com/best-ai-powered-firewalls-for-enterprise-security-in-2025/," https://avsistema.com/best-ai-powered-firewalls-for-enterprise-security-in-2025/, 2025, dostęp: 2025-05-20.

[6] M. Corporation, "Endpoint detection and response framework: Mitre attck," 2021. [Online]. Available: https://attack.mitre.org

[7] B. Biggio and et al., "Evasion attacks against machine learning at test time," *ECML PKDD*, 2013.

[8] F. Tramèr and et al., "Stealing machine learning models via prediction apis," *USENIX Security*, 2016.

[9] M. Jagielski and et al., "Manipulating machine learning: Poisoning attacks and countermeasures for regression learning," *IEEE SP*, 2018.

[10] T. Sowmya and E. Mary Anita, "A comprehensive review of ai based intrusion detection system," *Measurement: Sensors*, vol. 28, p. 100827, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2665917423001630

[11] M. Prabhakar, "Top 10 ai-based threat detection systems," 2024, dostęp: 2025-05-19. [Online]. Available: https://www.analyticsinsight.net/artificial-intelligence/top-10-ai-based-threat-detection-systems

[12] V. Wijekoon, "Ai in intrusion detection systems (ids): A game-changer in network security," https://medium.com/@ViduraAI/ai-in-intrusion-detection-systems-ids-a-game-changer-in-network-security-1e514c5753 2024, dostęp: 2025-05-20.

[13] C. H. N, "Network intrusion dataset(cic-ids-2017)," 2023, dostęp: 2025-05-17. [Online]. Available: https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset?fbclid=IwY2xjawKVV5hleHRuA2FlbQIxMABicmlkETBPcWREVm5QVmplOXpxbnNkAR7fzCWBmEKzSx2482VMOYOI44Pw5OtpO2JoUjJVZg_aem_KE4WRmGcNJnu8gPbHCe5jg

[14] IBM, "What is random forest?" https://www.ibm.com/think/topics/random-forest, 2025, dostęp: 2025-05-10.

[15] ——, "What is a neural network?" https://www.ibm.com/think/topics/neural-networks, 2021, dostęp: 2025-05-10.

[16] S. Myneni, "Unraveled- a semi-synthetic dataset for advanced persistent threats," https://gitlab.com/asu22/unraveledfbclid=IwY2xjawKVV8xleHRuA2FlbQIxMABicmlkETBPcWREVm5QVmplOXpxbnNkAR5aem_F0_cqkXjW3kK3fMrjQw_KQ, 2024, dostęp: 2025-05-17.