A Deep Learning-Based Dual-Model Framework for Real-Time Malware and Network Anomaly Detection with MITRE ATT&CK Integration

Migara H. M. S, Sandakelum M. D. B, Maduranga D. B. W. N,
Kumara D. D. K. C, Harinda Fernando, Kavinga Abeywardena
Department of Computer Systems Engineering, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

***Abstract*—The contemporary world of high connectivity in the digital realm has presented cybersecurity with more advanced threats, such as advanced malware and network attacks, which in most cases will not be detected using traditional detection tools. Static cybersecurity tools, which are traditional, often fail to deal with dynamic and hitherto unseen attacks, including signature-based antivirus systems and rule-based intrusion detection. To address this issue, we would suggest a two-part, AI-powered solution to cybersecurity which would allow real-time threat detection on an endpoint and a network level. The first element uses a Feedforward Neural Network (FNN) to categorize Windows Portable Executable (PE) files, whether they are benign or malicious, by using structured static features. The second component improves network anomaly detection with a deep learning model that is augmented by Generative Adversarial Networks (GAN) and effectively addresses the data imbalance issue and sensitivity to rare cyber-attacks. To enhance its performance further, the system is integrated with the MITRE ATT&CK adversarial tactics and techniques, which correlate real-time detection results with adversarial tactics and techniques, thus offering actionable context to incident response teams. Tests based on open-source datasets provided accuracies of 98.0 per cent of malware detection and 96.2 per cent of network anomaly detection. Data augmentation using GAN was very effective in improving the detection of less popular attacks, including SQL injections and internal reconnaissance. Moreover, the system is horizontally scalable and responsive in real-time due to Docker-based deployment. The suggested framework is an effective, explainable and scalable cybersecurity defense system, which is perfectly applicable to Managed Security Service Providers (MSSPs) and Security Operations Centers (SOCs), greatly increasing the precision rate and contextual insight of threat detection.**

Keywords*—Cybersecurity; malware detection; generative adversarial networks; deep learning; MITRE ATT&CK; feedforward neural network

I. Introduction

The breath-taking development of cloud computing, the mobile platform, and the Internet of Things (IoT) has given rise to a corresponding enhancement in complication and exposure of enterprise networks. Threat actors have also gone to best modern practices of polymorphic malware, fileless attacks, and exfiltration of encrypted data, all of which bypass traditional cybersecurity controls easily, as digital infrastructures become more modernized [1]. A traditional detection mechanisms like signature-based antivirus software and rule-based intrusion detection systems rely on old-fashioned rules and known signatures, which makes them very inefficient in case of dynamically new types of threats. The need to have cybersecurity solutions that are smarter and adaptive to identify endpoint malware and network anomalies in real-time is therefore noted as being urgent. The current research focuses on the following research question: "How can the deep learning-based approaches, once combined with standardized threat frameworks, enhance the accuracy of real-time detection and context awareness in endpoint malware classification and network anomaly detection?" In answering this question, we suggest a two-model-based cybersecurity technology that leverages intensity in deep learning technology, namely Feedforward Neural Network (FNN) and Generative Adversarial Networks (GAN), combined with the MITRE ATT&CK framework to contextualize and correlate the detection of threats in real-time. The end goal of such a solution is not only to achieve a high rate of malware detection, network inspection, but also to enable actionable intelligence to Managed Security Service Providers (MSSPs) and Security Operations Centers (SOCs).

The rest of this study is divided into the following sections:

The related work (Section II) is about existing literature and the gaps in research on malware detection, network anomaly detection, and integration with MITRE ATT&CK. Section III (Methodology) outlines the structure of the proposed system, pre-processing methods of data, and deep learning models that will be employed during the study. Section IV presents the system architecture. Section V (Experimental Setup) outlines datasets, computing resources, and performance measures that were used to analyze the suggested framework. Section VI (Results and Discussion) shows the results of the experiment, analyzes the work of the model, and comments on the practical significance of the integration of the MITRE ATT&CK framework. Section VII details the conclusion and future work of the study.

II. Related Work

Cybersecurity has not remained as it was historically built on simple methods (based on static rules), to what we nowadays know to be advanced models which utilize artificial intelligence (AI) and deep learning (DL) [2]. Security products resting on traditional methods have been found inadequate as threats increasingly become more evasive and sophisticated. In this

section, a description of the development of methods in five broad subject areas within this research study is provided.

A. Static Malware Detection Techniques

Static analysis consists of analyzing the structure of the file without running the file. Initial models relied on features like the size of the files, section headers, and import tables to characterize Windows Portable Executable (PE) files [3]. These models were beneficial at the beginning, but they failed to compete with polymorphic malware, which alters the code structure to avoid detection. The form of deep learning has expanded the horizons of possibilities, which include the ability of Feedforward Neural Networks (FNNs) to learn complicated patterns from static file features. Obfuscation resistance is greater in these models, and they lead to much higher classification accuracy. Nevertheless, they are black boxes and cannot be easily interpreted or rather provide only binary classification.

B. Dynamic and Behavioral Malware Analysis

Dynamic detection monitors the behavior of files at execution time, heat-tracing system calls, use of the registry and communication between processes. Such behavior-based monitoring can reveal stealthy or clone malware, which can be missed by static analysis. Such methods are beneficial, but they are expensive and can be bypassed by malware that will determine whether it is sandboxed or not. Besides, clean, labelled execution logs and controlled test environments are also required to work real-time behavioral models, and this is not present in production networks [4].

C. Network Anomaly Detection and Class Imbalance

Network anomaly detection. This can detect anomalous patterns of data flow, e.g. unexpected usage of protocols or increased traffic. Time standard statistical models and clustering methods produce false alarms and do not have enough real-time sensitivity [5].

Deep classifiers are more accurate and suffer from class imbalance problems. In real networks, malicious traffic is vastly overshadowed by normal traffic, and as such, models may find it hard to learn rare attack patterns. This issue of class imbalance causes models to be hypersensitive, and as a result, allows low-frequency types of attacks to go undetected (False Negatives).

D. Generative Models for Intrusion Detection

In the field of intrusion detection, Generative Adversarial Networks (GANs) have recently been applied to correct data imbalance. GANs can help classifiers better detect underrepresented attack categories by generating synthetic attack samples that are more similar to legitimate attacks, therefore making it easier to detect. The result of this is better generalization and memory of low-profile intrusions, including web-based attacks or internal port scanning. Although useful, GAN-based models are found in only a few instances to be applied to a real-time detection setting; instead, models are employed in an offline manner to train on a dataset, but are not specifically used in the detection process itself [6].

E. Explainability and Threat Attribution

Due to the emergence of smarter detection models, explainability is vital. Unnecessarily, black-box AI has high levels of accuracy, which result in opaque alerts, culminating in the development of analyst fatigue and mistrust. Explainable AI (XAI) methods have also entered the mainstream, as attempts to understand which models chose their conclusions, as well as more empower analysts to characterize and subordinate alerts with higher confidence. At the same time, threat models, like MITRE ATT&CK, have disrupted threat attribution. In contrast to the so-called static vulnerability repositories, ATT&CK describes adversarial activities as structured tactics and techniques. Even then, however, most of the detection controls continue to apply the framework in the rule-based post-analysis or reporting as opposed to hard-coding it into the consumer-based detection ruleset, which should enhance real-time decision making [7].

F. Gap Summary

Deep learning has enhanced both static and behavioral detection [8], yet the work done in these fields is a decontextualised map of its own. GANs have improved the detection practice of rare attacks; however, they cannot be used in scalable and explainable systems. Moreover, not many solutions integrate actively into the real-time methodologies of MITRE ATT&CK along the way of converting mere alerts into intelligence that can be acted upon. This study proposes to resolve them by implementing a dual-model paradigm which merges each of the detection modes with real-time adversarial mapping of behavior to enhance analyst perception and response.

III. METHODOLOGY

The research proposed will have a two-model threat detection system that is able to identify both static malware and dynamic network anomalies. The system will be composed of two independent AI processes, one to identify malicious files with the use of a Feedforward Neural Network (FNN), and one to identify suspicious network traffic with the use of a Generative Adversarial Network (GAN) coupled with a deep classifier. The results of the pipelines are sent into a centralized layer of threat intelligence, with results then being remapped to MITRE ATT&CK techniques to be acted upon [9].

A. Malware Detection Pipeline

The process of malware detection uses an FNN model with supervised learning based on a labelled dataset of Windows PE (Portable Executable) files. Every file is described through a 45-dimensional feature vector consisting of only static attributes, including header data, section statistics, the names of the imported DLLs, the number of API calls, and the options used during the compilation. The parameters are transferred to a dropout-regularised multi-dense network over ReLU activation. The completed output is divided into 6 classes with the help of the SoftMax function: benign, trojan, worm, spyware, adware, and ransomware. The model is trained by early stopping, along with batch optimization via the Adam algorithm, to avoid overfitting and increase generalization. The classification of files occurs in real time during transfer, execution or download inside the system.

When detected, all the malicious files are converted into a MITRE ATT&CK technique using an internal mapping database. In particular, the executables marked with signs of execution of scripts are mapped to the tactics based on user interaction, i.e. T1204 (User Execution) [10]. In the same way, imported functions connected to the process injection are traced to T1055, and those representing the functionalities of remote access are all traced to T1133 (External Remote Services). This mapping database was created manually using the observed malware behavior observations and official information from MITRE so that it could maintain the interpretability and traceability.

B. Anomaly Detection Pipeline

The second module is a method that can detect anomalous network traffic on a hybrid basis, incorporating the use of GAN-generated synthetic data and a deep feedforward classifier. Training of pipeline utilizes CICIDS2017 dataset with labelled samples of different enterprise attacks, including brute-force login attacks, attempts to port scan, and web-based intrusion.

Pre-processing is done using cleaning, normalization, and label encoding on raw streams of traffic. The GAN module ensures the improvement of the dataset when representing fewer common attacks that do not contain enough samples in the training set. The learning process teaches the generator network how to generate realistic variations, and the discriminator assists it with this by attempting to identify real and fabricated examples [11]. This then sends the final updated dataset through the ultimate deep classifier, which is to be optimized by dropout and sigmoid activation to produce binary classification results (benign or malicious).

The proposed GAN-based method also solves the issue of the number of classes represented, which often occurs with real-life data on traffic, as well as makes it more responsive to low-genre attacks.

C. Threat Mapping and Output Design

Common threats identified by the two modules are correlated to the associated MITRE ATT&CK with the help of a rule-based correlation engine. The engine studies traffic protocols, traffic flow patterns and traffic signatures to match the detections to known adversarial patterns. On the example, network probing patterns are paired with T1046 (Network Service Scanning), and volumetric denial-of-service attacks are associated with T1499 (Endpoint DoS) [12].

The results of all detections are unified into a real-time Security Operations Center (SOC) dashboard, where file or flow IDs, the type of threat detected, severity score, and technique that was mapped, and the possible remediation measures, such as isolating the file, logging or notifying the user, can be observed. This consolidated dashboard allows the correlation of many signals to continuing attack campaigns, which aids in the decision-making of the analysts [13].

IV. System Architecture

The offered system architecture is expected to deliver both intelligent and end-to-end malware threats reasoning against two major domains of cybersecurity: endpoint malware categorizing and network abnormality detection. It has a more constructive and distributable construction that can be used in Security Operations Centers (SOC) or Managed Security Service Providers (MSSP). All elements are implemented as a service and may be easily connected to SIEM or SOAR applications in place. All components are designed as services and can be directly integrated into existing SIEM or SOAR platforms. The architecture is extensible, cloud-native, and compatible with on-premises deployments [14].

A. Overview of Architecture

Its structure is based on two principal detection engines that run concurrently. The former is a supervised Feedforward Neural Network (FNN) that is used to identify file-based threats, and the latter is a GAN-augmented classifier that is used to identify network-based anomalies. The two engines work independently, and their results are consolidated to one common layer, which entails threat correlation and visualization of analysts.

The Unified Threat Intelligence Layer combines the output of the two engines and cross-correlates it with associated MITRE ATT&CK techniques. This layer gives the analysts an insight into the nature and level of threats. On the front-end, there is a Visualization and Alerting Dashboard that shows current alerts, severity, and recommendations to respond to them, including isolating hosts, forwarding logs, or administrative alerting [15].

The proposed framework in general is presented in Fig. 1, showing the dual-model framework including endpoint malware detection and network anomaly detection blocks, and the mapping of the MITRE ATT&CK and real-time alerting features.

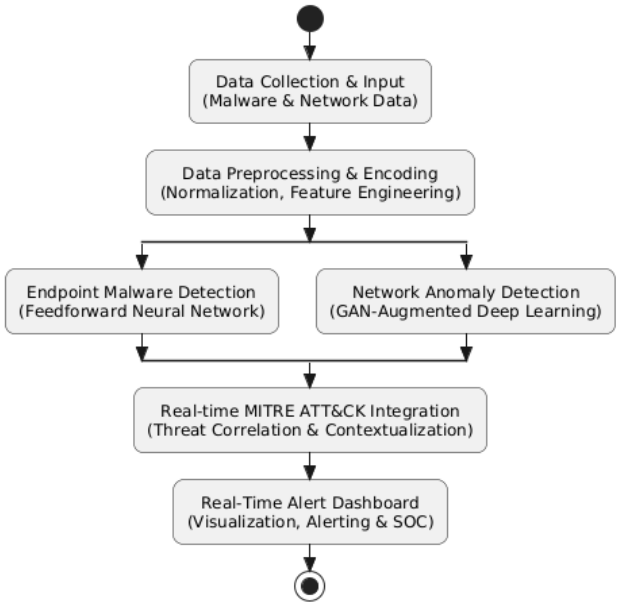

Fig. 1. High-level workflow of the proposed AI-based cybersecurity framework.

B. Malware Detection Engine

The malware detection engine can scan real-time PE (Portable Executable) files at the endpoints or network

gateways. A lightweight agent monitors all file downloads, executions, or transfers of files with a type of .exe, .dll, or .sys. After interception, the specific file undergoes a feature extraction procedure where it acquires the 45 most significant attributes of the PE header and import table, including section sizes, entropy values, time stamp data, and imported APIs [3].

These attributes are then entered into a pre-trained FNN model, which categorizes the file based on a small number of pre-determined categories, e.g., benign or malware families. When a file is determined to be malicious, it consults a curated mapping database of file behavior and file signatures to MITRE ATT&CK techniques. The detection output is recorded and sent to the intelligence layer, containing metadata such as file hash, threat label, and mapped techniques [16].

C. Anomaly Detection Engine

The network engine is implemented passively at strategic ingress and egress points where it keeps a check on the traffic flows via TAPs or mirror ports. The raw data is normalized and coded to transform it into structured flow vectors and processed by a GAN module that produces synthetic samples to enhance the class balance. These artificial samples are injected into live traffic streams and fed to a deep neural classifier, which detects anomalous traffic like port scans, protocol abuse and intrusion attempts. These flows are then correlated against the related adversarial techniques and sent to the same intelligence layer to have them alerted in aggregate [17].

D. Unified Threat Intelligence and Dashboard Layer

The MITRE Mapping Engine is a foundation of the mapping between model outputs and known adversarial behavior. Based on a set of pre-determined rules, the system associates the results of detection with ATT&CK techniques and methods like Initial Access, Execution, Lateral Movement, or Exfiltration. The Alert Correlator collects the results of both engines and prepares results to display in the analyst dashboard. Threat overviews, severity ratings, suggested responses, and filters are all given in the dashboard user interface, which supports filtering by timestamp, type of attack, or source/destination [18].

V. Experimental Setup

To validate the scalability, performance, and real-world applicability of the proposed dual-model cybersecurity framework, a series of controlled experiments was devised. The experiments replicate real-world enterprise-level network and endpoint setups, including data collection, training, testing, and performance benchmarking.

A. Environmental Configuration

The experimental setup included local and cloud-based computing facilities. FNN model training was conducted using a local PC with an Intel Core i7 (11th Gen) processor, 16 GB RAM, and an NVIDIA GeForce MX450 GPU (2 GB VRAM). For training GAN-based anomaly detection, Google Colab Pro+ was utilized with a virtual runtime environment supported by NVIDIA Tesla T4 and RTX 3080 GPUs with a maximum of 16 GB VRAM and 32 GB RAM.

Software configuration included Python 3.10 as the base programming environment, and TensorFlow 2.x for FNN modelling and PyTorch for GAN-based deep learning. Data preprocessing and testing used Scikit-learn, while MongoDB was used for alert logging, Flask for routing web services, and Docker for microservice containerization. This hybrid deployment strategy ensures flexibility and compatibility with modern SOC environments.

B. Dataset Description

This study was supported by two public datasets. The first dataset is applied in malware identification, and it was downloaded from Kaggle. It consists of 138,047 Windows PE (Portable Executable) files. The dataset consists of six categories, namely, benign, trojan, worm, ransomware, spyware, and adware. The files were processed in a static way to obtain a collection of 45 engineered features based on the header, section metadata, and import/export tables.

The second dataset, which was used in the detection of anomalies, was the CICIDS2017 dataset. It includes tagged network traffic of several days and different kinds of attacks, such as brute-force SSH, SQL injection, Heartbleed, port scanning, and intrusion. The traffic was transformed into vectors of structured flow with 78 features. These were coded and standardized according to the input scheme of the GAN-enhanced deep learning model [19].

VI. Results and Discussion

In this section, the accuracy, precision, recall, and F1-score are discussed as key performance indicators with the false positive rate to examine the performance and efficiency of the dual-model cybersecurity system. Also, the results of MITRE ATT&CK mapping and deployment factors, including scalability and latency, are discussed. The results show that the system is technically effective and can deliver contextual threat intelligence that is appropriate in an enterprise Security Operations Center (SOC) [20].

A. Malware Classification (FNN Results)

The FNN classifier model that was trained using PE header metadata was very successful in differentiating between malicious and benign executables. The model had a 98.0 level of accuracy with an F1-score of 98.6. It was also very successful in differentiating unfamiliar malware families, but solely mediocre in confusing trojans and spyware, probably because of similar static features. This was solved by dataset balancing during training [21].

Table I indicates the performance of the Feedforward Neural Network (FNN) model regarding the recognition of different types of malware. The model was accurate and generalized with several malware classes.

TABLE I. Malware Classification Performance (FNN Model)

Metric	Value
Accuracy	98.0%
Precision	95.2%
Recall	98.0%
F1-Score	98.6%
False Positives	1.7%

This superb accuracy in identifying benign files is particularly valuable in endpoint security, where false positives can interfere with critical user workflows. Performance was uniform across training and validation sets, further confirming the generalization capability of the model [21].

B. Anomaly Detection (GAN-DL Results)

The GAN-augmented anomaly detection pipeline was proven to be better at detecting heterogeneous network attacks, even the rare behaviors of attacks. The last classifier had an accuracy of 96.2%, an F1-score of 95.2%, and an AUC-ROC of 0.987, which means that it discriminated well between benign and malicious flows [22].

The good AUC-ROC gave rise to a good detection rate on different network intrusions as indicated in Table II.

TABLE II. ANOMALY DETECTION PERFORMANCE (GAN-DL MODEL)

Metric	Value
Accuracy	96.2%
Precision	94.0%
Recall	96.5%
F1-Score	95.2%
AUC-ROC	0.987

Data augmentation using GANs contributed greatly to the improvement of detection on rare threats. As an example, the recall of SQL Injection attacks increased by 30.9% to the levels of 87.1% after GAN augmentation. This is an advantage that can be seen in Table III, where there is a significant increase in recall of rare attack types such as SQL Injection and Heartbleed.

TABLE III. GAN EFFECT ON RECALL FOR RARE ATTACKS

Attack Type	Recall (Pre-GAN)	Recall (post-GAN)
SQL Injection	56.2%	87.1%
Web Infiltration	63.4%	90.2%
Heartbleed	59.7%	85.3%

C. MITRE ATT&CK Mapping and Interpretability

The two models generated threat outputs that were correlated to MITRE ATT&CK tactics by a rule engine developed in-house. This allowed analysts to think of alerts not as a type of malware or an anomaly signature but as an objective of the attacker and their techniques. To give an example, ransomware detection was assigned to T1486 (Data Encrypted for Impact), and port scanning to T1046 (Network Service Scanning). Such mappings provide additional depth to the operations environment, enabling SOC analysts to measure stages of an attack, normal activities, and likely attackers. The outputs of the example mapping are given in [23]:

- Lateral movement (GAN-DL) → T1021: Remote Services.
- Macro-execution (FNN) → T1203: Exploitation for Client Execution.

D. Comparative Evaluation

To contextualize the performance, the system was compared to baseline models, namely Random Forest, Support Vector Machine (SVM), and Autoencoder-based Intrusion Detection Systems (IDS). The models under consideration had higher accuracy and explainability [24].

E. Latency and Scalability

During deployment, the two models were tested on load with a realistic SOC. The FNN classifier required less than 0.025 seconds on CPU to classify a single file, whereas the GAN-DL pipeline could process around 8,500 network flows every second, with a median and mean flow latency of 0.06 seconds and 0.07 seconds, respectively. These findings clearly show that the system can work in a high-throughput environment with little latency.

Horizontal scaling was also possible as distributed systems could host thousands of instances of detection due to the Docker-based microservice architecture [25].

The comparison of the suggested model and baseline approaches is presented in Table IV. These findings show that the dual-model framework is more accurate compared to Random Forest, SVM, and Autoencoder IDS, with the former having higher F1-scores and less false positive rate.

TABLE IV. COMPARATIVE MODEL EVALUATION

Model	Accuracy	F1-Score	False Positive Rate	AUC
FNN (Malware Detection)	98.0%	0.98	1.7%	N/A
GAN-DL (Anomaly, Ours)	96.2%	0.952	2.3%	0.987
Random Forest (NIDS)	96.1%	0.951	2.5%	0.88
SVM	88.5%	0.87	6.5%	0.79
Autoencoder IDS	91.4%	0.89	4.8%	0.89

VII. CONCLUSION AND FUTURE WORK

This study showed a two-model cybersecurity system whose features are to integrate deep learning and threat intelligence to combat network anomalies and file-based malware. A Feedforward Neural Network (FNN) is used to classify Static Portable Executable (PE) files, and a GAN-augmented deep learning classifier is used to detect anomalous activity on the network traffic. These two models are also aligned with the MITRE ATT&CK framework, where the findings of the threat detection process can be placed within the adversarial tactics and techniques. The interpretability enables the SOC analysts to make informed and timely decisions not just based on the alerts but also based on the overall behavioral intent of the alerts. The system was tested with reference to two established datasets. The FNN classifier received 98.0% accuracy results when detecting malware, whereas the anomaly detection system based on GAN received 96.2% accuracy results, showing a much better result on low occurrences of attack types with data augmentation. It is important to note that the system showed real-time capabilities with limited latency and container-based deployment (both locally and cloud) in

high-throughput setups. These characteristics enable it to fit well in the contemporary, scalable SOC designs.

Although the system has a good performance and architectural flexibility, it is now restricted to the analysis of static files and supervised anomaly detection of binaries. Moreover, the models are used as black boxes, so their transparency is minor in terms of the rationale of classification, and it leads to the problems of trust and human verification.

Several upgrades are foreseen in the future. One of the main directions is to include explainable AI techniques like SHAP or LIME to make the model predictions more interpretable and help the analyst better understand. This recent interest in static detection of malware can be applied in the detection of fileless malware with the addition of behavioral analysis of memory or runtime detection. Privately, the use of federated learning methods would allow training shared models without disclosure of sensitive raw data across organizations. Additional optimization to resource-limited settings, e.g., IoT or edge computing, has the potential to increase applicability through methods like model pruning, quantization, or TinyML adaptations. Lastly, the implementation of automated response capabilities in the form of SOAR platforms would give the ability to the ability to take action, such as host isolation, IP blocking, or sandboxing, based on detection confidence.

Overall, this two-model AI-powered framework fills a significant gap in modern cybersecurity by combining high detection capability with contextual threat insight. It demonstrates that integrating deep learning with formalized adversarial behaviour modelling offers an interpretable and scalable defence solution suitable for enterprise-level security. As cyber threats grow increasingly sophisticated, such frameworks will play a crucial role in supporting proactive, intelligent, and dynamic security operations adapted to the evolving threat landscape.

REFERENCES

[1] "A Deep Learning-Based Dual-Model Framework for Real-Time Malware and Network Anomaly Detection with MITRE ATT&CK Integration".

[2] R. Kaur, D. Gabrijelčič, and T. Klobučar, "Artificial intelligence for cybersecurity: Literature review and future research directions," Information Fusion, vol. 97, p. 101804, 2023, doi: https://doi.org/10.1016/j.inffus.2023.101804.

[3] A. Al-ojaimi, "Advanced Framework for Detecting Malware in Portable Executable (PE) Files Using a Multi-Model," Journal of Information Systems Engineering and Management, vol. 10, pp. 769–781, Apr. 2025, doi: 10.52783/jisem.v10i36s.6562.

[4] D. Vidyarthi, S. P. Choudhary, S. Rakshit, and C. R. S. Kumar, "Malware Detection by Static Checking and Dynamic Analysis of Executables," International Journal of Information Security and Privacy, vol. 11, pp. 29–41, Jul. 2017, doi: 10.4018/IJISP.2017070103.

[5] S. Author and S. Roy, A comprehensive Survey on Network Traffic Anomaly Detection using Deep Learning. 2024. doi: 10.13140/RG.2.2.32071.30884.

[6] M. Rahman, G. Iii, and H. Shahriar, "Leveraging GANs for Synthetic Data Generation to Improve Intrusion Detection Systems," Journal of Future Artificial Intelligence and Technologies, vol. 1, pp. 429–439, Feb. 2025, doi: 10.62411/faith.3048-3719-52.

[7] W. Olabiyi and S. Daniel, "Explainable AI for Fraud Detection - Techniques for Understanding and Interpreting Adaptive Fraud Detection Systems," Oct. 2023.

[8] S. Katragadda, K. Odubade, and E. Isabirye, "Anomaly Detection Detecting Unusual Behavior Using Machine Learning Algorithms to Identify Potential Security Threats or System Failures," International Research Journal of Modernization in Engineering Technology and Science, vol. 2, pp. 2582–5208, May 2020, doi: 10.56726/IRJMETS1335.

[9] D. Katiyar, M. Tripathi, M. Kumar, M. Verma, D. Sahu, and D. Saxena, "AI and Cyber-Security: Enhancing threat detection and response with machine learning.," Educational Administration Theory and Practices, vol. 30, Apr. 2024, doi: 10.53555/kuey.v30i4.2377.

[10] N. Tuan, H. Nguyen Hoang, and N. Thien, "Detecting Malware in Portable Executable Files using Machine Learning Approach," International Journal of Network Security & Its Applications, vol. 14, pp. 11–17, May 2022, doi: 10.5121/ijnsa.2022.14302.

[11] A. K. S. Ali, A. Raza, and H. Arif, "Intelligent Intrusion Detection and Data Protection in Information Security Using Artificial Intelligence and Machine Learning Techniques," vol. 3, p. 818, Apr. 2025, doi: 10.5281/zenodo.15281247.

[12] "MITRE ATT&CK®." Accessed: Jun. 17, 2025. [Online]. Available: https://attack.mitre.org/

[13] R. Kryukov, V. Zima, E. Doynikova, E. Novikova, and I. Kotenko, "Mapping the Security Events to the MITRE ATT &CK Attack Patterns to Forecast Attack Propagation (Extended Abstract)," 2022, pp. 165–176. doi: 10.1007/978-3-031-21311-3_10.

[14] M. J. Goswami, "AI-Based Anomaly Detection for Real-Time Cybersecurity," pp. 1075–3006, Feb. 2024.

[15] S. Sileola and A. James, "Neural Network-Based Anomaly Detection in Cybersecurity," Feb. 2025.

[16] R. Vyas, X. Luo, N. McFarland, and C. Justice, Investigation of malicious portable executable file detection on the network using supervised learning techniques. 2017. doi: 10.23919/INM.2017.7987416.

[17] "Understanding Network TAPs – The First Step to Visibility and Monitoring Span Online." Accessed: Jun. 17, 2025. [Online]. Available: https://www.gigamon.com/resources/resource-library/white-paper/understanding-network-taps-first-step-to-visibility.html

[18] B. Al-Sada, A. Sadighian, and G. Oligeri, "Analysis and Characterization of Cyber Threats Leveraging the MITRE ATT&CK Database," IEEE Access, vol. PP, p. 1, Jan. 2023, doi: 10.1109/ACCESS.2023.3344680.

[19] "Network Intrusion dataset(CIC-IDS- 2017) | Kaggle." Accessed: Jun. 17, 2025. [Online]. Available: https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset

[20] P. Rafiey and A. Namadchian, Mapping Vulnerability Description to MITRE ATT&CK Framework by LLM. 2024. doi: 10.21203/rs.3.rs-4341401/v1.

[21] C. Connors and D. Sarkar, Machine Learning for Detecting Malware in PE Files. 2023. doi: 10.1109/ICMLA58977.2023.00331.

[22] T. Kumarage, S. Ranathunga, C. Kuruppu, N. De Silva, and M. Ranawaka, Generative Adversarial Networks (GAN) based Anomaly Detection in Industrial Software Systems. 2019. doi: 10.1109/MERCon.2019.8818750.

[23] "User Execution: Malicious File, Sub-technique T1204.002 - Enterprise | MITRE ATT&CK®." Accessed: Jun. 17, 2025. [Online]. Available: https://attack.mitre.org/techniques/T1204/002/

[24] I. Elezmazy and W. Abdullah, "Advanced Intrusion Detection in Software-Defined Networks through Ensemble Modeling," Information Sciences with Applications, vol. 4, pp. 1–11, Oct. 2024, doi: 10.61356/j.iswa.2024.4389.

[25] F. Pacheco, E. Exposito, M. Gineste, C. Baudoin, and J. Aguilar, "Towards the Deployment of Machine Learning Solutions in Network Traffic Classification: A Systematic Survey," IEEE Communications Surveys & Tutorials, vol. PP, p. 1, Nov. 2018, doi: 10.1109/COMST.2018.2883147.