

AI-Powered Vulnerability Detection and Adaptive Defense Strategies in Cybersecurity

Şahin Kara^{1,*} , Fatih İlkbahar² , Muhammed Zekeriya Gündüz³ 

¹Sakarya University of Applied Sciences, Information Technologies Vocational School Department of Computer Technologies, Sakarya, Türkiye, ror.org/01shwhq58

²Duzce University, Akcakoca Vocational School Department of Computer Technologies, Düzce, Türkiye, ror.org/04175wc52

³Bingol University, Technical Sciences Vocational School Department of Computer Technologies, Bingöl, Türkiye, ror.org/03hx84x94

Corresponding author:

Şahin Kara, Sakarya University of Applied Sciences, Information Technologies Vocational School, Department of Computer Technologies
sahinkara@subu.edu.tr



Article History:

Received: 02.06.2025
Revised: 24.07.2025
Accepted: 05.07.2025
Published Online: 29.09.2025

ABSTRACT

Cybersecurity threats are becoming increasingly complex and sophisticated. These challenges highlight the growing need for organizations and individuals to safeguard their digital assets. In this context, artificial intelligence (AI) technologies offer substantial capabilities to detect and mitigate cybersecurity vulnerabilities. AI enables effective protection by performing deep analyses on large datasets to identify abnormal activities and predict potential threats. By transforming traditional security paradigms, AI contributes to faster and more adaptive responses against cyberattacks. Furthermore, AI's ability to classify threats and respond in real time gives security professionals a strategic edge. In the following sections, the role of AI in identifying and addressing cybersecurity vulnerabilities will be examined in detail, supported by current real-world applications. Finally, the paper will explore the future of AI in cybersecurity and potential directions for further enhancement.

Keywords: Cyber Security, Cyber-attack Experiments, Artificial Intelligence in Cybersecurity, Machine learning, Random Forest.

1. Introduction

The growing frequency and sophistication of cyberattacks pose serious threats to the security and stability of modern digital infrastructures. These attacks can compromise personal data, expose corporate trade secrets, and jeopardize national security. In recent years, the need to strengthen cybersecurity measures has become increasingly evident for individuals and organizations. Identifying and addressing cybersecurity vulnerabilities is crucial for ensuring robust digital defense mechanisms within this context [1].

Advanced technologies—notably Artificial Intelligence (AI)—fundamentally reshape cybersecurity strategies. AI is pivotal in automating defensive mechanisms by supporting threat intelligence, anomaly detection, secure code development, generating datasets, and optimizing incident response workflows. This automation not only improves operational efficiency but also enhances overall system resilience.

AI can significantly strengthen cyber defense operations by enhancing human decision-making with machine intelligence. Using machine learning algorithms to process vast datasets, AI systems can detect subtle anomalies associated with emerging threats, enabling rapid identification and response. Moreover, integrating AI into cybersecurity workflows reduces repetitive tasks, allowing professionals to focus on more complex analytical challenges. One of AI's most vital contributions is its predictive capability. By leveraging historical data and real-time monitoring, AI can forecast and neutralize potential threats before they fully materialize [2].

AI-driven platforms, with their ability to process large-scale data inputs, enable the early identification of irregular behaviors, allowing for effective threat anticipation. Deep learning techniques deliver greater precision and speed than traditional detection methods. These techniques are particularly valuable in countering advanced exploits—such as zero-day attacks—by offering proactive and adaptive defense mechanisms.

Beyond detection, AI also helps minimize the impact of security breaches and supports system recovery, forming a potential basis for future research. In vulnerability management, it reduces manual workload, reduces human error risk, and improves operational efficiency.

Despite these advantages, deploying AI-based cybersecurity systems presents several challenges. Ensuring algorithmic accuracy and reliability is critical to maintaining system effectiveness. Equally important is the quality and accessibility of the training data on which these systems depend. Furthermore, ethical concerns—including data privacy and user confidentiality—must be carefully addressed, as AI inherently relies on vast amounts of sensitive information that require robust protection.

The following sections will examine the role of AI in identifying and addressing cybersecurity vulnerabilities, supported by real-world use cases. The paper will conclude by evaluating the future trajectory of AI in cybersecurity and outlining potential directions for technological advancement in the field.

2. Related Works

In cybersecurity, the escalating complexity and diversification of threats have significantly intensified the challenges in detecting and remedying system vulnerabilities. The convergence of artificial intelligence (AI) with cybersecurity marks a critical junction where innovation meets exposure. AI has revolutionized numerous industries through its capability to analyze vast datasets, recognize patterns, and execute autonomous decisions, transforming multiple dimensions of contemporary life [3]. This transformative force increasingly influences the cybersecurity landscape, offering enhanced defense capabilities while concurrently introducing novel challenges.

Within this context, advanced technological tools such as artificial intelligence (AI) and machine learning (ML) have become indispensable for cybersecurity professionals [4]. By scrutinizing massive datasets, deep learning algorithms can effectively identify anomalous behaviors, enabling robust protection against known and previously unseen threats [5]. Notably, their adaptive defense mechanisms show promise in addressing zero-day vulnerabilities and fortifying security operations.

Ali S. et al. (2022) conducted a comprehensive review highlighting the strengths and limitations of prominent machine learning and deep learning algorithms in detecting zero-day threats. Their study compares various datasets, offering empirical insights into detection efficacy [6].

Zeadally et al. (2020) explored the potential of AI—particularly ML and DL techniques—in countering the rapidly evolving threat landscape. Their findings delineate both the promising prospects and the inherent weaknesses of AI systems in cybersecurity applications [7].

Tayyab M. et al. (2023) presented a systematic investigation into the security and privacy challenges surrounding deep learning algorithms. Their study elaborates on defense techniques against prevalent cyber-attacks and proposes cryptographic approaches to data confidentiality [8]. In another contribution, Al-Mansoori S. et al. discussed critical issues related to AI reliability, data privacy, and ethical considerations, thoroughly evaluating AI's role in cyber defense systems [9].

Akhtar, M. S., and Feng, T. (2021) examined the practical implementation of AI-driven techniques in cybersecurity analytics. Their research emphasizes the superior performance of AI over traditional methods in detecting novel and sophisticated attack vectors, underlining AI's capacity to process and interpret massive volumes of security data efficiently [10].

Sontan, A. D. and Samuel, S. V. (2024) extensively analyzed the intersection of AI and cybersecurity, mapping out both the emerging opportunities and associated challenges. Their work details AI's transformative influence in threat detection, vulnerability analysis, and incident response. Moreover, the study contrasts conventional vulnerability analysis methods with AI-powered techniques such as automated scanning, threat prioritization, and adaptive risk evaluation. The critical role of AI-driven automation in accelerating incident response, minimizing human error, and enhancing overall security posture is also emphasized [11].

The effectiveness of AI and ML in cybersecurity spans a wide range, from identifying threats to mitigating the impact of breaches. Mudassar et al. (2021) focused on the malicious use of AI technologies in cyberattacks. Their research discusses how AI can be exploited in various attack scenarios, explores potential countermeasures, and assesses AI's utility in optimizing security operations [12].

Nonetheless, integrating AI and ML-based security systems also brings certain challenges. These include potential false positives, data integrity concerns, and ethical considerations [13]. Developing advanced methodologies that ensure secure deployment and mitigate these complications is imperative.

In conclusion, the existing literature underscores the pivotal role that advanced AI and ML technologies can play in detecting and mitigating cybersecurity vulnerabilities. It further highlights the necessity for continuous refinement and innovation in these systems to keep pace with an ever-evolving threat landscape [14].

3. Cyber Vulnerability Analysis

Cybersecurity vulnerabilities can be defined as weaknesses or flaws within the security controls of a system or application. In other words, these vulnerabilities represent the weak points in the protective mechanisms of computer systems, networks, or software. Such vulnerabilities enable malicious actors to gain unauthorized access, exfiltrate data, or disrupt the system's normal functioning. Typically, these weaknesses arise due to coding errors leading to software bugs, misconfigurations such as firewall rule mistakes or default settings, poor password policies, design flaws and human errors related to improperly set user permissions [15].

Since vulnerabilities create entry points for cyberattacks, they pose a significant threat. Exploiting these weaknesses can lead to data breaches, financial losses, and damage to organizational reputation. Consequently, accurate detection and swift remediation of vulnerabilities are of critical importance. For instance, if an application's input validation is weak, attackers can easily infiltrate the system and perform unauthorized actions. Similarly, insufficient network security exposes network traffic to unauthorized interceptions, increasing the risk of data leaks. Such vulnerabilities commonly serve as foundational elements for cyberattacks and malware campaigns [16].

Over the last decade, cybersecurity vulnerabilities have emerged as a widespread and severe issue. Rapid technological advancements and the pervasive digitization of all sectors have escalated the risk faced by individuals' information systems and digital assets. These vulnerabilities can result in substantial financial damage, reputational harm, and even legal consequences, particularly for large enterprises and government agencies. A notable example is the 2017 Equifax data breach, which exposed personal information of millions and severely tarnished the company's reputation [17]. Similarly, the WannaCry ransomware attack caused significant operational disruptions and financial losses across numerous organizations and governments worldwide [18]. The prevalence of such vulnerabilities is attributed to the complexity of software development processes, the continually evolving threat landscape, and the human factor.

According to OWASP's 2017 report, the top three risks were the most common security vulnerabilities: injection flaws, broken authentication, and sensitive data exposure [19]. By 2021, the leading vulnerabilities shifted to broken access control, cryptographic failures, and injection attacks [20]. Figure 1 illustrates the changes in the OWASP Top 10 web application security risks between 2017 and 2021.

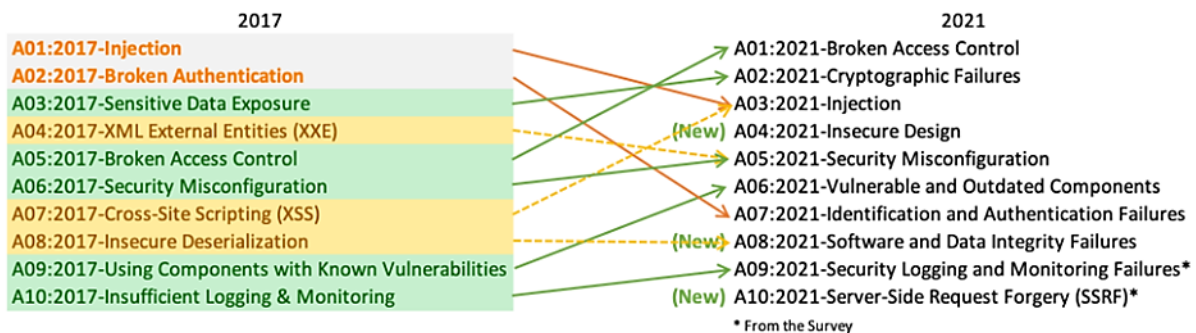


Figure 1. OWASP's Top 10 Security Vulnerability Categories in 2021.

As of 2023, the most prevalent security issues identified by OWASP emphasize broken object-level authorization, authentication failures, and sensitive data exposure [21]. These shifts underscore the evolving nature of cybersecurity threats and highlight new risks organizations must address. The vulnerabilities detailed below classify weaknesses at various security layers, with specific threats enumerated under each category, as depicted in Figure 2.

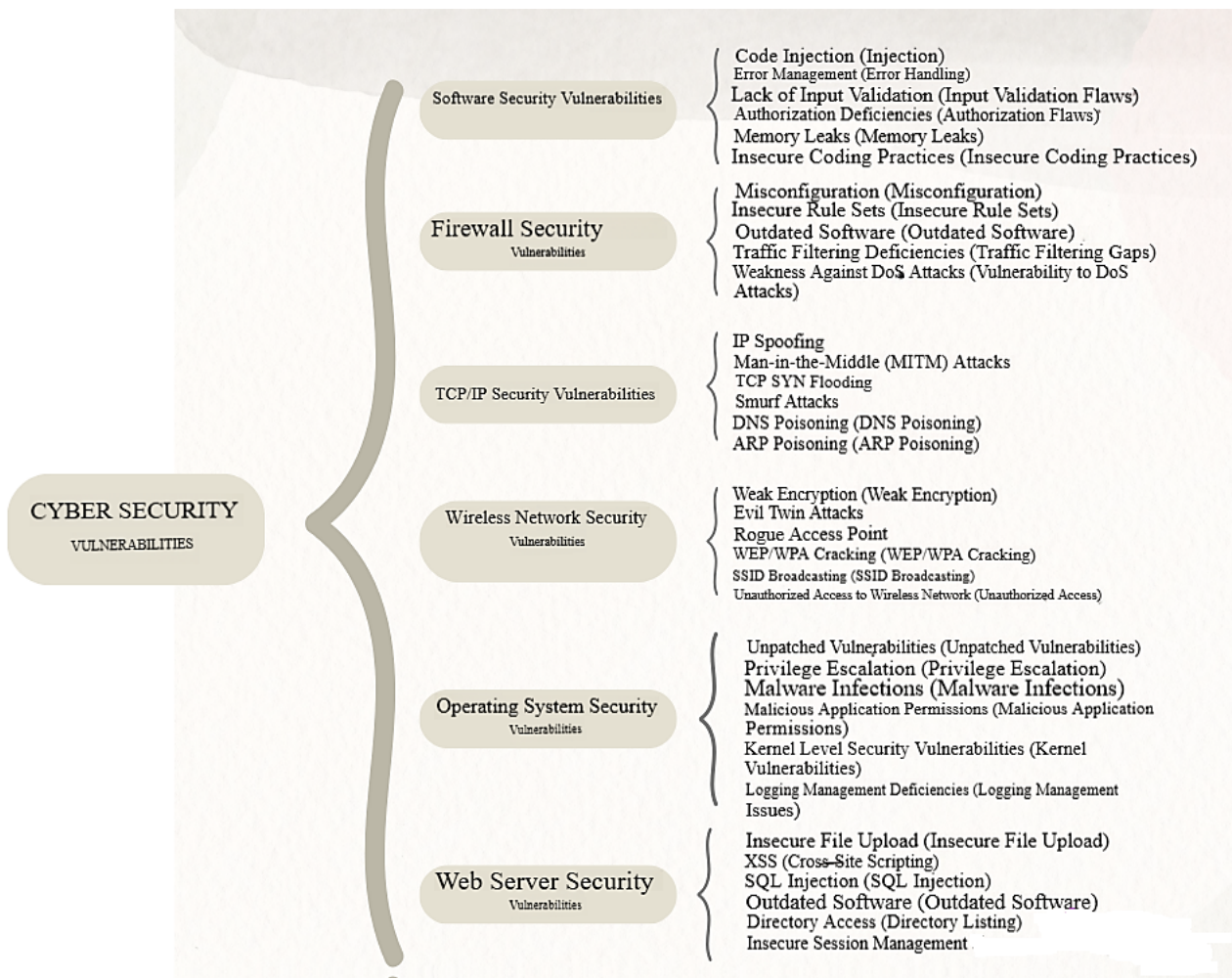


Figure 2. Examples of vulnerabilities

Vulnerabilities encompass a wide spectrum of weaknesses ranging from software coding errors leading to injection attacks to misconfigurations in firewall settings. Each category within these classified and categorized vulnerabilities represents distinct security threats. In cybersecurity, defense mechanisms are typically structured in layered architectures—spanning software, network, and hardware domains—each confronting unique threat vectors. For instance, software vulnerabilities pose risks that are different from those found in network security. The classification of vulnerabilities follows a scientific methodology, whereby the nature and impact of each weakness are systematically analyzed to facilitate a comprehensive understanding. This systematic approach enables the organization to categorize and manage vulnerabilities more effectively.

Software vulnerabilities originate from flaws or omissions during the software design or coding phases. Firewalls protect networks by monitoring and filtering traffic; however, improperly configured or outdated firewalls may allow adversaries unauthorized ingress or egress. For example, a malicious actor could exploit an invalid firewall rule to gain internal network access. The TCP/IP protocol suite, essential for internet communications, contains various security vulnerabilities. Address Resolution Protocol (ARP) attacks manipulate the mapping between IP and MAC addresses to intercept data traffic. In contrast, fragmentation attacks, caused by malicious or erroneous division of IP packets, can degrade network performance or cause denial-of-service conditions. Wireless networks, although similar in vulnerabilities to wired networks, introduce additional threats. Insecure wireless access points may permit attackers to infiltrate the network and exfiltrate data. Security protocols such as SSID and WEP can be vulnerable due to weak encryption or access controls.

Operating system vulnerabilities emerge when system providers fail to apply patches or when administrators misconfigure systems, enabling unauthorized users to gain system access or compromise sensitive data. Web servers hosting and serving websites are also frequent sources of vulnerabilities arising from design flaws or misconfigurations. For instance, spoofing attacks allow malicious users to steal data or disseminate misleading content from the server.

These vulnerabilities exemplify only a subset of the risks across diverse, continuously evolving technological domains. Each vulnerability type involves complex technical details, necessitating persistent vigilance and effort to develop effective protection strategies.

The advent of new vulnerabilities and attack vectors is accelerating alongside technological advancements. The proliferation of Internet of Things (IoT) devices, cloud computing, and mobile platforms presents an expanded attack surface for cyber adversaries. Zero-day attacks—exploits targeting previously unknown vulnerabilities—pose significant threats. Although artificial intelligence (AI) provides substantial advantages in identifying novel attack patterns, its malicious application may also augment the sophistication of cyberattacks. IoT devices, often with limited security measures, can serve as potential network entry points. AI systems are susceptible to manipulation via adversarial training data; for example, models trained on deceptive inputs may produce erroneous or misleading outcomes. These examples underscore vulnerabilities' diversity and dynamic nature, underscoring the necessity for security professionals to continuously adapt to new technologies and threat landscapes [22].

Traditional methods for vulnerability detection, relying heavily on manual scanning, log analysis, and signature-based detection, face increasing challenges related to speed, scalability, and efficacy against sophisticated threats. These conventional techniques often prolong response times and may fail to detect zero-day exploits. In contrast, AI-driven approaches enable real-time processing of large data volumes and rapid anomaly detection. Due to time and resource constraints, manual analysis and rule-based systems struggle to scale with expanding networks and big data.

AI and machine learning facilitate scalable monitoring of network traffic and events. While traditional systems are effective against known threats through signature matching, advanced persistent threats (APT) and zero-day exploits often evade such detection. Machine learning models offer greater flexibility by identifying unknown attack patterns through behavioral analytics and enabling adaptive defenses. Traditional detection relies on static rules, which attackers can circumvent; AI learns normative network behavior and more effectively detects deviations. Anomaly detection via AI allows faster identification of attempts that evade conventional controls.

Traditional systems are vulnerable to false positives—flagging benign activity as threats—and false negatives—failing to detect real attacks—resulting in unnecessary alerts or missed incidents for security analysts. AI improves detection accuracy by more nuanced data analysis, reducing false positives and uncovering hidden threats. AI-based cybersecurity methods comprehensively analyze and swiftly correlate collected data to identify attacks, including previously unseen intrusion attempts.

4. AI-Based Threat Detection

Applying artificial intelligence (AI) and machine learning (ML) systems in cybersecurity transcends traditional security paradigms by providing more dynamic and predictive solutions. AI's capability to process vast amounts of data rapidly and effectively offers significant advantages in cyberattack detection processes. AI-based systems learn complex attack patterns that may be overlooked by conventional methods and can detect these in real time, thereby enhancing the capacity to address continuously evolving threats. Machine learning models improve over time by better recognizing attack signatures, reducing false positives and negatives, and consequently alleviating the workload on cybersecurity professionals. The ability of these technologies to optimize vulnerability detection processes is a critical subject of discussion. While traditional manual methods for identifying vulnerabilities can be time-consuming and error-prone, AI algorithms automate this process, enabling large-scale scanning and yielding more accurate results. Advanced AI techniques such as deep learning and natural language processing have become powerful tools for predicting vulnerabilities and discovering previously unknown weaknesses.

AI methods have revolutionized the detection of cybersecurity vulnerabilities. These methods can identify previously unnoticed vulnerabilities by processing large datasets and swiftly detecting known weaknesses. When focusing on cyberattack detection, effective AI methods and approaches can be categorized according to the type of attack, data source, and threat environment. The primary AI and machine learning methods utilized in this domain include:

1. *Supervised Learning*: This approach employs labeled data to learn specific types of attacks or vulnerabilities. A model is trained on labeled examples and subsequently used to detect similar vulnerabilities in new data. For instance, an organization can train a machine learning model on historical attack data to identify similar attack patterns in the future. This method detects known threats such as phishing emails or Distributed Denial of Service (DDoS) attacks.

2. *Unsupervised Learning*: This technique is used to identify abnormal behaviors or patterns within unlabeled data, enabling the detection of known and previously unseen attacks. Models operating on network traffic or user behavior detect unusual or atypical activities. For example, a sudden large data extraction by a user or multiple logins from different locations within a short period can be flagged as anomalies.

3. *Automated Vulnerability Scanning with AI*: AI-based systems scan existing systems and software for vulnerabilities, delivering faster and more accurate results than traditional scanners. These systems continuously monitor weaknesses and help organizations stay prepared against emerging threats. A company may periodically use an AI-driven vulnerability scanner to assess server and network device weaknesses. Such scanners can identify both known and previously undetected vulnerabilities. Numerous AI-supported systems have been developed for web application security, network security, and cloud security. Table 1 compares widely used AI-based security tools for enterprise network security, highlighting their

advantages and limitations. Critical security features such as anomaly-based threat detection, encrypted traffic analysis, automated security response, and event correlation are emphasized.

Table 1. AI-Driven Cybersecurity Tools Comparison Table

Tool Name	Primary Purpose	AI Utilization	Strengths	Weaknesses	Application Areas	Key Features
Darktrace Enterprise	Detecting and preventing network threats	Machine learning & Anomaly detection	Analyzes anomalous behavior, detects insider threats	May have a high false positive rate	Large enterprise networks, critical infrastructures	Autonomous threat response, AI-driven network traffic analysis
Cisco Secure Analytics	Network visibility and threat analysis	AI-powered anomaly detection	Capable of analyzing encrypted traffic	Complex architecture, high deployment cost	IoT environments, large enterprises, and data centers	Behavior-based threat detection, encrypted traffic inspection
Palo Alto Cortex XDR	End-to-end threat management and incident response	AI & Big Data analysis	Integrated threat intelligence, SIEM compatibility	Requires complex configuration	SOC teams, finance and healthcare sectors	Fileless malware detection, advanced persistent threat (APT) analysis
IBM QRadar SIEM	Log analysis and event management	AI-based event correlation	Real-time attack detection, SIEM integration	Lengthy configuration process	Large-scale enterprises, regulated industries	Advanced event correlation, anomaly detection
Fortinet FortiAI	Threat hunting and attack analysis	AI-powered malware analysis	Automated incident response, compatible with the Fortinet ecosystem	Most effective when used with Fortinet products	Large corporate networks, the financial sector	Anomaly detection, automated threat response
ExtraHop Reveal(x)	Network threat detection	AI & Encrypted traffic analysis	Inspects encrypted traffic and conducts comprehensive threat hunting	Requires high-performance hardware for large data processing	Large enterprises, data centers, and government institutions	Real-time network security monitoring, insider threat detection

4. *Deep Learning*: Utilizing artificial neural networks, deep learning learns more complex attack patterns and applies these models to large datasets. It is particularly useful for detecting sophisticated attacks or vulnerabilities. Deep learning is also employed in malware detection, where models trained on thousands of malware samples can predict whether new files are malicious. This approach is especially effective in identifying zero-day threats, which are previously unknown malware.

5. *Anomaly Detection*: This method learns the normal behavior of a system and flags events or activities that deviate from these norms as threats. It is particularly useful for detecting unauthorized access attempts and unusual system activities. Anomaly detection can identify insider threats or external attacks by analyzing employee email and file movements and issuing alerts when abnormal email traffic or data exfiltration occurs.

6. *Natural Language Processing (NLP)*: NLP analyzes textual data to identify vulnerabilities. It extracts threats and weaknesses from written materials such as source codes, forums, and security reports. Data on newly disclosed vulnerabilities is collected and analyzed by scanning security bulletins, forums, or social media. Furthermore, source code analysis using NLP can identify weak points within the code.

7. *Reinforcement Learning*: Reinforcement learning enables systems to learn and improve continuously. Security systems learn how to better respond to attacks and enhance their decision-making mechanisms over time. This method is

particularly applicable in complex network security scenarios. For example, a firewall or intrusion detection system can be continuously trained against different attack scenarios to develop more effective defense strategies.

Darktrace, Cisco Secure Analytics, and ExtraHop Reveal(x) stand out in anomaly detection and insider threat analysis, while IBM QRadar SIEM and Palo Alto Cortex XDR excel in log analysis and incident management. Fortinet FortiAI provides automated security response, whereas Cisco Secure Analytics is notable for its capability to analyze encrypted traffic. The selection of the most appropriate tool depends on the size of the enterprise network, alignment with the threat model, and integration requirements.

8. *AI-driven Attack Simulations:* AI can simulate cyber attack scenarios to test how resilient systems are. This approach identifies potential attack scenarios and vulnerabilities early, facilitating faster remediation of security weaknesses. A security team may use an AI-generated attack simulation to test the most vulnerable points in the network and observe how the system responds to potential attacks.

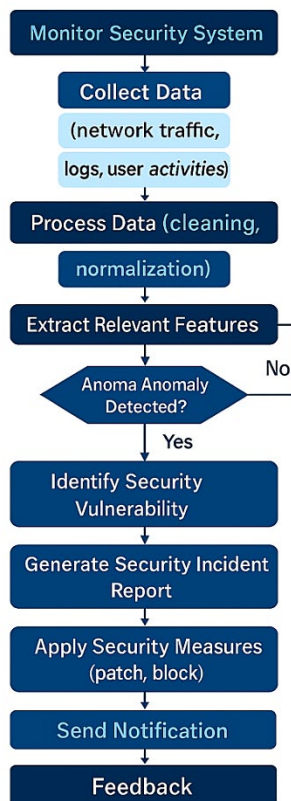


Figure 3. Traditional vulnerability detection process

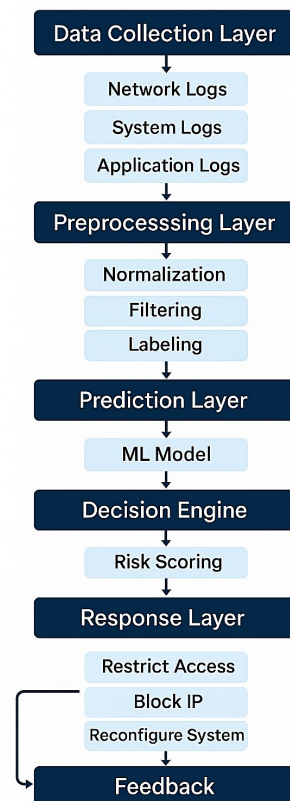


Figure 4. AI-powered vulnerability detection process

Artificial intelligence has become a powerful tool for detecting known and unknown cybersecurity threats. AI-based methods are highly effective at identifying previously unseen variants of attacks and play a critical role in securing large-scale systems. Various machine learning models such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest (RF), and Decision Tree (DT) have been extensively analyzed in the literature using different attack models and datasets, with performance evaluated through metrics including accuracy, precision, recall, and F1-score [23, 24, 25, 26, 27]. Across these studies, high accuracy rates ranging from 95% to 99.5% have been achieved, demonstrating the effectiveness of machine learning models in classifying network traffic as normal or abnormal.

Figure 3 above illustrates the conventional vulnerability detection process. This traditional approach involves identifying system security weaknesses using manual assessments or rule-based (signature-based) automated scans. Cybersecurity experts and vulnerability scanning tools typically carry it out. While it effectively detects known vulnerabilities, it often struggles to identify zero-day attacks due to its reliance on predefined signatures. Figure 4 presents the general flow of an AI-driven dynamic defense process. As depicted, the process begins with a data collection layer, encompassing logs from systems, networks, and applications. The collected data is then preprocessed—normalized and labeled—to ensure compatibility with machine learning models. During the prediction phase, trained models analyze the data to detect anomalous behaviors and forward them to a decision engine. The decision engine performs risk scoring and determines appropriate response actions within the response layer. Actions such as access restriction, IP blocking, or system reconfiguration may be executed. Finally, the resulting feedback contributes to the continuous improvement of the model,

effectively closing the loop and enhancing future threat detection.

Unlike conventional approaches, the AI-driven vulnerability detection process proactively analyzes security threats using machine learning (ML) and deep learning (DL) models. This methodology enables anomaly detection and behavioral analysis to identify unusual or malicious activities. Through adaptive learning, these models can detect emerging threats and zero-day attacks. Real-time analysis also facilitates rapid response to incidents. Table 2 outlines the key differences between traditional and AI-assisted vulnerability detection systems.

Table 2. Comparison of Traditional and AI-Based Vulnerability Detection Methods

Criteria	Traditional Method	AI-Based Method
Methodology	Rule-based (signature-based)	Machine learning and anomaly detection
Detection Capability	Effective against known threats	Can detect unknown threats as well
Zero-Day Vulnerabilities	Hard to detect	Possibility of early detection through model training
False Positive Rate	Maybe high	Lower, learning system
Speed and Automation	May require manual analysis	Real-time analysis and automated action
Flexibility	Requires updates for new threats	Self-adapting model
Prevention Capability	Reactive (post-attack response)	Proactive (pre-attack detection)

Traditional systems, which rely on static rules and signature-based scans, are effective against known threats but often fall short when faced with novel or sophisticated attacks. In contrast, AI-powered systems enhance security by incorporating approaches such as anomaly detection and behavioral analysis, enabling the identification of previously unknown or advanced attack vectors. While traditional models require frequent updates to maintain effectiveness—particularly as new threats emerge and signature databases must be revised—AI-based systems offer a more adaptive and proactive defense strategy, especially in large-scale networks, cloud-based infrastructures, and critical systems.

Nevertheless, it is important to acknowledge that artificial intelligence is not infallible and must operate under human oversight. AI models inherently depend on training data, and adversaries may exploit this dependency by developing techniques to deceive or manipulate AI algorithms. Therefore, integrating human expertise with AI capabilities in a collaborative security framework is essential. A hybrid strategy combining the strengths of both AI-based and conventional approaches will likely yield a more resilient and robust cybersecurity posture.

5. Case Study And Experimental Results

In this study, a subset of the CICIDS2017 dataset was utilized, specifically focusing on the "BENIGN", "Web Attack – Brute Force", "Web Attack – XSS", and "Web Attack – SQL Injection" classes. The dataset, derived from real network traffic, comprises 170,231 instances, encompassing normal activity and various web-based attacks. The distribution of instances among the selected classes is illustrated in Table 3.

Table 3. Class Distribution of the Selected Subset from the CICIDS2017 Dataset

Class Name	Number of Instances
BENIGN	168.051
Web Attack – Brute Force	1.507
Web Attack – XSS	652
Web Attack – SQL Injection	21

5.1 Data Preprocessing

Due to the class imbalance observed in the dataset, the Synthetic Minority Oversampling Technique (SMOTE) was applied to balance the training data. As a result of SMOTE, a balanced training dataset was obtained with 117,631 samples per class. Categorical labels were converted into a numerical format. The dataset was examined for missing (NaN) and irrelevant (infinite) values, and it was confirmed that no such values were present. Therefore, the dataset was considered ready for classification tasks.

All data preprocessing, modeling, and evaluation procedures were conducted using the Python programming language on the Google Colab platform, which provides a Linux-based environment with Python version 3.11, 12 GB of RAM, an Intel Xeon 2.20GHz virtual CPU, and pre-installed libraries including scikit-learn 1.6.1, pandas 2.2.2, imbalanced-learn 0.13.0, matplotlib, and seaborn.

An initial analysis of numerical features revealed the presence of infinite (inf, -inf) and undefined (NaN) values. Such anomalies may arise from division-by-zero errors or measurement faults in network traffic records. All inf and -inf values were first replaced with NaN, and any rows containing NaN values were subsequently dropped. This step was essential for the robust functioning of the learning algorithms.

Machine learning models cannot process categorical string values directly. Therefore, the Label column, which contains the target classes (e.g., "BENIGN" and "ATTACK"), was converted into a binary numerical format using a LabelEncoder. The result was a binary classification setup, where benign flows were labeled 0 and attack flows were labeled 1.

Certain features, such as IPV4_SRC_ADDR, IPV4_DST_ADDR, and DNS_QUERY_ID, were identified as irrelevant for model training or potential sources of data leakage. The dataset excluded these attributes to reduce complexity and enhance model generalizability.

To facilitate correlation analysis and model training, only numerical columns were retained. This ensures compatibility with algorithms that require numerical input for computation and distance-based measures.

The preprocessing procedures described above were implemented to enhance the analysis's reliability and optimize model performance. The resulting cleaned and transformed dataset was subsequently used in machine learning models training and evaluation phases of machine learning models.

5.2 Model Training and Evaluation

In developing the model, the Random Forest algorithm was selected due to its proven effectiveness in classification tasks. As an ensemble method composed of multiple decision trees, Random Forest delivers high accuracy, particularly in multiclass datasets characterized by high dimensionality and class imbalance. Through random subsampling (bootstrap) and feature selection, the model is rendered more robust against the risk of overfitting.

During the experimental analysis phase, a correlation matrix—presented in Figure 5—was constructed to examine the interrelationships among numerical features derived from network traffic. This matrix provides visual insights into positive and negative linear correlations among variables, offering valuable guidance for feature engineering. Notably, strong positive correlations were observed among variables related to traffic volume, such as IN_BYTES, OUT_BYTES, and NUM_PKTS_256_TO_511_BYTES. Similarly, symmetric protocol-related features like TCP_WIN_MAX_IN and TCP_WIN_MAX_OUT demonstrated parallel patterns. These findings suggest that certain flow behaviors are recurrent within the network and that some features may be evaluated collectively. Considering this, feature selection was carefully executed to avoid potential performance degradation caused by redundant variables. This correlation analysis was conducted to enhance the precision of AI-based intrusion detection models while minimizing the impact of redundant information.

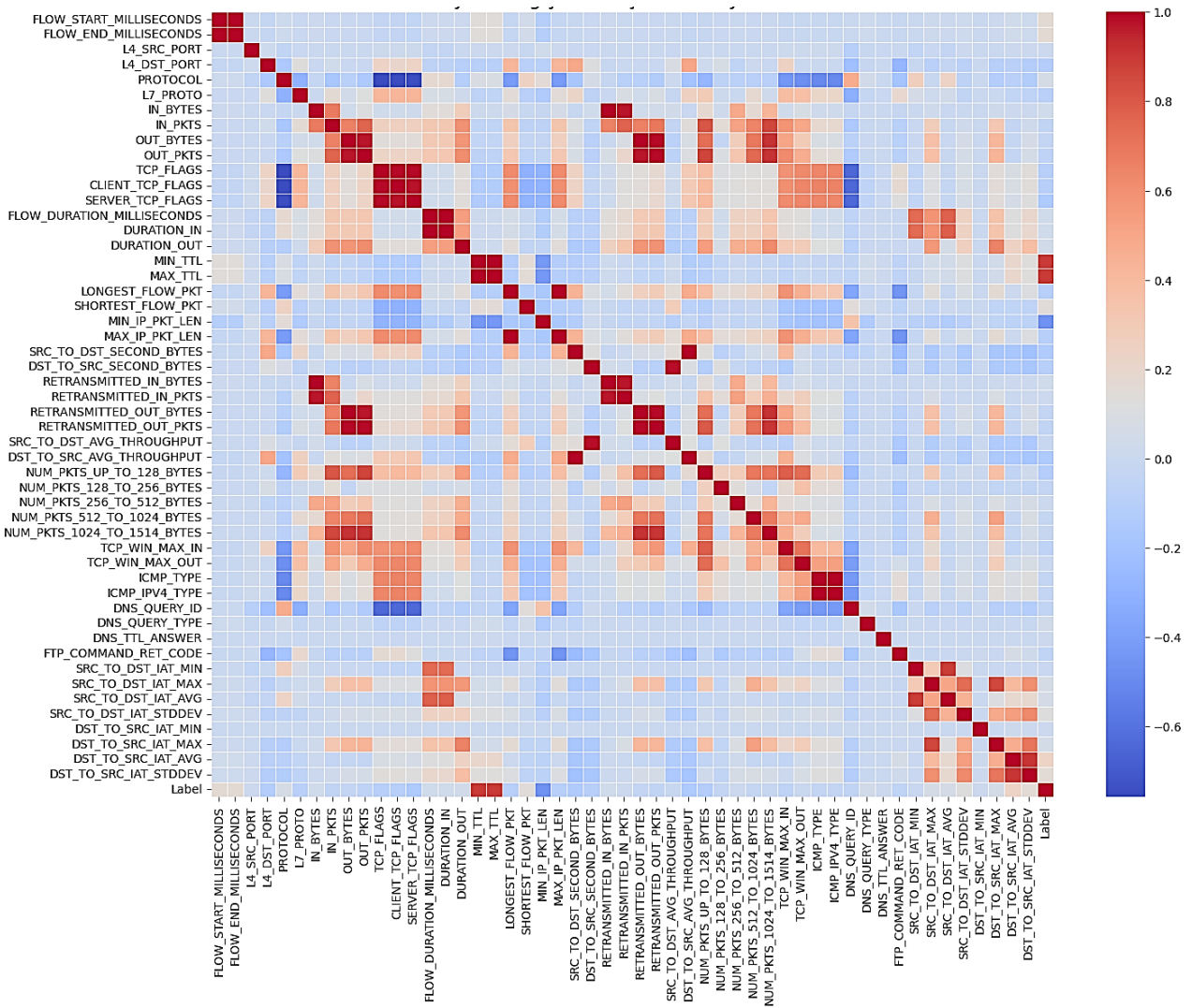


Figure 5. Correlation matrix among numerical network traffic features.

The dataset was split into 80% training and 20% testing subsets, and training was conducted using the Random Forest classifier. The performance of the classification model, trained to detect anomalies in network traffic data, was evaluated through various metrics and graphical analyses. The model trained on the SMOTE-balanced training set demonstrated the following performance on the test data.

The confusion matrix shown in Figure 6 indicates that the model generally achieves successful discrimination between classes. Specifically, 50,413 samples in the "BENIGN" (normal traffic) class were correctly classified, with only 7 misclassifications demonstrating the model's near-perfect ability to distinguish normal traffic. For the "Web Attack – Brute Force" class, 313 452 samples were correctly classified, while 134 were mostly confused with the "XSS" attack class. This suggests that Brute Force and XSS attack types share certain similarities, making it challenging for the model to differentiate between these two classes. Although the "Web Attack – SQL Injection" class consisted of only 6 samples, 5 were correctly predicted. While this indicates high accuracy, the small sample size limits generalizability. Finally, in the "Web Attack – XSS" class, only 93 out of 196 samples were correctly classified, with 101 samples confused with the "Brute Force" class. This reveals that XSS attacks tend to be misclassified with other classes, indicating the need for performance improvements specifically for this category.

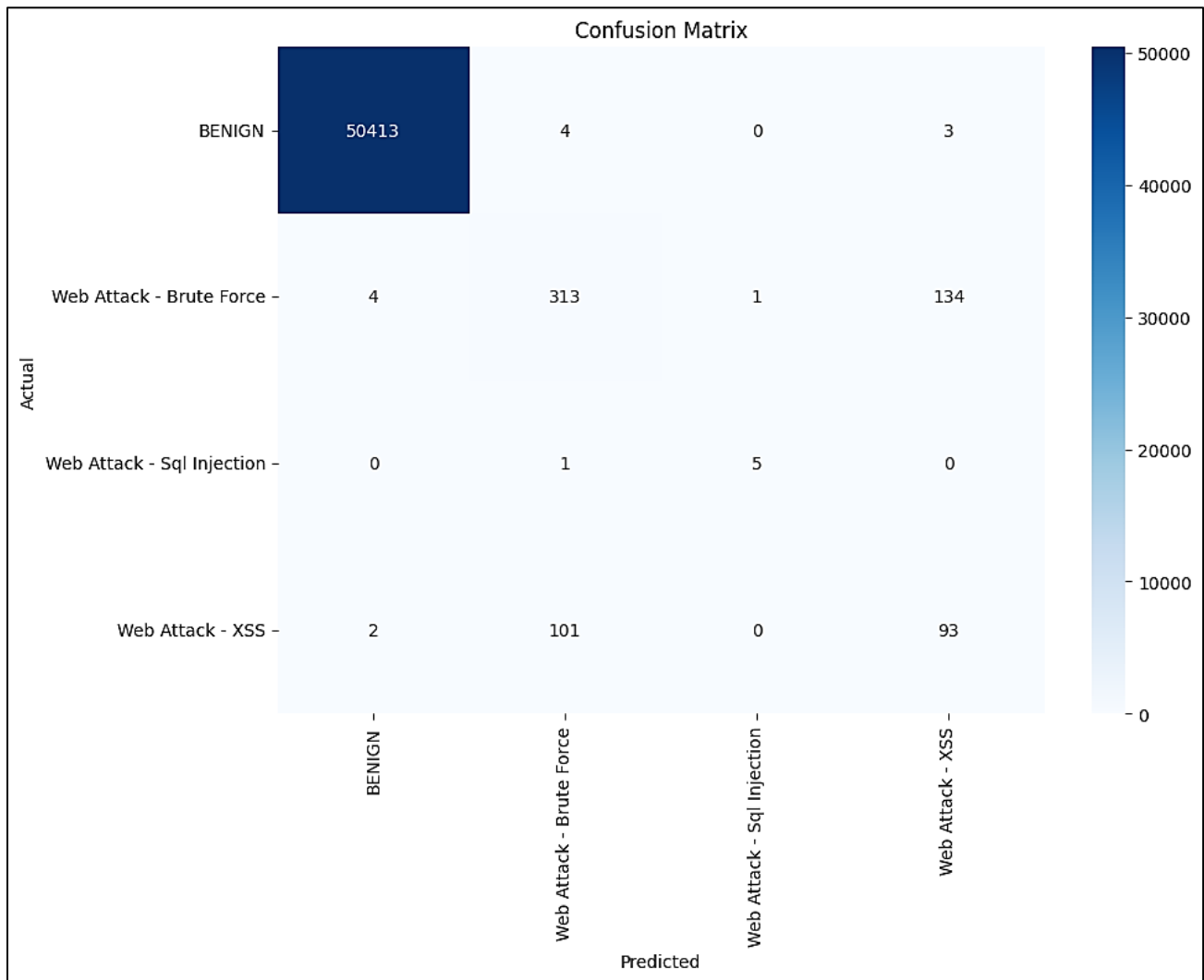


Figure 6. Confusion matrix

When evaluating the class-based performance metrics of the model presented in Table 4, it is evident that it demonstrates near-perfect performance in the BENIGN (normal traffic) class. All metrics for this class—Precision, Recall, and F1-score—are 1.00, indicating the model’s exceptional capability in distinguishing normal traffic. For the Web Attack – Brute Force class, the model achieved reasonable performance with a Precision of 75%, a Recall of 69%, and an F1-score of 72%; however, occasional confusion with other attack types, particularly XSS, was observed. Despite containing only 6 samples in the dataset, the SQL Injection attack class attained an F1-score of 83%. Nevertheless, this result should be interpreted cautiously due to the limited sample size. XSS attacks proved to be one of the most challenging classes for the model, with Precision, Recall, and F1-score values remaining at 40%, 47%, and 44%, respectively. This outcome suggests a potential overlap of data characteristics specific to XSS attacks with those of other attack types, highlighting the need for further model enhancement.

Table 4 Class-Based Performance Metrics

Class	Precision	Recall	F1-Score	Support
BENIGN	1.00	1.00	1.00	50420
Web Attack – Brute Force	0.75	0.69	0.72	452
Web Attack – SQL Injection	0.83	0.83	0.83	6
Web Attack – XSS	0.40	0.47	0.44	196
Accuracy			1.00	51074
Macro Avg	0.75	0.75	0.75	
Weighted Avg	1.00	1.00	1.00	

According to the ROC (Receiver Operating Characteristic) curve analysis in Figure 7, the model demonstrates a strong ability to distinguish between classes. The curves, particularly for the "BENIGN" class, show an AUC (Area Under the Curve) value close to 1.0, indicating that the model accurately distinguishes normal from anomalous network traffic. Notably, despite the limited number of samples in the "SQL Injection" attack class, the area under the curve is relatively high, suggesting that the model has learned this class well. However, the ROC curves for the "Brute Force" and "XSS" classes tend to overlap more, with comparatively lower AUC values. This suggests a potential feature overlap between these two attack types, making them more difficult for the model to distinguish. The ROC curves indicate that the model possesses effective class discrimination capabilities and performs reasonably well in classification despite class imbalance.

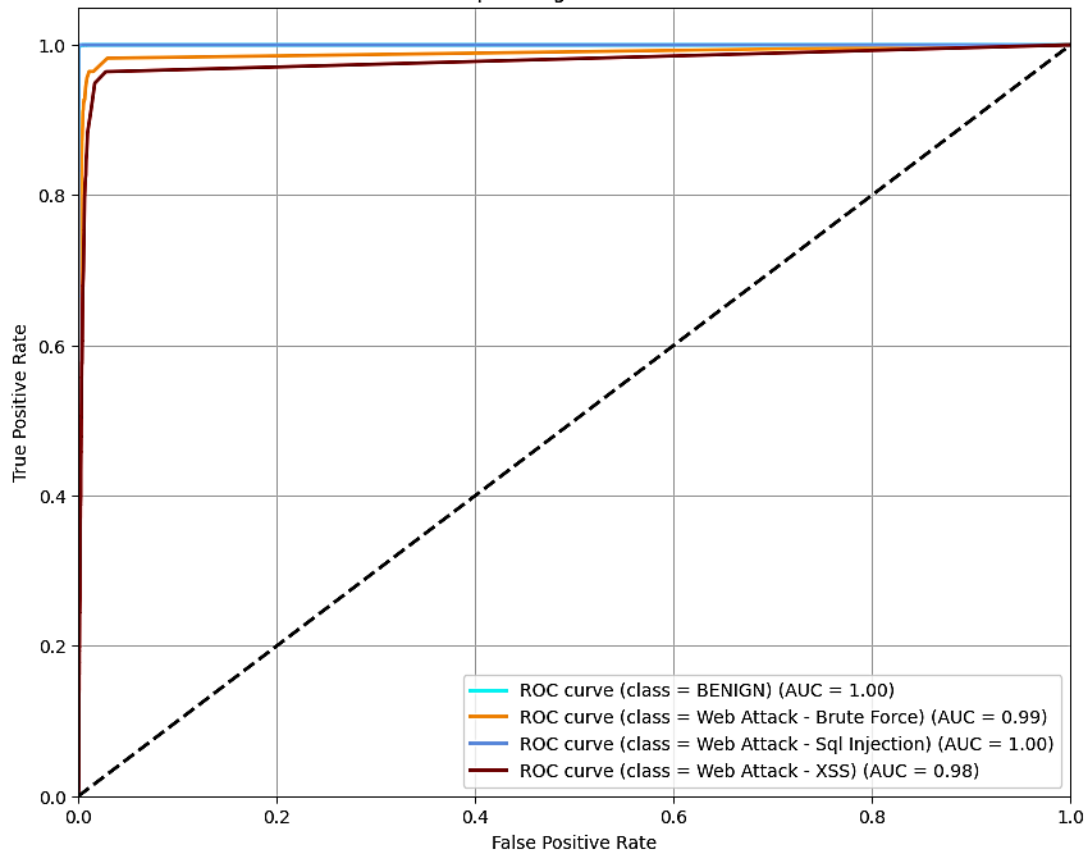


Figure 7. Receiver Operating Characteristic-ROC

Table 5 presents the AUC (Area Under the Curve) values, which reflect the model’s ability to distinguish each class from the others. Classes with an AUC of 1.00—namely, BENIGN and Web Attack – SQL Injection—indicate that the model is nearly perfect in distinguishing these categories. The 0.99 AUC for Web Attack – Brute Force also denotes very strong performance, while the 0.98 AUC for Web Attack – XSS suggests that the model is still highly effective at identifying this class, albeit with slightly lower discriminative power than the others. Overall, with all AUC values at or above 0.98, the model demonstrates a strong ability to differentiate between classes. This outcome confirms that the model has achieved effective and robust learning across all categories despite the inherent class imbalance.

Table 5. Class-Based AUC (Area Under the Curve) Values

Class	AUC Value
BENIGN	1.00
Web Attack – Brute Force	0.99
Web Attack – SQL Injection	1.00
Web Attack – XSS	0.98

As a result of the feature importance analysis derived from the Random Forest model, the top 15 most influential features were identified. These features are visualized in Figure 8 as a horizontal bar chart sorted by their importance scores. The most significant feature was Init_Win_bytes_backward, which represents the initial window size of packets in the backward direction and is a strong indicator for distinguishing attack behaviors. Other prominent features include Fwd IAT

Min, Flow Packets/s, and Flow IAT Mean, primarily temporal and volumetric metrics related to network flow. This analysis enhances the interpretability of the system by revealing which network traffic characteristics the model emphasizes during its decision-making process.

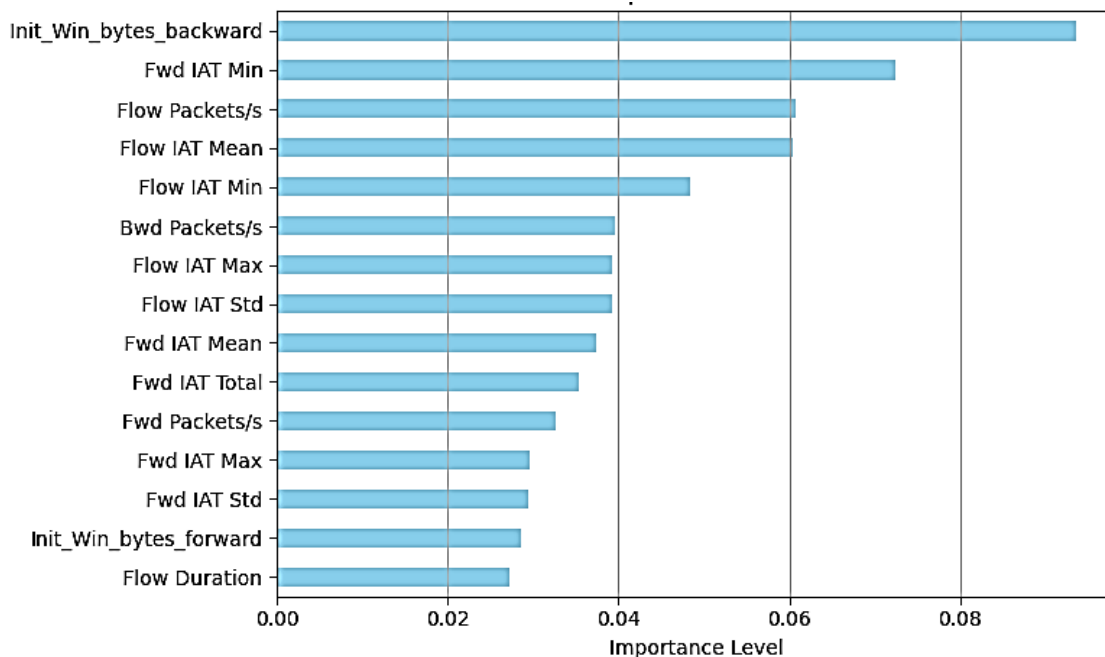


Figure 8. Top 15 features

The Web Attacks subset of the CIC-IDS2017 dataset used in this study serves as a significant benchmark for intrusion detection due to its inclusion of class imbalance, realistic network traffic structure, and multiple attack types. During preprocessing, missing and infinite values were addressed, and the training data was balanced using the SMOTE technique. The Random Forest classifier trained on this processed dataset accurately detected the BENIGN and SQL Injection classes. The model achieved 99.93% accuracy, a high F1-score, and strong class-specific ROC-AUC curves, indicating a highly effective intrusion detection capability. Table 6 below summarizes a comparison of the results of this study with similar works in the existing literature.

Table 6. Comparative Analysis of Similar Studies in the Literature

Study	Model(s)	Accuracy (%)	Description
Kurniabudi et al. (2021) [28]	Random Forest	99.86	High accuracy achieved through information gain-based feature selection
M. L. Ali et al. (2025) [29]	Random Forest	99.88	High performance obtained after feature selection
R. A. Disha et al. (2021) [30]	SVM + Naive Bayes	98.92	Hybrid model applied to the CICIDS2017 dataset
O. Edosa et al. (2024) [31]	Deep Learning Model	99.68	High accuracy achieved through deep learning approaches
M. T. Abdelaziz et al. (2024) [32]	Random Forest	99.96	High accuracy with a low false positive rate
This Study	Random Forest	99.93	Balanced classification, feature importance analysis, and detailed ROC/F1 reports

The key contributions of this study include the application of SMOTE to address class imbalance, analysis based on feature importance, a multi-class classification structure, performance evaluation supported by visualizations, and the balanced classification of all web attack types, which are rarely addressed collectively in existing literature. Additionally, the reproducibility of the employed methods within the Google Colab environment enhances the practicality and applicability of the study.

6. Dynamic Defense Approaches

In the face of advanced cyber threats, AI-based dynamic defense systems have become indispensable components of modern security architectures. These systems detect threats in advance, generate real-time responses, and enhance system adaptability. Dynamic defense approaches refer to systems that can continuously and instantly adapt to the threat landscape instead of relying on static security policies. Often supported by artificial intelligence or machine learning, such systems can self-update and respond to emerging threats. When integrated with AI-driven mechanisms, dynamic defense strategies offer proactive and reactive solutions against cyberattacks. Unlike traditional methods based on fixed rules, dynamic defense systems are adaptive and predictive, making them essential in a constantly evolving threat environment.

Scenario Example 1: Consider a financial institution using an AI-enabled intrusion detection system (IDS). Upon detecting unusual data exfiltration attempts during non-business hours—such as large outbound traffic to a foreign server—the system immediately isolates the affected endpoint and blocks outbound communication. Simultaneously, it sends real-time alerts and logs forensic data for analysis. This kind of real-time, autonomous reaction would be impossible with static rule-based systems [33].

A core component of dynamic defense is AI-based anomaly detection, which can learn system and network behavior deviations. Normal traffic patterns are modeled using machine learning and deep learning algorithms, and deviations from these patterns are flagged as potential threats. Notably, algorithms such as Autoencoders, LSTM (Long Short-Term Memory), and Isolation Forest effectively detect time-dependent anomalies [34].

Scenario Example 2: In a cloud infrastructure, an AI system trained using LSTM detects a slow, stealthy data leak where small chunks of sensitive information are exfiltrated over several hours. Because this deviates from historical data patterns, the system flags the session and throttles the connection speed while notifying the security team [35].

Components like firewalls, access control lists (ACLs), and IDS/IPS systems can automatically update based on AI-assisted risk analysis. For instance, if an anomaly is detected in traffic from a specific IP address, the system may dynamically block the IP or quarantine the traffic for deeper analysis [36]. This enables automated defense reflexes without requiring human intervention.

Scenario Example 3: A university network experiences a brute-force SSH attack from multiple IPs. A dynamic defense system trained with Isolation Forest instantly detects the behavior as anomalous, updates the firewall rules to block those IP ranges, and adjusts threshold rules for future logins. All this occurs within seconds—far faster than manual response [37].

Systems that continuously monitor user behavior and dynamically update access policies replace static Role-Based Access Control (RBAC) structures. For example, if a user attempts to access a system from a previously unused geographic location, the system may require additional multi-factor authentication or restrict access privileges. Behavioral analyses in such systems are typically conducted using algorithms like Naive Bayes, K-Nearest Neighbors (KNN), or Deep Neural Networks (DNN) [38].

Scenario Example 4: An employee logs in from Germany regularly in a multinational company. One morning, an access attempt was made in Vietnam. The AI-based access control system flags this as high-risk, prompts for biometric verification, and temporarily limits access to sensitive modules until verification is complete [39].

Automation in vulnerability remediation is critical in rapidly and effectively eliminating detected security flaws, offering a significant advantage in securing systems. AI and automated systems enable the quick detection, neutralization, and mitigation of threats. This facilitates fast and efficient vulnerability management while minimizing the need for manual intervention. As a result, cybersecurity teams can operate more efficiently and effectively, significantly improving the overall security posture of an organization.

Scenario Example 5: Following a vulnerability scan in a DevOps pipeline, an AI agent identifies a vulnerable software dependency. It checks a vulnerability database (e.g., NVD), finds an approved patch, applies the update in the staging environment, performs regression tests, and deploys it—all within minutes, without manual oversight [40].

7. Conclusion and Future Work

This study used a subset derived from the CICIDS2017 dataset, which was reduced to four classes. The SMOTE technique was applied to address the imbalanced class distribution, and a Random Forest classifier was trained. The experimental results demonstrate that the model achieves high performance in detecting cyber-attacks.

The model successfully distinguished between anomalous and normal network traffic on the test set with an accuracy rate of 99.93%. Notably, the normal traffic (BENIGN) class achieved maximum Precision, Recall, and F1-score values of 1.00, indicating extremely low false positive and false negative rates. Among attack classes, the Web Attack – SQL Injection class showed high performance (% 83% F1-score), though the limited number of samples restricts the generalizability of this result. Similarly, the low frequency of samples in classes such as XSS and SQL Injection may impact the robustness of model performance for these specific categories, which should be considered in future evaluations. The model's performance was relatively lower for the Web Attack – Brute Force and XSS classes, with a notable tendency for

misclassification between these two classes. This suggests that the similar characteristics shared by Brute Force and XSS attacks pose challenges to the model, highlighting the need for advanced feature engineering and model improvements to enhance discrimination between these classes in future work.

ROC curve and AUC analyses confirmed the model's strong discriminatory power across all classes; AUC values exceeding 0.98 reinforce the model's overall classification success despite the imbalanced data structure. Feature importance analysis revealed that temporal and volumetric network traffic metrics—such as `Init_Win_bytes_backward`, `Fwd_IAT_Min`, and `Flow_Packets/s`—play a critical role in attack detection. These findings increase the explainability of the model's decision-making processes, contributing to reliability and transparency in cybersecurity applications.

Compared to similar studies using comparable datasets and models, the obtained 99.93% accuracy and detailed class-based performance metrics demonstrate this work's effectiveness and practical applicability. Furthermore, the study's key strengths include using SMOTE for class balancing, balanced modeling of multiple attack types, and comprehensive visualization-supported analyses.

In addition, the potential for real-time deployment of the model could be enhanced by integrating it into stream-processing frameworks such as Apache Kafka or Flink. This would enable near-instantaneous detection in dynamic network environments, thereby extending the practical relevance of the proposed approach.

Future research should focus on advanced feature engineering to improve the discrimination of similar attack types, integrate deep learning-based methods, and adapt to real-time attack detection scenarios. Such developments could create more robust and scalable protection mechanisms against the increasing variety of cyber threats.

References

- [1] N. G. Camacho, "The Role of AI in Cybersecurity: Addressing Threats in the Digital Age," *J. Artif. Intell. Gen. Sci. (JAIGS)*, vol. 3, no. 1, pp. 143–154, 2024.
- [2] N. Mohamed, "Artificial intelligence and machine learning in cybersecurity: a deep dive into state-of-the-art techniques and future paradigms," *Knowledge and Information Systems*, 2025. [Online]. Available: <https://doi.org/10.1007/s10115-025-02429-y>.
- [3] K. M. Roba Abbas, J. Pitt, K. M. Vogel, and M. Zafeirakopoulos, "Artificial Intelligence (AI) in Cybersecurity: a socio-technical research roadmap," 2022. [Online]. Available: https://www.turing.ac.uk/sites/default/files/2023-11/ai_in_cybersecurity.pdf
- [4] N. Mohamed, "Current trends in AI and ML for cybersecurity: A state-of-the-art survey," *Cogent Eng.*, vol. 10, no. 2, 2023.
- [5] M. I. Alghamdi, "Survey on Applications of Deep Learning and Machine Learning Techniques for Cyber Security," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 16, 2020.
- [6] S. Ali, S. U. Rehman, A. Imran, G. Adeem, Z. Iqbal, and K.-I. Kim, "Comparative Evaluation of AI-Based Techniques for Zero-Day Attacks Detection," *Electronics*, vol. 11, no. 23, p. 3934, 2022.
- [7] S. Zeadally, E. Adi, Z. Baig, and I. A. Khan, "Harnessing Artificial Intelligence Capabilities to Improve Cybersecurity," *IEEE Access*, vol. 8, pp. 23817–23837, 2020.
- [8] M. Tayyab, M. Marjani, N. Z. Jhanjhi, I. A. T. Hashem, R. S. A. Usmani, and F. Qamar, "A comprehensive review on deep learning algorithms: Security and privacy issues," *Comput. Secur.*, vol. 131, p. 103297, 2023.
- [9] S. Al-Mansoori and M. B. Salem, "The Role of Artificial Intelligence and Machine Learning in Shaping the Future of Cybersecurity: Trends, Applications, and Ethical Considerations," *Int. J. Soc. Analytics*, vol. 8, no. 9, pp. 1–16, 2023.
- [10] M. S. Akhtar and T. Feng, "An overview of the applications of Artificial Intelligence in Cybersecurity," *EAI Endorsed Trans. Creat. Technol.*, vol. 8, no. 29, p. e4, 2021.
- [11] A. D. Sontan and S. V. Samuel, "The intersection of Artificial Intelligence and cybersecurity: Challenges and opportunities," *World J. Adv. Res. Rev.*, vol. 21, no. 2, pp. 1720–1736, 2024.
- [12] M. M. Yamin, M. Ullah, H. Ullah, and B. Katt, "Weaponized AI for cyber attacks," *J. Inf. Secur. Appl.*, vol. 57, p. 102722, 2021.
- [13] G. Apruzzese, L. Ferretti, M. Marchetti, M. Colajanni, and A. Guido, "On the effectiveness of machine and deep learning for cyber security," in *10th Int. Conf. Cyber Conflict (CyCon)*, IEEE, pp. 371–390, 2018.
- [14] R. Kaur, D. Gabrijelečić, and T. Klobučar, "Artificial intelligence for cybersecurity: Literature review and future research directions," *Inf. Fusion*, vol. 97, p. 101804, 2023.
- [15] J. Doshi and B. Trivedi, "Comparison of vulnerability assessment and penetration testing," *Int. J. Appl. Inf. Syst.*, vol.

- 8, no. 6, pp. 51–54, 2015.
- [16] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, “A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions,” *Electronics*, vol. 12, no. 6, p. 1333, 2023.
- [17] S. Kuipers and M. Schonheit, “Data breaches and effective crisis communication: a comparative analysis of corporate reputational crises,” *Corp. Reputation Rev.*, vol. 25, no. 3, pp. 176–197, 2022.
- [18] U. Bansal, “A review on ransomware attack,” in *2nd Int. Conf. Secure Cyber Comput. Commun. (ICSCCC)*, IEEE, pp. 221–226, 2021.
- [19] OWASP, “Top 10 Web Application Security Risks,” OWASP, 2017. [Online]. Available: <https://owasp.org/www-project-top-ten/2017/>.
- [20] OWASP, “Top 10 Web Application Security Risks,” OWASP, 2024. [Online]. Available: <https://owasp.org/Top10/>.
- [21] OWASP, “API Security Top 10 – 2023 Edition,” OWASP, 2024. [Online]. Available: <https://owasp.org/API-Security/editions/2023/en/Ox11-t10/>.
- [22] P. Radanliev and O. Santos, “Adversarial Attacks Can Deceive AI Systems, Leading to Misclassification or Incorrect Decisions,” *Preprints*, 2023.
- [23] U. Ahmed, M. Nazir, A. Sarwar et al., “Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering,” *Sci. Rep.*, vol. 15, p. 1726, 2025.
- [24] V. Kanimozhi and D. T. P. Jacob, “Calibration of various optimized machine learning classifiers in network intrusion detection system on the realistic cyber dataset CseCic-Ids2018 using cloud computing,” *Int. J. Eng. Appl. Sci. Technol.*, vol. 4, pp. 209–213, 2019.
- [25] A. Verma and V. Ranga, “On evaluation of network intrusion detection systems: statistical analysis of CIDDS-001 dataset using machine learning techniques,” *Pertanika J. Sci. Technol.*, vol. 26, pp. 1307–1332, 2018.
- [26] W. Yassin, N. I. Udzir, and Z. Muda, “Anomaly-based intrusion detection through Kmeans clustering and Naive Bayes classification,” in *Proc. 4th Int. Conf. Comput. Informatics (ICOI)*, 2013.
- [27] I. F. Kilincer, F. Ertam, and A. Sengur, “Machine learning methods for cyber security intrusion detection: Datasets and comparative study,” *Comput. Netw.*, vol. 188, p. 107840, 2021.
- [28] K. Kurniabudi, D. Stiawan, D. Darmawijoyo, M. Y. B. Idris, B. Kerim, and R. Budiarto, “Important features of CICIDS-2017 dataset for anomaly detection in high dimension and imbalanced class dataset,” *Indones. J. Electr. Eng. Inform.*, vol. 9, no. 2, pp. 498–511, 2021.
- [29] M. L. Ali, K. Thakur, S. Schmeelk, J. DeBello, and D. Dragos, “Deep learning vs. machine learning for intrusion detection in computer networks: A comparative study,” *Appl. Sci.*, vol. 15, no. 4, p. 1903, 2025.
- [30] R. A. Disha and S. Waheed, “Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique,” *Cybersecurity*, vol. 5, no. 1, p. 1, 2022.
- [31] O. Edosa, P. E. Orukpe, and U. Iruansi, “Design and implementation of a deep neural network approach for intrusion detection systems,” *e-Prime – Adv. Electr. Eng., Electron. Energy*, vol. 7, p. 100434, 2024.
- [32] F. Idhammad, M. Bakkali, and M. Elghazi, “Deep learning for cybersecurity intrusion detection: Approaches, datasets, and comparative study,” *Comput. Secur.*, vol. 123, p. 102968, 2023.
- [33] S. Mishra, “Exploring the impact of AI-based cyber security financial sector management,” **Applied Sciences**, vol. 13, no. 10, p. 5875, 2023. doi:10.3390/app13105875
- [34] A. Z. Alomari and M. H. Alshammari, “Cybersecurity applications of machine learning: State of the art and challenges,” *Cybersecurity*, vol. 3, no. 1, p. 1, 2020.
- [35] S. P. Singh and N. Afzal, “THE MESA SECURITY MODEL 2.0: A DYNAMIC FRAMEWORK FOR MITIGATING STEALTH DATA EXFILTRATION,” **Int. J. Network Security & Its Applications**, vol. 16, no. 3, May 2024, doi:10.5121/ijnsa.2024.16302
- [36] H. Yin, D. He, S. Qian, J. Liu, and K. Wang, “A survey on cybersecurity intrusion detection based on deep learning,” *J. Cyber Secur. Technol.*, vol. 5, no. 4, pp. 231–255, 2021.
- [37] K. Tallam, “CyberSentinel: An Emergent Threat Detection System for AI Security,” **arXiv**, Feb. 20, 2025. Available: <https://arxiv.org/abs/2502.14966>
- [38] T. M. Nguyen, D. T. Nguyen, and S. Y. Shin, “A comprehensive review of machine learning for cybersecurity,” *IEEE*

Access, vol. 9, pp. 88968–89004, 2021.

- [39] ISACA, “Adaptive Access Control: Navigating Cybersecurity in the Era of AI and Zero Trust,” *ISACA Now Blog*, Apr. 2025. [Online]. Available: <https://www.isaca.org/resources/news-and-trends/isaca-now-blog/2025/adaptive-access-control-navigating-cybersecurity-in-the-era-of-ai-and-zero-trust>
- [40] M. Seo, W. Choi, M. You, and S. Shin, “AutoPatch: Multi-Agent Framework for Patching Real-World CVE Vulnerabilities,” *arXiv*, May 2025. Available: <https://arxiv.org/abs/2505.04195>

Article Information Form

Authors Contributions

The artificial intelligence studies and code development were carried out by Şahin Kara, the literature review and initial drafting of the manuscript by Fatih İlkbahar, and the manuscript was reviewed and finalized by Muhammed Zekeriya Gündüz.

Acknowledgments

We would like to thank the Canadian Institute for Cybersecurity (CIC) and the Communications Security Establishment (CSE) for providing the publicly available dataset used in this study.

Conflict of Interest Notice

The authors declare that no conflicts of interest are associated with this article's publication.

Artificial Intelligence Statement

The dataset analyzed in this study is publicly available and can be accessed from the Canadian Institute for Cybersecurity (CIC) at: [<https://registry.opendata.aws/cse-cic-ids2018/>]

Plagiarism Statement

This article has been scanned by iThenticate™.

Ethical Statement

This study does not involve any personal data. All analyses were performed on the publicly available CICIDS2017 dataset. Therefore, ethical committee approval is not required. The dataset used in this study was anonymized and released for research purposes by the Communications Security Establishment (CSE) and the Canadian Institute for Cybersecurity (CIC).