Journal of Artificial Intelligence and Autonomous Intelligence Received 29/10/2025; Accepted 24/12/2025; Published 31/12/2025

Machine Learning for Cyber Defense: Comprehensive Survey of Datasets and Techniques for Network, Host and Application based Cyber Attacks

Aakanksha aakanksha.1@rajguru.du.ac.in
Shaheed Rajguru College of Applied Sciences for Women
University of Delhi, India

Anamika Gupta anamikargupta@sscbsdu.ac.in
Shaheed Sukhdev College of Business Studies
University of Delhi, India

Richa richa.24722@sscbs.du.ac.in
Shaheed Sukhdev College of Business Studies
University of Delhi, India

Sanjay Singh sanjay.24725@sscbs.du.ac.in
Shaheed Sukhdev College of Business Studies
University of Delhi, India

Corresponding Author: Anamika Gupta

Abstract

The rapid advancements in the communication technology and information exchange in cyber space has led to the issue of cyber-attacks. As the attackers are finding new techniques of designing the cyber-attacks, there is an urgent need to design a robust cyber-attack detection and mitigation system. This study explores the various Artificial Intelligence (AI) and Machine Learning (ML) based approaches for the detection of cyber-attacks. The different threats and risks have been categorized into three main types: network-based, host-based, and application-level attacks. Various AI/ML algorithms such as Random Forest(RF), Support Vector Machines (SVM), Convolutional Neural Networks (CNN), and LSTM-based architectures used on existing datasets are compared based on their detection capabilities, accuracy, and application contexts. The paper also tries to identify challenges related to the quality of datasets, model interpretability, and the detection of zero-day attacks in an attempt to highlight the need for AI-driven smart, hybrid and adaptive solutions. The survey conducted in this paper serves as a foundation for researchers and practitioners who are aiming to develop robust intrusion detection systems to mitigate advance cyber-attacks.

Keywords: Cybersecurity, Intrusion Detection, Machine Learning, Network Security, AI, Anomaly Detection, Cyber Attacks, Datasets

Citation: Anamika Gupta et al. Machine Learning for Cyber Defense: Comprehensive Survey of Datasets and Techniques for Network, Host and Application based Cyber Attacks. **Advances in Knowledge-Based Systems, Data Science, and Cybersecurity**.Research 2025;2(3):19.

Page No. 386-411.

1. INTRODUCTION

The complexity of cyber-attacks and their frequency have increased drastically over the years. It has greatly impacted functionality of individuals, organizations, and critical infrastructure. Traditional rule-based security systems are ineffective in the presence of new complex and unknown cyber threats, such as polymorphic malware, zero-day exploits, and stealthy intrusions. Therefore, AI can be a saviour and an ally in creating intelligent detection systems that can automate the detection process to enhance cybersecurity [1]. Machine Learning (ML) techniques can assist in mining the historical data to find new attack patterns and unseen threats to better combat the zero day attacks. Researchers have used AI and ML techniques to analyze cyber-attacks across different attack surfaces [2], including network traffic, host system behavior, and web application interactions. For different tasks, specialized ML methodologies are employed, like, supervised learning for classification tasks, unsupervised learning is applied for anomaly detection, and complex patterns are recognized using deep learning techniques. Researchers have proposed various AI and ML techniques [3] to enhance the efficiency of Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). This paper tries to present a detailed review of AI and ML techniques used for detecting different types of cyber-attacks. It covers three broad categories of cyber-attacks: network-level, host-level, and application-level, based on the type of attack surface. It also discusses various attack types within these broad categories. The effectiveness of different ML models has been evaluated on various datasets. This study aims to unveil current challenges and opportunities in this field, in order to equip researchers and practitioners with an in-depth overview of the state-of-the-art in AI-based cyber-attack detection to counter and predict new, unknown cyber threats.

2. OBJECTIVE OF THE STUDY

The objective of this paper is to survey various AI/ML techniques used in the literature for the detection of cyber-attacks. It aims to categorize cyber-attacks into three categories based on their impact and point of origin. The three key domains or categories are network-based attacks, host-based attacks, and application-level attacks. This study highlights the need to examine the varying nature of threat vectors and the need for domain-specific detection strategies under these categories. Moreover, a diverse set of publicly accessible datasets is analyzed and assessed that have been utilized by the researchers.. These datasets are compared on the basis of various features, attack types, and their applicability to supervised or unsupervised learning models. The primary goal is to assist researchers in selecting the intelligence strategies, most effective techniques, or a combination of techniques and setting the benchmarks for the best results in this area. It provides a detailed survey of ML and AI techniques followed, including discussions on the extraction of features, training a model, performance evaluation, and data preprocessing (i.e., data preparation, data cleaning, and data transformation). Also, the models ranging from classical ML (Machine Learning) algorithms to advanced DL (Deep Learning) frameworks are considered. Finally, the study identifies key research challenges and gaps, such as limited dataset realism, lack of explainability and transparency in AI models, and the growing threat of adversarial attacks, and it also recommends future directions that include hybrid detection models, explainable AI, and context-aware threat intelligence systems.

3. ORGANISATION OF THE PAPER

This paper is divided into several well-defined sections to facilitate a logical flow of information. It starts with a background section that explains and introduces the most common types of cyber-attacks, i.e., network-based, host-based, and application-level attacks, and the sub-section also includes an outline of the ML techniques that are typically designed for detecting such attacks. Following the introductory background, the paper provides an in-depth examination of the three main categories of cyber-attacks. Each attack category is explored with a detailed analysis focusing on the relevant datasets and various AI/ML techniques that achieve the best results and are most effective for detecting each type of attack. For example, known attack patterns can be detected in the network traffic using supervised learning models, while unsupervised and deep learning models are better suited for identifying anomalies in host behavior and application-layer interactions. Finally, the conclusion and future direction section synthesizes insights across the different categories of attacks, followed by a summary of the key findings. Recommendations for future research address current limitations and are aimed at enhancing the effectiveness of AI-driven cybersecurity solutions.

4. BACKGROUND

With the ever-growing digital landscape and infrastructures, cyber-attacks are becoming a persistent and evolving threat to digital systems. Broadly, these attacks have been classified into three main types based on their point of execution and intended impact: network-based, host-based, and application-level attacks as depicted in FIGURE 1.

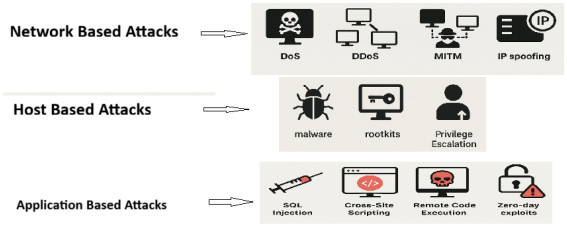

Figure 1: Types of Attacks

This section provides a brief overview of each category and its sub-types. Network-based attacks aim to disrupt communication across networks. Data being transmitted can be intercepted and altered within these networks. Common examples of network-based security threats include DoS (Denial-of-Service), DDoS (Distributed Denial-of-Service), and Man-in-the-Middle (MITM) attacks, as well as IP spoofing, ARP poisoning, and the utilization of botnets to overwhelm systems. A summary of these network-based attacks can be found in TABLE 1. These types of attacks take advantage of vulnerabilities present in routers, switches, firewalls, and other network infrastructure components.

Host-based attacks focus on individual systems or host machines such as personal computers, servers, or virtual machines. Attackers may try to enter the target system and gain unauthorized access. For gaining access to sensitive data and stealing critical information, the attackers may

Table 1: Network -based attack Types

Attack Type	**Definition**	**Typical Scenario**	**Attackerś Goal**	**Technique used**	**Target Location**
DoS	Flooding a system to make services unavailable	During high-traffic times to cause disruption	Disrupt service, extort money, or harm reputation	Massive fake traffic sent from a single source	Web servers, online services
DDoS	Distributed DoS attack using multiple machines	Often timed during events, sales, or launches	Similar to DoS but harder to mitigate due to scale	Botnets flood the system from many infected devices	Network infrastructure, e-commerce sites
MITM	Intercepting communication between two parties	During unsecured communication (e.g., public Wi-Fi)	To eavesdrop, alter or steal sensitive information	Attacker positions themselves between users and the service	Client-server connections, banking sites
Botnets	Network of compromised devices controlled by an attacker	Before launching large-scale DDoS or spam attacks	To control many systems for massive coordinated attacks	Malware infects devices and links them to a central command server	IoT devices, unsecured systems, global endpoints
IP Spoofing	Using a forged IP address to disguise identity	Common during DDoS or MITM attempts	To impersonate or bypass network filters	Modifying IP headers of outgoing packets	Routers, firewalls, networked devices
ARP Poisoning	Corrupting ARP tables to intercept or redirect traffic	On local area networks (LANs)	To intercept data or perform MITM attacks	Sending fake ARP messages linking IP to attacker's MAC address	Internal networks (LAN)

install malware, rootkits, or keyloggers and persist in the system undetected for launching Advanced Persistent Threats (APT). Techniques to escalate system and user privileges are also used to elevate attacker rights within a system. The host-based attack types are discussed and presented in TABLE 2.These attacks are usually carried out with user-level compromise using malicious attachments, infected media, or phishing attempts.

malware, rootkits, or keyloggers, Privilege Escalation are the host based cyber attacks

Table 2: Host -based attack Types

Attack Type	**Definition**	**Typical Scenario**	**Attacker's Goal**	**Technique used**	**Target Location**
Malware	Malicious software installed on a host	Triggered by downloads, email attachments, USB devices	To damage, control, or spy on the host system	Via phishing, drive-by downloads, or infected files	End-user devices, enterprise systems
Rootkits	Tools to hide presence of malicious processes	After gaining admin/root access	To maintain long-term undetected access	Installs deep within OS, bypassing detection tools	Operating systems (Windows/Linux)
Keyloggers	Tools that record keystrokes	During login or sensitive data entry	To steal credentials or personal data	Captures keyboard inputs via software or hardware	Personal computers, ATM systems
Privilege Escalation	Gaining higher system privileges than intended	After initial system compromise	To gain full control or access restricted data	Exploiting OS or app vulnerabilities or misconfigurations	Operating systems, corporate endpoints

Application-level attacks Attacks, including injection attacks, i.e., SQL Injection (SQLi), Cross-Site Scripting (XSS), Remote Code Execution (RCE), and zero-day exploits, exploit vulnerabilities in web applications, APIs, and software services. Such attacks are known as Application-level attacks and are summarized in TABLE 3. They try to find logical flaws in the application code, input validation errors, or unpatched vulnerabilities in the target application and often results in unauthorized access, data exfiltration, or system control.

To detect and mitigate these attacks, AI and ML techniques are increasingly being used because of their adaptability to learn sophisticated, high-dimensional patterns and detect anomalies/outliers in large-scale data. These techniques include supervised learning models for identifying known patterns, unsupervised learning for discovering novel anomalies, and DL models to analyze high-dimensional and temporal data. Reinforcement learning and natural language processing (NLP)

methods are also gaining ground, particularly for adaptive threat response and textual log analysis, respectively. Ensemble learning techniques are often used to merge the strengths of various models, enhancing the robustness and accuracy of detection systems.

Table 3: Application -level attack Types

Attack Type	**Definition**	**Typical Scenario**	**Attacker's Goal**	**Technique used**	**Target Location**
SQL Injection	Injecting malicious SQL code into an application	When input fields are not properly sanitized	To view, modify, or delete database contents	Entering crafted SQL statements into forms or URLs	Web applications with SQL backends
Cross-Site Scripting (XSS)	Injecting malicious scripts into websites	When a site fails to sanitize user input	To hijack sessions or steal cookies	Embedding scripts in input fields or URLs	Web pages, user dashboards
Remote Code Execution (RCE)	Executing arbitrary code on a remote system	When applications have unpatched vulnerabilities	To take control or inject malware	Exploiting flaws to upload/execute code remotely	Web servers, application servers
Zero-day Exploits	Attacks exploiting unknown or unpatched vulnerabilities	Before vendor or defenders are aware of the flaw	To take control before a patch is available	Leveraging undisclosed vulnerabilities in software	Operating systems, applications, browsers

With an increase in the number of cyber-attacks, intrusion detection and prevention of networks have become a major concern. Machine Learning and Deep Learning methods can help in predicting the risk of a cyber-attack and thus detect network intrusions. Over the years, researchers have compared various machine learning and deep learning methods and proposed the ones that showed higher accuracy of detection in different scenarios. A brief summary of various machine learning and deep learning algorithms [4] is given below:

- Supervised learning techniques such as Random Forest, SVM, and XGBoost rely on labeled datasets to classify known attack patterns. These models are ideal for detecting known threats with high accuracy in binary or multi-class classification tasks, commonly implemented in intrusion detection/prevention systems (IDS/IPS) and malware scanners. The typical strategy involves extracting features from labeled data, training a model, and performing classification.
- On the other hand, unsupervised learning techniques like K-Means, DBSCAN, and Isolation Forest work with unlabeled data to identify anomalies and novel attack patterns. These techniques are important for detecting previously unseen threats and understanding botnet behavior. These techniques organize data into clusters and identify deviations from normal

behavior, and generate alerts in case of any anomaly or abnormal behavior. These are best suited for network monitoring and anomaly-based IDS frameworks.

- For analyzing high-dimensional and sequential data, such as system logs and network traffic, Deep learning models like Convolutional Neural Networks, Recurrent Neural Networks, Long Short-Term Memory networks, and Autoencoders are highly effective.
- Reinforcement learning algorithms, including Q-Learning and Deep Q-Networks, enable agents to identify optimal defense strategies through interaction with an environment while receiving feedback in the form of rewards or penalties. This methodology is particularly effective for addressing dynamic threat landscapes and is employed in adaptive systems such as honeypots, firewalls, and software-defined network (SDN) controllers.
- NLP models like BERT, TF-IDF, and Word2Vec help the analysis of unstructured textual data, such as logs, emails, or chat messages. These methods are essential for real-time email and web content filtering, especially for spam and phishing detection, and are commonly integrated into email gateways and Security Information and Event Management (SIEM) tools.
- Lastly, ensemble methods like bagging, boosting, and stacking enhance model reliability by integrating multiple weaker models into a unified decision-making process. These techniques reduce bias and variance.

5. RELATED WORK

Phishing is an attack used by attackers to trick their targets and collect confidential information of the target or to send malware as attachments to the users. Email phishing is the most prevalent cyber threat now a days. Phishing, especially email and URL phishing, has become a very common cyber-attack due to its ease of use and huge success. Despite the high alertness and awareness of the users, it becomes harder for users to be aware of the malicious URLs. Users tend to click on malicious URLs in a hurry. Blocklists and content analysis are used traditionally for detection of phishing attacks. But both the techniques require time-consuming human verification. Authors in their paper tried to detect fraudulent URLs using predictive. Splunk platform was used to train the ML model. SVM and Random Forests algorithms were trained using malicious and benign datasets and evaluated the algorithms' performance with precision and recall, reaching up to 85% precision and 87% recall in the case of Random Forests while SVM achieved up to 90% precision and 88% recall using only descriptive features [5]

In another paper, the authors [6] have used open-source intelligence (OSINT) tools and machine learning (ML) models for detection of phishing attacks on multilingual datasets. The study highlights that ML models that are majorly trained on English data and therefore have limitations. A total of 17 features were extracted using Nmap and Harvester tools. The features like domain names of the hosts, open TCP and UDP ports and IPs were included primarily to elevate accuracy of attack detection. Different classification algorithms such as SVM, XGBoost, DT (Decision Trees), RF (Random Forest) and Multinomial Naïve Bayes were used for experiments. Random Forest was found to be the best algorithm with 97.37% accuracy for both the datasets used for experiments (English and Arabic). The results show that phishing emails can be detected with higher accuracy when ML models are combined with OSINT tools and have the potential to be used effectively for detection across different languages.

A new cyber security framework utilizing Federated Learning (FL) for IoT networks is proposed by the authors [7]. It employs Recurrent Neural Networks (RNNs) for anomaly detection and real-time threat response while preserving privacy by training models locally on edge devices. These models are securely aggregated using homomorphic encryption, enabling collaborative learning without exposing sensitive data. The system demonstrates a 20% reduction in resource consumption and achieves 98% accuracy in detecting advanced threats like DDoS attacks.

Recent research highlights the increasing threat of False Data Injection Attacks (FDIAs) in smart grids. To tackle this problem, researchers have combined federated learning with transformer models and the Paillier cryptosystem. This approach improves detection and protects sensitive data [8]. Researchers create synthetic datasets using GANS to mimic potential attack paths and analyze data from system logs and incident reports using NLP [9].

As IOT devices are small in size and have limited processing capabilities, authors in their work [10] have proposed a lightweight ML based ensemble methods for multi-class attack detecting in IoT networks. The CICIoT 2023 dataset was chosen to assess the various ML techniques and to find the best one that achieves high accuracy and efficiency. A total of 34 distinct attack types are mentioned in this dataset which are further categorised in 10 groups. Among all classifiers, decision tree classifier was the best with 99.56% accuracy and 99.62% F1 score which is impressive. Following closely, the Random Forest (RF) model achieved an accuracy of 98.22% and an F1 score of 98.24%. These findings underscore the effectiveness machine learning methods in accurately and reliably detecting threats within high-dimensional datasets.

Hari Gonaygunta [11] proposed in his paper that an effective ML technique is needed to detect cyberattacks which can minimize false positives as there is a huge possibility of false-positive detections in large set ups. The algorithm should be able to analyze large amount of security and infrastructure logs in order to identify cyber-attacks quickly and automatically. He proposed Logistic regression as an effective ML technique to detect the cyber-attacks in big set ups where the probability of false positives is more.

Mehdi et al. [12] created a dataset of legitimate and phishing emails to train models for detecting adversarial phishing using text attacks. Their experiments demonstrated improved accuracy and F1 scores, although black box attack methods showed limited accuracy gains. Using a K-Nearest Neighbor approach, they achieved 94% accuracy in classifying adversarial text.

It is difficult to classify the intrusions and cyber-attacks due to a wider range and unknown attacks types. Further, the network connections which are later detected as intrusions, often start with benign behavior and makes it harder to be detected as malicious till it is too late. Due to their initial benign behavior, the traditional classification techniques are not able to classify and detect the intrusions/ cyber-attacks accurately due to class imbalance and classifier biasness towards benign behavior. This may lead to many of the attacks being bypassed and undetected. Dong et al [13], designed a system named DeepIDEA. It does intrusion detection and classification using a loss function that is based on sharing of attack information to enhance robustness. It eliminates the classifier bias and achieves high detection accuracy on imbalanced data. It is done by shifting the decision boundary away from benign classes and taking it towards the attack classes. The proposed system uses a loss function that penalize intrusion mis-classification more than attack mis-classification. The results show high detection accuracy and best class-balanced accuracy on different datasets using different approaches.

The cybersecurity has become a challenging task as the attackers now use new tactics that are sophisticated. For example, polymorphism can be used to change the attack patterns continuously and design new attacks. Novel attacks are difficult to detect using signature-based detection methods with a higher rate of false positives. These attacks can evade the detection models for long and completely go undetected. Chakraborty et al [14] in their study, proposed a rule-based deep neural network technique for detecting novel attacks. The proposed model has an accuracy of more than 99% and tries to balance between attack detection, false positives and false negatives for novel attacks. This technique able to classify and identify different attacks efficiently while maintaining security and privacy of IOT devices which is the key factor in these networks.

The author [15] in their research highlights the power of Generative AI in designing sophisticated cyber threats and its use in developing strong and robust Generative AI-driven defences.

A recent study [16] demonstrates that Generative Artificial Intelligence (GAI) can predict cyber-attacks and zero-day vulnerabilities with high accuracy, achieving 87% for threat prediction and 80% for vulnerability detection. It reports a 98% detection rate for known threats and 92% for unknown threats, with low false positive rates. GAI effectively discovers vulnerabilities, simulates realistic attacks, and automates threat responses.

Fattahi J. [17] in his research, reviewed the various ML and DL approaches used in the field of cybersecurity focusing on their advantages, drawbacks and possibilities. The AI techniques for intrusion detection, malware classification and resilience enhancement are discussed in detail. Network attacks can be detected by analyzing network traffic data and training neural networks. But data sets available for public networks have limited variations in sample data. Also, the data is unbalanced with respect to malicious and benign samples. The author used [18] protocol fuzzing, a technique to automatically generate network data with high-quality.The fuzzed data was then used to train DL models and to uncover the network attacks. The results show that the data generated with fuzzing simulates real real-world data and DL models trained on fuzzy dataset can successfully detect real network attacks.

In recent years, new network attacks have been designed that exploit the flaws in program or application logic [19]. The attacks have become a major security concern as these can easily bypass the IDS based on signature-based detection mechanisms, as these attacks do not have distinguishing signatures. The results show that data samples generated using protocol fuzzing and DL models trained on this fuzzed data, can detect these attacks successfully.

Cyber-physical systems (CPS), especially CPS-IoT, are susceptible to DDoS attacks, which are often launched through TCP SYN from IoT subsystems aimed at cloud-based servers. A research study by [20] evaluated different machine learning algorithms for detecting DDoS in CPS-IoT, utilizing an unsupervised K-Means algorithm for data labeling followed by various supervised models. The combined model achieved an accuracy of 100% with no false positives, whereas the other models maintained an accuracy rate exceeding 94%.

The authors [21] utilized three machine learning algorithms—extreme gradient boosting (XGB), multilayer perceptron (MLP), and random forest (RF)—to detect DoS/DDoS attacks on IoT devices. They employed Particle Swarm Optimization (PSO) for feature selection on the CICIoT2023 dataset, achieving an impressive accuracy, precision, recall, and F1 score of 99.93% with the XGB

model, which also had a shorter execution time (491.023 seconds) compared to Recursive Feature Elimination (RFE) and Random Forest Feature Importance (RFI) methods.

Similar research by [22] evaluated models such as XGBoost, K-Nearest Neighbours, Stochastic Gradient Descent, and Naïve Bayes for detecting DDoS attacks in IoT networks. They discussed the strengths and weaknesses of each model and concluded that ML techniques can provide adaptive, efficient, and reliable DDoS detection.

Becker et al in their study [23], compare eleven ML models for attack detection on edge IoT device. The dataset with fourteen different attacks was used to find the best ML technique.

The study in [24] used popular datasets like CIDDS-001, UNSW-NB15, and NSL-KDD for a comprehensive analysis of ML classifiers. They also used Raspberry Pi to find the best classifier in terms of response time for specific IoT hardware. They suggested using ensemble learning and statistical assessment of the classifier's performance for further experiments and research in cybersecurity and developing a strong IDS.

Authors in [25] reviewed intrusion detection in IoT systems using various machine learning approaches, suggesting that these techniques can enhance IoT device security. [26] evaluated seven ML algorithms on the new Bot-IoT dataset, finding improved results with the extracted features.

In [27], the author proposed a deep learning model that integrates ResNet and EfficientNet for intrusion detection in IoT systems, showing significant improvements in true positive and false positive rates compared to the LSTM model.

In their study [28], the authors worked on detecting cyber-attacks on financial institutions using various ML algorithms like KNN, RF and SVM. SVM achieved the highest accuracy at 99.5% while in another research [1], RF outperformed KNN, SVM, and DT models in IoT intrusion detection, achieving an accuracy of 99.72%.

6. DATASETS TO DETECT CYBER ATTACKS

6.1 Network Based Cyber Attacks

Datasets

A variety of publicly available datasets have been used for evaluating AI/ML-based network intrusion detection systems.The KDD CUP 99 [29] dataset, one of the earliest benchmarks, includes 41 features spanning basic, content-based, and traffic statistics, labeled for attacks such as DoS, Probe, R2L, and U2R. However, its limitations in redundancy and outdated traffic led to the development of the NSL-KDD [30] dataset, which retains the same feature structure but with improved class balance and reduced duplicates.

The UNSW-NB15 dataset [31] introduces a more modern set of 49 flow-based features and includes a diverse set of attack categories like Exploits, Fuzzers, DoS, and Backdoors. It is more representative of real-world traffic and is widely used in evaluating supervised ML models. Some other

important datasets, CIC-IDS2017 [32] , contain more than 80 extracted features from network flows using CICFlowMeter, representing attacks such as Brute-force, Botnets, DDoS, and Infiltration, also offer a balanced blend of normal and malicious traffic with accurate timestamps, durations, and payload data.

Other specialised datasets, such as CIC-DDoS 2019 [33], concentrate on DDoS-specific attacks, such as variants based on HTTP, TCP, and UDP, whereas CTU-13 [34], comprises netflow data from 13 botnet scenarios that are classified as either normal or botnet traffic. IoT-focused datasets such as TON_IoT [35], and BoT-IoT [36], are designed to evaluate cybersecurity threats in smart environments, containing telemetry and network flow data. These datasets offer high volume and feature diversity, suitable for deep learning and anomaly detection approaches.

Additionally, the other datasets, like CIC-FlowMeter [37], which acts as a flow feature extractor, and IDS2018 (CSE-CIC) [38], an enhanced version of CICIDS2017 [32], with newer attack scenarios, further enrich and ensure the benchmark landscape. These datasets have made it possible to categorically identify different supervised and unsupervised machine learning models for different attack scenarios and traffic types.
A comprehensive summary of the datasets that are frequently used for network-based cyber attack detection is given in TABLE 4. The details, such as the Dataset names, Features, Labels, Attack Types, etc. are listed in the table.

ML Techniques Applied in Network-Based Attack Detection
Using various preprocessing and feature engineering methods, numerous machine learning models have been developed and tested on these datasets.

Random Forests have demonstrated consistent performance, reaching up to 91% accuracy on datasets such as NSL-KDD [30], particularly when paired with label encoding and recursive feature elimination (RFE).

Dimensionality reduction for Support Vector Machines (SVM) often uses Z-score normalization and Principal Component Analysis (PCA), which shows effective results on complex datasets such as UNSW-NB15 [31], with 82% accuracy. Similarly, for Decision Trees (CART) leverage entropy-based feature selection and achieve high accuracy on simpler datasets like KDD99 [29]. With feature importance ranking and min-max scaling, ensemble methods like XGBoost are especially effective on rich datasets like CIC-IDS2017 [32], achieving up to 96% accuracy. Meanwhile, Naive Bayes, although simpler, performs reasonably well (80%) on balanced datasets like NSL-KDD [30] when continuous features are discretized. On datasets like BoT-IoT [36], applying instance-based methods such as k-Nearest Neighbors (k-NN), combined with feature scaling and distance-based voting, has shown over 85% accuracy. With Neural network-based models like Multilayer Perceptron (MLPs) and Long Short-Term Memory (LSTM) networks, normalized inputs and either autoencoder-based or temporal feature extraction achieve high detection rates i.e., above 95% particularly on large-scale datasets like CICIDS2017 [32], and CTU-13 [34]. In terms of unsupervised, Autoencoders trained on normal traffic are used to detect anomalies via reconstruction error, achieving F1-scores between 90%–93% on datasets like CTU-13 [34], and BoT-IoT [36]. And lastly, Convolutional

Table 4: Commonly Used Datasets for Network-Based Cyber Attack Detection

Dataset Name	**Features**	**Labels**	**Attack Types**	**Source**	**Size**
NSL-KDD [30]	41	5 classes (Normal, DoS, Probe, R2L, U2R)	DoS, Probe, R2L, U2R	Downloadable (CSV)	70MB
KDD CUP 99 [29]	41	5 classes (same as NSL-KDD)	DoS, Probe, R2L, U2R	Downloadable (CSV)	4GB
CICIDS2017 [32]	80+	Multi-class (15+ attack types)	Brute Force, DoS, Botnet, DDoS, Infiltration, etc.	Downloadable (CSV, PCAP)	40GB
UNSW-NB15 [31]	49	10 attack types + Normal	Fuzzers, Backdoors, Exploits, Generic, Reconnaissance, etc.	Downloadable (CSV)	2GB
CTU-13 [34]	Varies (depends on capture)	Binary (Normal/Botnet)	Botnet (13 scenarios),	Normal Web-based	25GB
BoT-IoT [36]	40+	Binary (Attack/Normal)	DDoS, DoS, Reconnaissance, Information Theft	Downloadable (CSV, PCAP)	16GB
CSE-CIC-IDS2018 [38]	80+	Multi-class (15 attack types)	DDoS, Brute Force, SQL Injection, Botnet, etc.	Downloadable (CSV, PCAP)	60GB
NF-ToN-IoT-v2 [39]	43	Multi-class	DoS, DDoS, Ransomware, Backdoor, etc.	Downloadable (CSV, PCAP)	80GB
CIC-IDS2019 [33]	80+	Multi-class	PortScan, DoH, Infiltration, Web Attacks	Downloadable (CSV, PCAP)	40GB
ISCX IDS 2012 [40]	20	Binary (Attack/Normal)	HTTP DoS, DDoS, Brute Force, Infiltration	Downloadable (PCAP, CSV)	30GB
CIC-DDoS 2019 [33]	88+ (flow and packet-level)	Multi-class	TCP, UDP, HTTP DDoS attacks	CIC, Canada	50 GB, 12 attack types

TON_IoT [35]	IoT telemetry + network flow	Multi-class	DDoS, Backdoor, XSS, Reconnaissance	UNSW Canberra	22M records
CIC-FlowMeter [37]	Extracted features from raw PCAPs	Depends on dataset	Used for multiple CIC datasets	CIC	Varies

Neural Networks (CNNs), when reshaped for spatial pattern recognition, work well on flow-level features from datasets like CIC-DDoS2019 [33].

The list of commonly used ML techniques for the detection of network attacks is presented in TABLE 5 that summarizes the preprocessing, feature extraction, dataset, and accuracy achieved.

6.2 Host Based Cyber Attacks

Datasets

Data collected directly from endpoints, including logs, system calls, audit trails, and user behavior is used in Host-based intrusion detection. A number of benchmark datasets have been developed to support research in this domain. A foundational dataset is DARPA BSM, which consists of Basic Security Module (BSM) audit logs collected from Solaris systems. Though dated, it was critical in early system-call-based intrusion detection research. Building upon it, the UNM [41] Dataset (from the University of New Mexico) focuses specifically on system call traces from programs such as sendmail, lpr, and xlock. In order to facilitate early anomaly detection models, it offers labelled traces for both normal and anomalous behaviour. A related and more granular dataset is the ADFA-LD (Australian Defence Force Academy Linux Dataset), which offers contemporary Linux-based system call sequences and includes zero-day attacks, making it suitable for modern host-based anomaly detection systems. From KDD, the Syscall Dataset (part of KDD98) represents system call sequences extracted from program execution traces and has been frequently used to evaluate HIDS models using sequence-based analysis. More recently, the host logs integrated from the NGIDS-DS [42] (Next-Generation Intrusion Detection System Dataset) also include command histories and security-relevant system behaviors, which offer a modern reflection of endpoint behavior in enterprise environments. In the Windows ecosystem, Windows HIDS datasets, like Procmon-based logs or the HUNT-HIDS Dataset, collect registry activities, file access patterns, and process creations, supporting the detection of ransomware and malware via behavioral analysis. For evaluating deep learning models on high-dimensional time-series data, these datasets are very useful. Further, TON_IoT [35] Windows Logs integrate telemetry and event-based logs from IoT-based Windows endpoints, bridging the gap between traditional HIDS and smart environments. These datasets are notable for containing realistic attack scenarios with labeled sequences across authentication, registry, and service-level activities. These datasets range from short system call sequences to rich log data offering diverse feature modalities. Thus the datasets form the backbone of host-based cyber attack detection using both classical and deep learning models. TABLE 6 consists of commonly used datasets for Host-Based Cyber Attack Detection. These datasets are

Table 5: ML Techniques for Network-Based Attack Detection

ML Technique	**Pre-processing**	**Feature Extraction**	**Dataset**	**Accuracy Achieved**
Random Forest	Normalization, label encoding	Recursive Feature Elimination (RFE)	NSL-KDD [30]	~85–91%
SVM	Z-score normalization	PCA	UNSW-NB15 [31]	~82%
Decision Trees (CART)	One-hot encoding	Gini/Entropy-based selection	KDD99 [29]	~90–92%
XGBoost	Min-Max Scaling	Feature importance ranking	CIC-IDS2017 [32]	~95–96%
Naive Bayes	Discretization of continuous features	None	NSL-KDD [30]	~75–80%
k-NN	Scaling	Distance-based weighting	BoT-IoT [36]	~87%
ANN (MLP)	Normalization	Autoencoder for dimensionality	CIC-IDS2017 [32]	~97%
LSTM	Sequence padding, scaling	System call or flow time-series	CTU-13 [34], CICIDS2017 [32]	~94–98% (depending on data)
CNN	Reshaping for 2D input	Temporal + spatial filters	CIC-DDoS2019 [33]	~95%
Autoencoder (Unsupervised)	Min-Max, anomaly score thresholding	Reconstruction error	CTU-13 [34], BoT-IoT [36]	~90–93% (Anomaly F1-score)

usually based on system call traces, audit logs, or host behaviour data.

ML Techniques Applied in Host-Based Attack Detection
A wide variety of machine learning techniques have been applied to host-based intrusion detection tasks as given in TABLE 7, often involving preprocessing steps such as sequence encoding, windowing, and log parsing. One of the most common methods is Hidden Markov Models (HMMs), which are adept at modeling system call sequences as probabilistic state transitions. These are typically trained on normal data and used to detect anomalies based on low-likelihood sequences, achieving high detection accuracy (often above 90%) on datasets like UNM [41] and ADFA-LD [43].

For classification tasks, Support Vector Machines (SVMs) and Random Forests are widely used on feature-engineered representations of logs or command histories. Feature extraction may include frequency-based metrics, token embeddings (e.g., TF-IDF), or manually crafted statistical attributes. These models achieve around 85–92% accuracy on datasets such as NGIDS-DS [42] and Syscall traces.

Due to the ability of model temporal dependencies in command sequences or system calls, there are LSTM networks which have become popular. When trained on sequence-encoded system calls (e.g., using one-hot or word2vec embeddings), LSTMs can achieve over 95% detection rates on datasets like ADFA-LD [43]. Similarly, CNNs, when applied to embedded syscall patterns or converted log images, capture local spatial features in data and yield competitive results with minimal feature engineering.

By learning to reconstruct normal patterns - autoencoders, both feedforward and sequence-based (like LSTM autoencoders), are used for anomaly detection, and reconstruction errors that are significant indicate possible attacks. When applied on datasets such as UNM [41] and TON_IoT [35] Windows Logs, these models achieve F1-scores of more than 90%. To capture both local and global dependencies in command sequences and logs additionally, transformer-based models have begun to emerge in these spaces, leveraging self-attention mechanisms. These approaches are particularly useful in Windows-based HIDS datasets where events are contextually rich and dispersed over time. Preprocessing steps often include log normalization, system call indexing, and sliding-window segmentation. For example, fixed-size windows of system calls are used to train sequential models like LSTMs or HMMs. In log-based HIDS, parsing into structured key-value pairs followed by vectorization enables classical ML models to perform effectively. Overall, host-based intrusion detection benefits from a combination of symbolic sequence modeling, temporal deep learning, and statistical anomaly detection, each suited to the specific nature and granularity of the underlying data.

6.3 Application-Level Cyber Attacks

Datasets

Table 6: Commonly Used Datasets for Host-Based Cyber-Attack Detection

Dataset Name	**Features**	**Labels**	**Attack Types**	**Source**	**Size**
ADFA-LD [43]	System calls (raw & sequences)	Binary (Normal/Attack)	Mimicry, Reverse Shell, Add User, Java Meterpreter	UNSW Canberra	~ 1MB
UNM [41] Dataset	System call sequences	Binary (Normal/Intrusion)	Buffer overflow, input validation, Trojan	Univ. of New Mexico	500KB
DARPA BSM [44]	Audit logs (BSM format)	Multi-class	Host-based Unix intrusions	MIT Lincoln Labs	~6GB
CERT Insider Threat [45] (R4.2)	Host activity logs, email, web usage	Multi-class or insider anomaly	Masquerade, data exfiltration, email misuse	Carnegie Mellon	~30GB
Two sigma Linux Dataset [45]	System call logs, command traces	Multi-class(Normal/Attack)	Malware, privilege escalation, process injection	Two Sigma	8GB
HDFS Log Dataset [46]	Log entries from HDFS clusters	Anomalies marked (timestamps)	Host event anomalies	Open-source Hadoop	1.5GB
NGIDS-DS [42]	Host features (system calls, files)	Binary	Various Linux-based attacks	NGIDS Research Team	500MB
Syscall-AI Dataset [47]	System call arguments, PID, etc.	Multi-class	Host-based malware behaviors	GitHub/ Academic	250MB
TUIDS Syscall Dataset [48]	System call sequences from Linux	Binary (Normal/Attack)	Code injection, buffer overflow, shell spawning	Turkish Univ. Cyber Lab	1GB
KDD99 [29] (host logs)	System-level features derived from logs	Normal, DoS, R2L, U2R, Probe	Multiple attack types (U2R, R2L, Probe, DoS)	DARPA / UCI	4,900,000 records 1GB

Table 7: ML Techniques for Host-Based Attack Detection

ML Technique	**Pre-processing**	**Feature Extraction**	**Dataset**	**Accuracy Achieved**
Random Forest (RF)	Feature normalization, one-hot encoding	Call frequency, path features	ADFA-LD [43]	98.7%
SVM	Scaling, feature selection	n-gram of system calls	UNM [41] Dataset	96.3%
RNN (LSTM)	Sequence padding, normalization	Sequential modeling of system call logs	ADFA-LD [43]	98.9%
Decision Tree	Discretization, binning	Statistical features from logs	KDD99 [29] (host)	92.5%
Naive Bayes	Encoding, frequency	count Bag-of-words from system call traces	UNM [41] Dataset	89.4%
XGBoost	Label encoding, regularization	Behavioral patterns (registry/process access)	NGIDS-DS [42]	97.6
CNN	Embedding sequences	System call traces as 2D arrays	DARPA BSM [44]	94.2%

Cyber attacks are specifically targeted at vulnerabilities in web applications, software services, and user-facing APIs, particularly at the application level. Unlike network- and host-based attacks, which focus on packets or system-level behavior, application-layer threats exploit flaws such as SQL Injection (SQLi), Cross-Site Scripting (XSS), Remote Code Execution (RCE), and directory traversal. To support machine learning-based detection of such attacks, researchers have developed datasets rich in HTTP traffic, user requests, and log events.

TABLE 8 lists commonly used datasets for Application-Level Cyber Attack Detection, especially those that focus on web applications, APIs, databases, and application-layer traffic. CSIC 2010 HTTP Dataset, released by the Spanish Research National Council is a widely referenced dataset in this domain. It contains thousands of legitimate and malicious HTTP GET and POST requests to a simulated e-commerce application. Each request is well-structured and labelled, enabling supervised learning for web attack detection. These datasets consist of several application-level attack types like SQL injection, cross-site scripting and buffer overflow to name a few.

There is another commonly used dataset known as Web Application Attack Dataset (WAAD). This dataset collects the HTTP request logs generated by using tools like Metasploit and OWASP ZAP for multiple attack scenarios and also records real HTTP traffic patterns with attack payloads, making it ideal for training anomaly detection or binary classifiers.

There is another important and very useful dataset is ISCX HTTP. This dataset is extracted from the ISCX IDS 2012 [40] traffic and applies to provide HTTP-specific sessions extracted from real and synthetic attack scenarios and includes unauthorized login attempts, SQL injection attacks, and various botnet-driven application-layer scans. And it also allows for temporal modeling of request behaviors across user sessions, as it is flow-based and session-aware.

For advanced Web Application Firewall (WAF) research, the CIC-DDoS2019 [33] Dataset also contains traces of HTTP-based flooding and volumetric attacks targeting application services. Though primarily used for DDoS detection, its logs are useful for identifying abusive request patterns and application misuse.

In addition, the TON_IoT [35] HTTP Logs and UNSW-NB15 [31] App Layer Logs bring attention to IoT and hybrid network environments where HTTP and MQTT requests are logged. These datasets reflect modern API-based attacks in smart home and industrial control systems. Collectively, these datasets enable researchers to study and classify application-level attacks by providing structured, labeled, and timestamped web traffic. The granularity of HTTP headers, URL parameters, and payload content makes them highly effective for both rule-based filtering and machine learning-based pattern recognition.

ML Techniques for Application-Level Intrusion Detection

Application-layer attack detection typically deals with high-level structured input such as URLs, parameters, cookies, and user-agent strings. A brief summary of the ML techniques used for application-

Table 8: Commonly Used Datasets for Application level Attack Detection

Dataset Name	**Features**	**Labels**	**Attack Types**	**Source**	**Size**
CSIC 2010 HTTP [49]	HTTP request fields (GET, POST, etc.)	Binary (Normal/Anomalous)	XSS, SQLi, buffer overflow, remote file inclusion	Spanish CSIC	$\sim 300MB$
UNSW-NB15 [31] (App subset)	Application payloads (HTTP, FTP, DNS, etc.)	Multi-class	SQLi, shellcode, exploits	UNSW Canberra	$\sim 2.5GB$
CICIDS2017 [32] 2017 (App subset)	Network+App features (protocols, payloads)	Multi-class	Brute force, web attack, SQLi, DoS	Canadian Institute for Cybersecurity	$\sim 15GB$
HTTP DATASET CSIC 2010 HTTP [49]	HTTP parameters, URLs, headers	Binary	SQLi, XSS, buffer overflow	CSIC	$\sim 400MB$
OWASP WebGoat Logs [50]	Web server logs, interaction traces	Labeled manually	All OWASP Top 10 (XSS, SQLi, IDOR, CSRF, etc.)	OWASP	Varies
PT 2021 Web Attack Dataset [51]	Web traffic, logs, parameters	Multi-class	Web shell, RCE, path traversal, SQLi	Positive Technologies	$\sim 1GB$
WAIA Dataset [52]	Web API logs + parameters	Multi-class	API fuzzing, SQLi, authentication bypass	WAIA Research Lab	$\sim 800MB$
ModSec Web Attack Logs [53]	ModSecurity WAF logs	Binary (Normal/Blocked)	Web exploits (XSS, LFI, RFI, etc.)	Open Source WAF Logs	$\sim 2GB$
Imperva Web Attack Dataset [54]	HTTP requests, headers, payloads	Multi-class	SQLi, XSS, command injection	Imperva	$\sim 5GB$
ISCX-WEB [55]	HTTP traffic, URLs, user-agent, content types	Benign / Malicious	Web attacks: SQLi, XSS, DoS	ISCX (CIC, Canada)	0.5 GB
WebShell Dataset [56]	Web logs + PHP shell detection indicators	WebShell / Legitimate	WebShells (China Chopper, WSO, etc.)	Chinese Cybersecurity Research	$\sim 0.2GB$

level attack detection is given in TABLE 9. Preprocessing often involves parsing and tokenizing HTTP requests, encoding categorical fields, and normalizing payload content. The bag-of-words model and TF-IDF vectorization are common methods for converting HTTP request data into numerical features suitable for classifiers like Logistic Regression, Naive Bayes, and Random Forests. These models can achieve detection accuracies of more than 90% on datasets such as CSIC 2010 HTTP [49] and ISCX HTTP. More recent efforts apply deep learning techniques to raw or embedded request sequences. For example, CNNs have been trained on character-level representations of URLs and payloads to detect obfuscated attacks like encoded SQLi or script injections. These models excel at identifying local patterns of malicious tokens and achieve high precision (often >95%) on datasets like WAAD. And to model sequences of user requests over time - LSTM networks and GRUs are employed, which helps to detect slow and stealthy attacks such as low-and-slow SQLi or logic bombs hidden across multiple HTTP steps. This temporal modeling is especially valuable when sessions are segmented by user IP or session ID. By learning the manifold of legitimate requests, Autoencoders and Variational Autoencoders (VAEs) are applied for anomaly detection. These models help to reconstruct normal traffic with low error, while anomalous inputs (such as attacks) produce high reconstruction errors. Such models yield F1-scores above 90% when used on normalized CSIC payloads or structured HTTP logs from TON_IoT [35].

Advanced models, such as Transformers, are beginning to gain traction in application-level detection due to their ability to process long payloads and capture complex attention patterns across headers, cookies, and parameters. Pretrained language models like BERT and RoBERTa, fine-tuned on labeled attack payloads, have demonstrated impressive performance in classifying obfuscated XSS or SQLi vectors.

Preprocessing steps may also include payload decoding, feature hashing, and session reconstruction to group logically related HTTP requests. And expression filtering and heuristic analysis, particularly in WAF pipelines, are further augmented with attack classification models.

In conclusion, temporal or contextual modelling, intelligent payload encoding, and structured parsing are all combined in application-level attack detection for secure web-facing systems. The rich semantics of application-layer data allow for expressive machine learning models capable of detecting both known and zero-day web-based attacks.

7. CONCLUSION AND FUTURE DIRECTIONS

This review of the literature has examined a variety of AI and ML methods used in network, host, and application layer cyberattack detection. Some traditional machine learning algorithms provide reliable classification when supported by high-quality, labelled datasets; these ML algorithms are Random Forest, SVM, and Decision Trees. Further, real-time monitoring and behavourial analysis has been demonstrated using deep learning approaches which address the scenarios having complex, temporal or high-dimensional data. Several deep learning approaches, like CNNs and LSTMs have been used.

Table 9: ML Techniques for Application-Level Attack Detection

ML Technique	**Pre-processing**	**Feature Extraction**	**Dataset**	**Accuracy Achieved**
Random Forest (RF)	URL tokenization, content cleaning	N-grams of request parameters	CSIC 2010 HTTP [49]	99.2%
SVM	Text normalization	Header field vectorization	ISCX-WEB [55]	97.6%
CNN	Encoding of request content	1D/2D Convolutions on sequence embeddings	CIC-IDS2017 [32]	98.4%
XGBoost	L log parsing, one-hot encoding	Statistical HTTP feature aggregation	UNSW-NB15 [31]	96.1%
LSTM	Sequence normalization	Temporal modeling of request behavior	CSIC 2010 HTTP [49]	98.7%
Naive Bayes	Token-based parsing	Term frequency (TF), presence of keywords	WebShell Dataset [56]	94.8%
Autoencoder (AE)	Scaled feature vectors	Unsupervised reconstruction error detection	CIC-IDS2017 [32]	97.3%

Even with these achievements, several challenges remain the same, like there being many existing datasets which are outdated or lack diversity, limiting the generalizability of trained models. And concerns were raised about confidentiality, transparency and trust, especially in critical decision-making contexts, due to the black-box nature of deep learning models. In addition, adversarial attacks pose a serious threat by exploiting model vulnerabilities.

Moreover, future research must focus on creating more realistic and diverse benchmark datasets, which helps to develop interpretable AI and ML models, and should integrate hybrid detection architectures that combine signature-based and anomaly-based techniques. Furthermore, self-supervised learning and transformer-based models have the potential to comprehend intricate attack patterns using little labelled data. Additionally, privacy-preserving techniques like federated learning and homomorphic encryption can enable collaborative learning across organizations without exposing sensitive data.

By addressing the roots of these areas, the cybersecurity community can move towards building more robust, scalable, and explainable AI-driven detection systems capable of defending against the dynamic landscape of cyber threats and risks.

References

[1] Avci I, Koca M. Cybersecurity attack detection model, using machine learning techniques. Acta Polytech Hung. 2023;20:29-44.

[2] Islam MR, Nasiruddin M, Karmakar M, Akter R, Khan MT, et al. Leveraging advanced machine learning algorithms for enhanced cyberattack detection on us business networks. J Bus Manag Stud. 2024;6:213-224.

[3] Dina AS, Manivannan D. Intrusion detection based on machine learning techniques in computer networks. Internet Things. 2021;16:100462.

[4] Xin Y, Kong L, Liu Z, Chen Y, Li Y, et al. Machine learning and deep learning methods for cybersecurity. IEEE Access. 2018;6:35365-35381.

[5] Christou O, Pitropakis N, Papadopoulos P, McKeown S, Buchanan WJ. Phishing url detection through top-level domain analysis: A descriptive approach. Arxiv preprint arxiv: https://arxiv.org/pdf/2005.06599

[6] An P, Shafi R, Mughogho T, Onyango OA. Multilingual email phishing attacks detection using OSINT and machine learning. Arxiv preprint arxiv: https://arxiv.org/pdf/2501.08723.

[7] Rahmati M. Federated learning-driven cybersecurity framework for iot networks with privacypreserving and real-time threat detection capabilities. 2025. Arxiv preprint arxiv : https://arxiv.org/pdf/2502.10599.

[8] Li Y, Wei X, Li Y, Dong Z, Shahidehpour M. Detection of false data injection attacks in smart grid: a secure federated deep learning approach. IEEE Trans Smart Grid. 2022;13:4862-4872.

[9] Ramya P, Guntupalli HC. Advanced cyber attack detection using generative adversarial networks and nlp. J Cybersecurity Inf Manag. 2024;14:161-172.

[10] Alve SR, Mahmud MZ, Islam S, Chowdhury MA, Islam J. Smart iot security: lightweight machine learning techniques for multi-class attack detection in iot networks.2025. Arxiv preprint arxiv: https://arxiv.org/pdf/2502.04057.

[11] Gonaygunta H. Machine learning algorithms for detection of cyber threats using logistic regression. Department of Information Technology, University of the Cumberlands; 2023.

[12] Mehdi Gholampour PM, Verma RM. Adversarial robustness of phishing email detection models. In: Proceedings of the 9th ACM international workshop on security and privacy analytics. New York, USA: ACM; 2023:67-76.

[13] Dong B, Wang HW, Varde AS, Li D, Samanthula BK, Sun W et al. Cyber intrusion detection by using deep neural networks with attack-sharing loss. In: 5th International Conference on Big Data Intelligence and Computing (DATACOM). IEEE; 2019:9-16.

[14] Chakraborty S, Pandey SK, Maity S, Dey L. Detection and classification of novel attacks and anomaly in iot network using rule based deep learning model. SN Comput Sci. 2024;5(8):1056. doi: 10.1007/s42979-024-03429-5.

[15] Blake H. Generative ai in cyber security: new threats and solutions for adversarial attacks.2024.

[16] Chaganti KC. Leveraging generative ai for proactive threat intelligence: opportunities and risks. Authorea preprints.2024.

[17] Fattahi J. Machine learning and deep learning techniques used in cybersecurity and digital forensics: a review.2024. Arxiv preprint arxiv: https://arxiv.org/pdf/2501.03250.

[18] Zou Q, Singhal A, Sun X, Liu P. Deep learning for detecting network attacks: an end-toend approach. In: Barker, K., Ghazinour, K. (eds) Data and Applications Security and Privacy XXXV. DBSec . Lecture Notes in Computer Science. Springer, Cham. 2021;12840:221-234.

[19] Zou Q, Singhal A, Sun X, Liu P. Deep learning for detecting logic-flaw exploiting network attacks: an end-to-end approach. J Comput Secur. 2022;30:541-570.

[20] Machaka P, Ajayi O, Kahenga F, Bagula A, Kyamakya K. Modelling ddos attacks in iot networks using machine learning. In: International Conference on Emerging Technologies for Developing Countries. Springer; 2022:161-175.

[21] Alabdulatif A, Thilakarathne NN, Aashiq M. Machine learning enabled novel real-time iot targeted dos/ddos cyber attack detection system. Comput Mater Continua. 2024;80:3655-3683.

[22] Shakya S, Abbas R. A comparative analysis of machine learning models for ddos detection in iot networks. Arxiv preprint arxiv: https://arxiv.org/pdf/2411.05890.

[23] Becker E, Gupta M, Aryal K. Using machine learning for detection and classification of cyber attacks in edge iot. In: IEEE International Conference on Edge Computing and Communications (EDGE). IEEE; 2023:400-410.

[24] Verma A, Ranga V. Machine learning based intrusion detection systems for iot applications. Wirel Pers Commun. 2020;111:2287-2310.

[25] Kikissagbe BR, Adda M. Machine learning-based intrusion detection methods in iot systems: a comprehensive review. Electronics. 2024;13:3601.

[26] Alsamiri J, Alsubhi K. Internet of things cyber attacks detection using machine learning. IJACSA. 2019;10.

[27] Kodyš M, Lu Z, Fok KW, Thing VL. Intrusion detection in internet of things using convolutional neural networks. In: 18th International Conference on Privacy, Security and Trust (PST). IEEE; 202:1-10.

[28] Gill MA, Ahmad N, Khan M, Asghar F, Rasool A, et al. Cyber attacks detection through machine learning in banking. Bull Bus Econ (BBE). 2023;12:34-45.

[29] Lippmann RP, Fried DJ, Graf I, Haines JW, Kendall KR, et al. Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation. InProceedings DARPA Information survivability conference and exposition. DISCEX'00 . IEEE. 2000;2:12-26.

[30] Tavallaee M, Bagheri E, Lu W, Ghorbani AA. A detailed analysis of the kdd cup 99 dataset.In: IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA). IEEE; 2009:1-6.

[31] Moustafa N, Slay J. Unsw-nb15: A comprehensive data set for network intrusion detection systems .In: Military Communications and Information Systems Conference (MilCIS). IEEE; 2015:1-6.

[32] Sharafaldin I, Habibi Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: International Conference on Information Systems Security and Privacy (ICISSP); 2018:108-116.

[33] Sharafaldin I, Lashkari AH, Hakak S, Ghorbani AA. Developing realistic distributed denial of service (ddos) attack dataset and taxonomy. In: 53rd International Carnahan Conference on Security Technology (ICCST). IEEE; 2019:1-8.

[34] Garćıa S, Grill M, Stiborek J, Zunino A. An empirical comparison of botnet detection methods. Comput. Secur. 2014;45:100-123

[35] Booij TM, Chiscop I, Meeuwissen E, Moustafa N, den Hartog FTH. ToN IoT: the role of heterogeneity and the need for standardization of features and attack types in iot network intrusion datasets. IEEE Internet Things J. 2021;8:13988-14003.

[36] Koroniotis N, Moustafa N, Sitnikova E, Turnbull B. Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: BoT-IoT dataset. Future Gener Comput Syst. 2019;100:779-796.

[37] Lashkari AH, Kadir AF, Taheri L, Ghorbani AA. Toward developing a systematic approach to generate benchmark android malware datasets and classification. In2018 International Carnahan conference on security technology (ICCST) IEEE. 2018:1-7.

[38] Ferrag MA, Shu L, Djallel H, Choo KK. Deep learning-based intrusion detection for distributed denial of service attack in agriculture 4.0. Electronics. 2021;10:1257.

[39] Sarhan M, Layeghy S, Portmann M. Towards a standard feature set for network intrusion detection system datasets. Mob Netw Appl. 2022;27:357-370.

[40] Ferriyan A, Thamrin AH, Takeda K, Murai J. Generating network intrusion detection dataset based on real and encrypted synthetic attack traffic. Applied Sciences, 2021;11:7868.

[41] Forrest S, Hofmeyr SA, Somayaji A, Longstaff TA. A sense of self for Unix processes. In: IEEE Symposium on Security and Privacy. IEEE; 1996:120-128.

[42] Shiravi A, Shiravi H, Tavallaee M, Ghorbani AA. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. computers & security. 2012;31:357-374.

[43] Creech G, Hu J. A semantic approach to host-based intrusion detection systems using contiguous and discontiguous system call patterns. IEEE Trans Comput. 2013;63:807 - 819.

[44] Lippmann R, Fried D, Graf I, Haines J, Kendall K, et al. Evaluating intrusion detection systems: the 1998 DARPA off-line intrusion detection evaluation. In: Proceedings of the DARPA information survivability conference and exposition (DISCEX '00). IEEE; 2000;2:12-26.

[45] Cappelli DM, Moore AP, Trzeciak RF. The CERT guide to insider threats: how to prevent, detect, and respond to information technology crimes (theft, sabotage, fraud). SEI Series in Software Engineering. Addison-Wesley; 2012.

[46] Zhu J, He S, Liu J, He P, Xie Q, et al. Tools and benchmarks for automated log parsing. In2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP) IEEE. 2019:121-130.

[47] Ezeme O, Mahmoud Q, Azim A, Lescisin M. SysCall dataset: A dataset for context modeling and anomaly detection using system calls .Mendeley Data. 2019.

[48] Shen Y, Yu F, Zhang LF, An JY, Zhu ML. An intrusion detection system based on system call. In2005 1st IEEE and IFIP International Conference in Central Asia on Internet. IEEE. 2005:4.

[49] Rigaki M. and Garcia S. 2023. A Survey of Privacy Attacks in Machine Learning. ACM Comput. Surv. 2023;56:1-34.

[50] Song X, Zhang R, Dong Q, Cui B. Grey-box fuzzing based on reinforcement learning for xss vulnerabilities. Applied Sciences. 2023;13:2482.

[51] thttp://ptsecurity.com/ww-en/analytics/web-application-attacks2021-dataset/. A set of traffic records for training and testing machine learning models for WAFs.

[52] Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. Proceedings of the 4th International Conference on Information Systems Security and Privacy. 2018:108-116.

[53] https://zenodo.org/records/17178461

[54] https://www.imperva.com/docs/hii_web_application_attack_report_ed2.pdf

[55] Sharafaldin, Iman Habibi Lashkari, Arash Ghorbani Ali. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. Proceedings of the 4th International Conference on Information Systems Security and Privacy ICISSP. 2018:108-116.

[56] Zhu T, Weng Z, Fu L, Ruan L. A Web Shell Detection Method Based on Multiview Feature Fusion. Applied Sciences, 2020;10:6274.