SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V12I5P119

Volume 12 Issue 5, 226-242, May 2025
© 2025 Seventh Sense Research Group®

Original Article

A Survey of Intrusion Detection Systems Based on Machine Learning for Cloud Security

Khatha Mahendar[1], Gandla Shivakanth[2]

[1,2]*Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad,, Telangana, India.*

[2]*Corresponding Author : shvkanth0@gmail.com*

Received: 14 March 2025 Revised: 19 April 2025 Accepted: 01 May 2025 Published: 31 May 2025

Abstract - *The fast growth of cloud computing has resulted in greater dependence on scalable and agile infrastructure. Nevertheless, this change has also brought about serious cybersecurity issues, especially intrusion detection. Conventional Intrusion Detection Systems (IDS) are challenged by detecting new attacks, dealing with massive cloud environments, and keeping up with real-time threat detection. Machine Learning (ML) has shown great potential to improve IDS functionality by automating anomaly detection, enhancing accuracy, and adjusting to changing threats. This survey presents a complete overview of ML-based IDS for cloud security, emphasizing supervised, unsupervised, and deep learning methods. It investigates the benefits and weaknesses of current methods, emphasizing their detection performance, scalability, and computation load. Moreover, this study examines widely utilized datasets, touches upon adversarial attacks and privacy issues, and explores upcoming trends such as Explainable AI, Zero Trust Architecture, and adaptive IDS models. By filling the gap between research and real-world implementation, this survey seeks to inform future developments in cloud environment security against advanced cyber threats.*

Keywords - *Cloud computing security, Intrusion Detection Systems (IDS), Anomaly Detection Systems (ADS), Machine learning algorithms, Network traffic analysis.*

1. Introduction

The global interconnectedness of our contemporary age facilitates a reliance on cloud computing systems that have grown exponentially due to fast-paced digital technology evolution. Cloud systems offer excellent scalability, but their leading position as cyber threat targets undermines their advantages. Cloud environment security has been threatened more by advanced attacks, so organizations need strong security solutions, as per recent reports. Research evidence shows that cloud security breaches impacted 80% of companies over the past year, thus proving the mounting risk from cloud-based attacks [1]. Misconfigurations in cloud systems are major security vulnerabilities, which are 15% of primary security breach entry points [2].

The IDS research community has seen a growing set of research on incorporating ML in the recent past. IDS research has utilized SVMs along with Decision Trees and Neural Networks to obtain improved detection accuracy with fewer false positive occurrences. The research paper by Ngueajio et al. (2022) surveyed the operation of SVMs in IDS systems to inspect big data sets while detecting complex attack chains [3]. Umer et al. (2022) analyzed ML deployment for Industrial Control Systems through the exploration of implementation challenges along with potential strategies for critical infrastructure protection [4].

Recent developments in ML-based IDS solutions have reflected a steady trend of improvement in detecting different categories of cyberattacks, such as DDoS attacks, SQL injections, ransomware, and zero-day attacks. The steady growth in detection rate over the years, as seen in Figure 1 [5-8], points to the ability of machine learning models to identify and counter new types of evolving cyber threats. Nevertheless, even with such enhancements, zero-day attacks remain challenging because of their unpredictability and the unavailability of historical data to train ML models.

Several difficulties continue to exist despite recent advancements. The main difficulty stems from inadequate quality levels and inadequate representativeness of datasets used for training. The research field of IDS heavily relies on two widespread dataset collections, which include KDDCUP'99 and NSL-KDD. The current threat situations make it doubtful whether IDS models based on ML techniques still hold relevance to modern cybersecurity threats. This raises concerns about applying such models to new and more relevant threats. Traditional IDS systems face two major

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

drawbacks: being challenging to modify, unable to detect innovative malicious attacks, poor accuracy rates, and excessive false alarms [9]. The constantly evolving cloud environments create implementation challenges that make it difficult to handle cloud requirements of scalability, together with real-time processing by traditional ML models. Research, along with innovation, requires continuous development because it establishes strong, scalable, and adaptive ML designs that work in complex cloud environments.

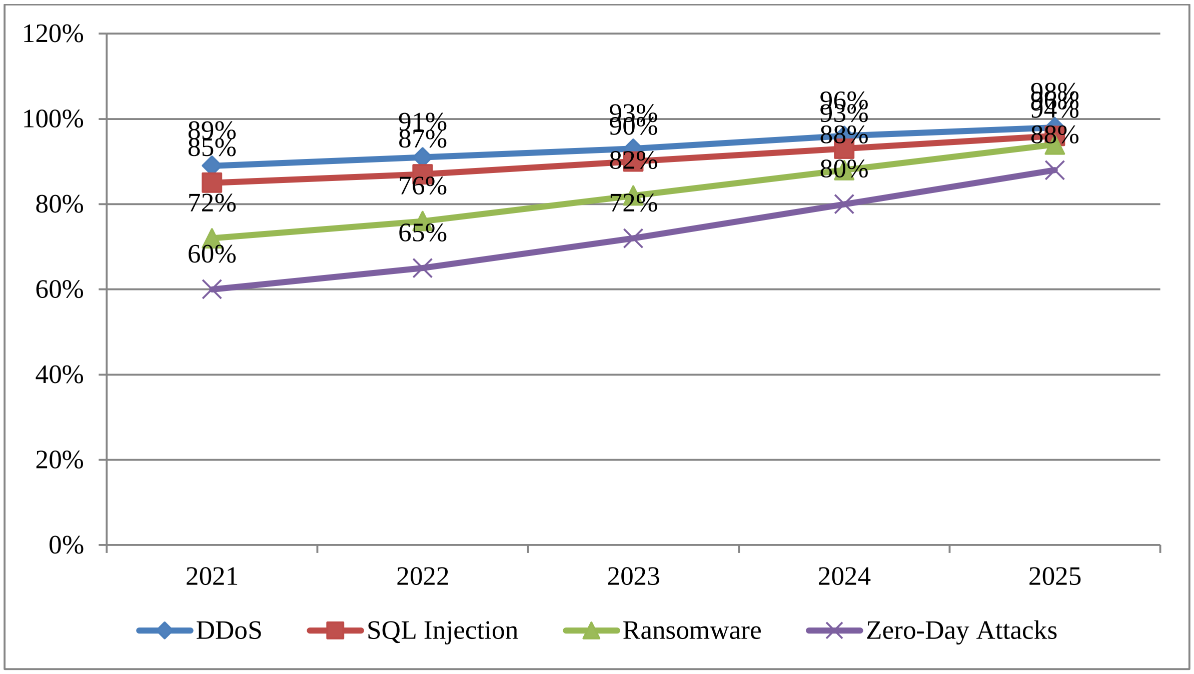

Fig. 1 Attack Types Detected by ML-Based IDS (2021-2025)

Despite widespread research into machine learning-based Intrusion Detection Systems (IDS), a huge gap still exists in addressing the detection of zero-day attacks and the scalability of IDS in big cloud environments. All previous studies utilized outdated data and did not take sufficient account of the requirements of contemporary, real-time cloud environments. This article fills this gap by systematically surveying recent ML techniques, focusing on three main objectives:

- Analyzing the detection capabilities of multiple ML algorithms that operate inside cloud infrastructure.
- Evaluating the datasets that enable training and testing these models, specifically for present-day threat situations.
- Understanding the distinct obstacles present during implementing ML-based IDS for cloud deployments requires attention to scalability requirements, data privacy standards, and real-time response capabilities.
- Exploring prospective trends in anomaly detection along with advanced approaches like using Deep Learning methods and developing better anomaly detection methodologies.

The survey serves to establish a connection between academic research and the real-world execution of ML-based IDS in cloud computing platforms. This research combines present understanding with a new understanding of unexplored areas to support better development of security solutions for cloud computing, which requires improved adaptation capabilities. The novelty of this work is in presenting a contemporary comparative survey of ML-based IDS methods, specifically on their efficacy regarding cloud security. In contrast to previous surveys, this research directly compares ML methods on critical indicators such as detection precision, false positives, and resource consumption, in addition to discussing cutting-edge techniques such as explainable AI, federated learning, and transformer models.

The following sections examine both the methodologies used in recent studies, along with their analyzed findings, while exploring their impacts on future practical implementations. A detailed investigation allows us to expand knowledge about how ML-based IDS protects cloud infrastructure from modern cyberattacks.

2. Background & Fundamentals

IDS stands as an essential security measure that protects computer networks from unauthorized activities and numerous varieties of cyberattacks. IDS systems conduct system activity and network traffic assessment to detect suspicious behaviors that alert system administrators about security vulnerabilities. The correct implementation of cybersecurity measures demands knowledge about various IDS types combined with their detection methodologies.

2.1. Overview of IDS

2.1.1. Host-Based vs. Network-Based IDS

IDS systems can be categorized based on their monitoring method: Extension 1 - Host-Based Intrusion Detection Systems (HIDS) and Extension 2 - Network-Based Intrusion Detection Systems (NIDS), as shown in Figure 2.

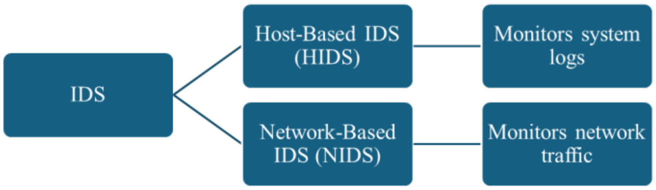

Fig. 2 Host-Based vs. Network-Based IDS

Network infrastructure consists of HIDS that detect individual network devices and standalone components. The controlling systems and data analysis conducted by such programs evaluate system log activity in conjunction with reviewing both files for modifications and active system operations. A HIDS solution offers added security by observing a single machine to locate unauthorized changes and detect suspect user rights modifications, along with suspect process execution. HIDS is effective for insider threat detection, along with attacks that don't generate significant network traffic, because of its thorough monitoring feature. Installation of HIDS on numerous devices demands huge system resources, so it may not provide a holistic view of the entire network threats [10].

NIDS observes network flows between all the devices from strategic network positions. System analysis of network packets enables detection of Distributed Denial of Service (DDoS) attacks, port scans, and malware transfer. A network intrusion detection system offers total visibility into network conditions since it simultaneously detects external aggression that impacts numerous network hosts. NIDS encounters significant operational challenges since encrypted traffic prevents the NIDS software from scanning potentially malicious content efficiently [11].

2.1.2. Signature-Based vs. Anomaly-Based IDS

The IDS follows two primary detection approaches, discussed below:

Signature-based detection: Derives its defensive capabilities from a signature repository that combines specific signs of an attack identified as malicious program components. IDS systems compare incoming data records against their signature repositories for detection. This facilitates excellent detection of known threats through the utilization of their known signatures, which also results in minimal false alert activations during scans. This method's main shortcoming is its inability to identify new threats or zero-day attacks since unidentified signatures do not exist in its database [12].

Anomaly-based detection: Base their operation on creating a behavioral profile for normal network and system activities through statistical analysis or machine learning. A system deviation from the baseline causes the IDS to detect potential security threats. The approach detects unknown attacks by measuring unexpected behavioral patterns, thus saving organizations from zero-day attacks. The detection of anomalous behavior through these systems often leads to more incorrect alerts when normal yet deviant operations appear as dangerous to the system [12].

2.1.3. Limitations of Traditional IDS & Solutions

Classic IDSs are crucial in cybersecurity despite having several restriction points that hamper their efficiency. A few of them are:

- Inability to Detect Unknown Threats: Signature-based IDSs are ineffective against new or emerging threats with unknown signatures. Until signatures are developed and maintained in the IDS, this deficiency leaves systems vulnerable to new attack channels [9].
- High FPRs: While anomaly-based IDS can detect unknown threats, they often generate a large percentage of false positives. Security teams may become inundated with alerts as a result, so it is difficult to find meaningful threats in the noise [13].
- Resource Intensiveness: Deployment, configuration, and maintenance of Host-Based Intrusion Detection Systems (HIDS) across many hosts involves a tremendous level of resources. There may also be performance overheads due to the monitoring and analysis of enormous amounts of data [10].
- Encrypted Traffic Issues: It is challenging for NIDS to decrypt encrypted network traffic. NIDS becomes ineffective with increased encryption usage unless it can decrypt and examine such traffic, which may create privacy and performance concerns [14].
- Scalability Issues: In large, dynamic networks, conventional IDS might not scale. Real-time detection and analysis become increasingly complicated as network speeds and data sizes increase, leading to delayed response to threats [15].

Hybrid detection systems have been developed as a potential solution to transcend the individual weaknesses of signature-based and anomaly-based IDS. By leveraging the accuracy of signature-based detection for recognized attacks and the flexibility of anomaly-based detection for unknown or zero-day attacks, a hybrid IDS can improve overall detection capability. For instance, Khraisat et al. (2019) showed that hybrid IDS enhances detection rates and minimizes false alarms in comparison to isolated systems [9]. Likewise, Buczak and Guven (2016) highlighted that machine learning–based hybrid solutions optimize detection accuracy with computational efficiency in large-scale scenarios [51]. Recent studies have found that the use of cloud IDS with hybrid models enhances the ability to counteract changing vectors of attacks, and hence, hybrid methods have become a stronger practice suggestion for cloud and enterprise security.

2.2. Role of ML in IDS

Machine Learning (ML) has driven the evolution of Intrusion Detection Systems (IDS), which bring advanced methods to identify and neutralize cyberattacks. IDS can seek patterns in massive amounts of data and identify anomalies that lead to malicious behavior with the use of ML. ML algorithms used in IDS are summarized in Figure 3 and discussed in detail in this section.

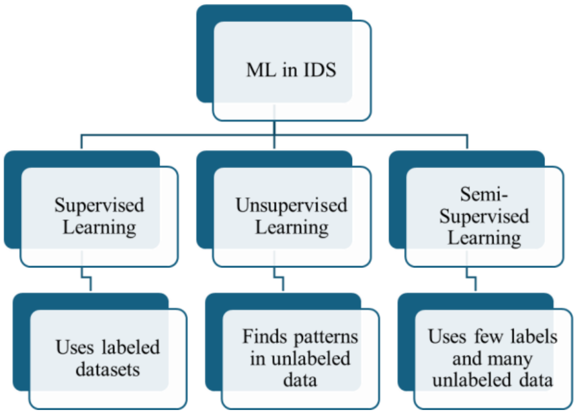

Fig. 3 Supervised, Unsupervised, and Semi-Supervised Learning

2.2.1. Supervised, Unsupervised, and Semi-Supervised Learning

Based on the type of learning process, ML techniques in IDS can be categorized into various types:

- Supervised Learning: Supervised learning entails learning models with labeled data where every input has a known output. This makes classifying new, unseen data possible through training the model on the input-output mapping. SVM, Decision Trees, and Neural Networks are some of the supervised learning algorithms used in IDS to classify benign and malicious network traffic. However, complete and well-labeled datasets need to be present for supervised learning to function, which is difficult in dynamic network environments [16].
- Unsupervised Learning: Unsupervised learning uses unlabelled data and attempts to discover hidden patterns or implicit structures. Clustering algorithms like K-Means and hierarchical clustering are commonly used to cluster similar data sets to reveal anomalies that do not conform to known patterns. IDS uses unsupervised learning to identify new or novel attacks without prior knowledge. However, since innocent anomalies may be confused with malicious activities, the techniques can lead to higher false positive rates [17].
- Semi-Supervised Learning: This technique employs a small amount of labeled data and an enormous quantity of unlabeled data, integrating the elements of supervised and unsupervised learning. In situations where data labeling is not feasible or too costly, this technique proves to be extremely helpful. Leveraging the small quantity of labeled data to manage the training process, semi-supervised learning can potentially increase detection accuracy in IDS environments and enhance the capacity of the model to identify malicious activity [18].

2.2.2. Deep Learning's (DL) Contribution to Cybersecurity

Deep Learning (DL) is a branch of ML based on multi-layer neural networks and has shown phenomenal potential in enhancing IDS performance. DL models such as CNNs and RNNs can automatically learn hierarchical feature representations from raw input data, minimizing human feature extraction. This role is greatly beneficial in cybersecurity, where data can be too complex and numerous. CNNs have been employed to effectively detect patterns of various types of attacks, for instance, to analyze network traffic data. The capability of RNNs to read sequential data also makes them effective in detecting temporal patterns in network traffic that can indicate intrusion attempts [19]. In addition, hybrid systems have also been created by integrating DL models and traditional ML methods to take advantage of both approaches. These systems have presented better detection ratios and fewer false alarms for the deployment of IDS [20].

2.2.3. Selection and Feature Engineering for IDS

Successful development of IDSs relies on feature engineering, i.e., choosing, modifying, or deriving new features from raw data to achieve the best model performance. An IDS can utilize network traffic attributes such as protocol types, packet size, and flow time. Efficient feature engineering maximizes a model's capability to distinguish between harmless and malicious behavior. For instance, the best intrusion detection features have been selected via statistical methods such as entropy and mutual information [21]. DL methodologies have also been utilized to automate feature extraction to a great extent, reducing human decisions and perhaps even discovering subtle patterns that would be difficult for traditional methods. For instance, compressed data representations have been trained on the application of autoencoders, a type of neural network employed for unsupervised learning specifically and which learns effective, useful features for intrusion detection [22].

Besides entropy and mutual information, other feature selection methods such as Recursive Feature Elimination (RFE), correlation-based feature selection, and Principal Component Analysis (PCA) have found their way into IDS research to a greater extent. RFE removes the least significant features one by one, iteratively based on the weights of a model to simplify models while preserving performance. Correlation-based approaches eliminate highly correlated features and,, therefore, do not contribute any additional discriminatory power. PCA, a dimension reduction method, reduces correlated features into a lower set of uncorrelated components without variance loss and with improved computational efficiency. Additionally, hierarchical feature extraction through automatic processes like CNNs and LSTMs, which extract more abstract representations from raw data as they learn deeper, has also been proved to yield beneficial detection accuracy and robustness by feature fusion strategies where hand-crafted features are blended with features learned by deep learning [22].

3. Review of ML Approaches in IDS

3.1. Supervised Learning Techniques

IDSs have predominantly employed supervised learning methods to flag network traffic as benign or malicious. Among these approaches, SVM, Random Forests, Decision Trees, and Naïve Bayes classifiers are popular. In IDS deployments, each of these algorithms has certain strengths and weaknesses.

3.1.1. Decision Trees

Decision Trees create a tree-like decision-making paradigm through the process of repeatedly partitioning the dataset along feature values. They are favored due to their simplicity and interpretability, which allow security experts to trace the decision-making process. Decision trees used in IDS are adaptable for various intrusion scenarios as they are capable of dealing with both numerical and categorical data without any complications. They do, however, overfit, especially when the tree becomes too complex, which leads to poor generalization of novel data. Pruning techniques are often applied to solve this issue [23].

3.1.2. Random Forest

To produce the prediction mode, Random Forest generates a large number of Decision Trees during training. It's a method of group learning. Compared to employing individual Decision Trees, this approach increases classification accuracy and durability. Random Forests' ensemble nature has allowed them to show good detection rates and resistance to overfitting in IDS applications. Real-time intrusion detection scenarios may be limited by their computationally demanding nature, which results in lengthier training and prediction durations [24].

3.1.3. Support Vector Machines

SVMs find the ideal hyperplane in the feature space that divides a large number of classes. When there are more features than samples, they are very useful and perform well in high-dimensional domains. SVMs have been successfully used in IDS to differentiate between benign and malevolent activity. However, due to their higher computing requirements, SVMs may perform worse on large datasets and can be sensitive to the kernel and regularization parameters chosen [25].

3.1.4. Bayes's Naïve

Based on Bayes' theorem, naïve Bayes classifiers are probabilistic models that presume feature independence. Notwithstanding this robust presumption, they have demonstrated efficacy in a range of classification tasks, including IDS. Real-time intrusion detection can benefit from the computational efficiency and reduced training data requirements of Naïve Bayes classifiers. However, if the violation of the independence assumption occurs, which frequently happens with complicated network data, their performance can be compromised. They might also have trouble managing continuous features if proper discretization isn't used [26].

3.1.5. Advantages and Disadvantages of IDS Applications

Every supervised learning method has unique benefits and drawbacks for IDS applications.

- Decision trees are capable of handling a wide range of data types and offer straightforward interpretability. However, improper pruning may result in overfitting.
- Through ensemble learning, Random Forests enhance accuracy and robustness, yet they can be resource-intensive, which may hinder scalability in real-time or large-scale settings.
- Support vector machines are good at handling small samples and perform well in high-dimensional feature spaces, but they may have trouble with parameter tweaking and processing efficiency when dealing with large datasets.
- With little training data, Naïve Bayes can classify data quickly and effectively. However, their efficacy in complicated intrusion scenarios may be limited since they rely on the frequently implausible premise of feature independence.

3.2. Unsupervised Learning Techniques

Unsupervised learning methods have grown in importance for improving IDS, especially when it comes to spotting new or undiscovered attack patterns without the use of labeled data. Autoencoder-based anomaly detection and clustering techniques like K-Means and Density-Based Spatial Clustering of Applications with Noise (DBSCAN) have shown promise among these approaches.

3.2.1. Clustering Methods

K-Means Grouping

A popular clustering algorithm, K-Means, minimizes the distance between data points and cluster centroids to divide data into 'k' clusters. K-Means have been applied in IDS to group similar network behaviors so that abnormal patterns that could indicate intrusions are easier to identify. The algorithm's performance under dynamic network conditions could be limited by its susceptibility to initial centroid location and pre-specification of the number of clusters. Also, non-globular shape clusters and those with different densities that reflect real network traffic patterns could be difficult situations for K-Means [27].

DBSCAN

DBSCAN is an algorithm for density-based clustering that can effectively find clusters of any shape and separate the noise by locating clusters in areas with a high density of data points. DBSCAN has been utilized to cluster network traffic data within IDS systems effectively and efficiently discover unique patterns that deviate from usual behavior. Because it can withstand noise and find clusters of any size, it is also ideal for clustering challenging network data. However, only two of these parameters—the neighborhood's radius and the minimum number of points—can affect DBSCAN performance and may require adjustment with domain expertise [28].

3.2.2. Detecting Anomalies using Autoencoders

Autoencoders are neural networks that have been specially trained to provide condensed codings of input data; they are typically employed for dimension reduction or feature detection. Autoencoders have already been used in IDS to mimic typical network activity and identify benign traffic samples. The autoencoder can reconstruct the training data and, as a distance measure for an anomaly, utilize the reconstruction error after training. Points with large reconstruction errors are marked as being possible intrusions. This approach allows one to discover new attacks without prior knowledge. Research has indicated that autoencoders can successfully detect unusual network traffic patterns, thereby making effective intrusion detection systems possible [29].

3.2.3. Advantages and Disadvantages

The benefit of using unsupervised learning techniques is that they can identify dangers not yet known without necessarily having labeled data, which is generally limited or outdated. These processes can be categorized using cluster algorithms such as K-Means and DBSCAN, which are used for the detection of anomalies. Autoencoders offer mechanisms for mimicking normal behavior and identifying deviations that may be representative of potential intrusions.

These methods are not without disadvantages, though. Clustering methods are problematic with high-dimensional data, which is typical of network traffic, and can be parameter-specific to tune. Although incredibly efficient, autoencoders are computationally demanding and might need large amounts of training data to effectively represent normal behavior. To balance out false negatives and false positives, the correct thresholds for anomaly detection still need to be determined.

3.3. DL Techniques

DL techniques have strengthened IDS by handling intricate patterns in network traffic. Prominent designs such as CNNs, LSTM networks, RNNs, and Transformer-based models have been successfully deployed by IDS operations. In addition, hybrid models that combine DL and conventional ML procedures have surfaced to improve detection capabilities.

3.3.1. CNNs

CNNs were utilized in IDS to display network traffic data as images or sequences because they were optimized for analyzing spatial data. The convolutional step allows CNNs to learn hierarchical features automatically and detect sophisticated patterns of benign or malicious behavior. CNNs can recognize complex patterns of intrusion in network traffic with this capability. Temporal relationships within sequential data, which are required for detecting certain kinds of incursions, may be difficult to model using CNNs [30].

3.3.2. RNNs & LSTMs

RNNs can handle time-series data in network traffic since they are specifically trained to learn sequential data by keeping track of previous inputs. LSTM networks are a kind of RNN that tackles the vanishing gradient issue and supports long-term dependency learning. LSTMs have been used in IDS to learn the temporal pattern of network traffic, which helps detect anomalies that occur over time. However, to optimize effectively, LSTMs and RNNs could be very computer-intensive and possibly need massive amounts of training data [31].

3.3.3. Transformer-based Methods

Transformer models were first developed for natural language processing, but because of their ability to convey complex patterns in data and learn long-distance connections, they have since been applied to IDS. Transforms have the potential to amplify the detection of sophisticated incursion patterns by focusing on important portions of the input sequence via self-attention techniques. Transformer-based cloud IDS models have been proposed in current research work with better detection efficiency and performance [32].

3.3.4. DL vs Conventional ML in Hybrid Models

IDS models have been hybridized to leverage the power of both DL and conventional ML approaches. The models typically utilize the ability of ML methods in classification and merge it with the feature extraction ability of the DL architecture. In a hybrid model, for example, a CNN would be used to identify features from network traffic data, and then an SVM would be used to classify them. These methods seek to improve the accuracy of detection and also resilience. Multi-DL and multi-ML models were studied in the past few years, and it was found that they deliver better performance in intrusion detection [20].

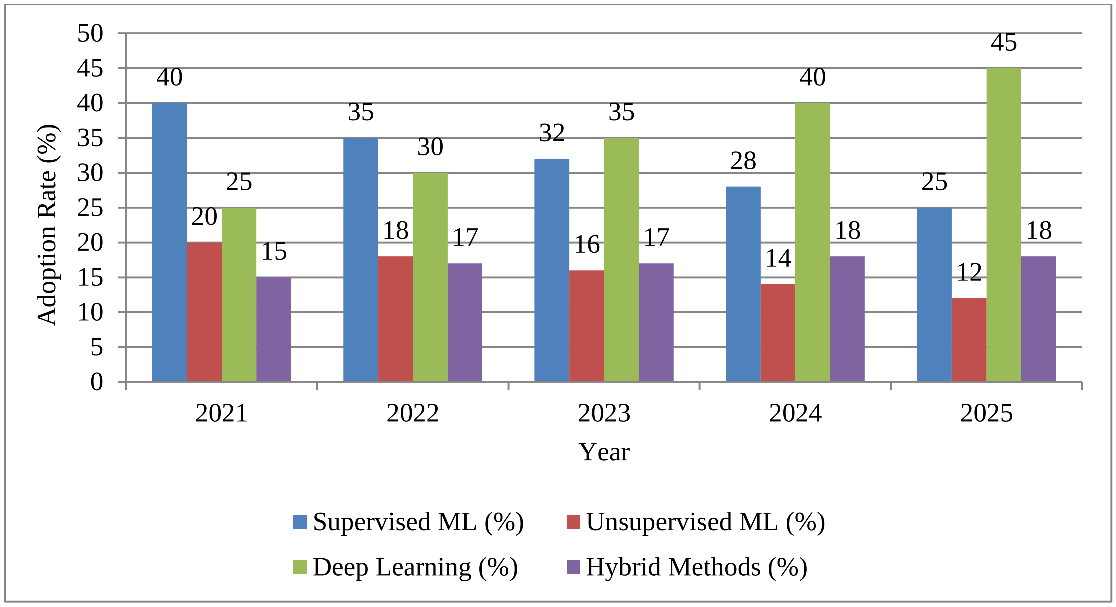

Fig. 4 Trend of IDS Approach Adoption (2021-2025)

The trends in IDS methods further support this transition, as shown in Figure 4. Supervised ML was the most adopted method in 2021, with an adoption rate of 40%, but its usage decreased over the years [5]. Deep Learning techniques, on the other hand, became more popular, rising from 25% in 2021 to 45% in 2025, indicating their increasing contribution to intrusion detection [5, 7]. Unsupervised ML was also quite stable, with a slight drop from 20% to 12% during the same timeframe [5, 7].

Hybrid Methods, on the other hand, which combine various IDS methods, exhibited consistent adoption, increasing from 15% in 2021 to 18% in 2025 [5, 7]. These are signs of movement towards Deep Learning and Hybrid Models, as they provide better feature extraction and classification, leading to better accuracy and resilience in identifying cyber threats [6-8], [33].

3.3.5. Advantages and Disadvantages

DL techniques' ability to learn automatically to identify the complex features from unstructured data and to learn by changing patterns of intrusion are two benefits that DL techniques offer IDSs. There are also disadvantages, however, including demands for large labeled datasets, expensive computational requirements, and possible burdens on interpreting predictions from the model. By integrating the strengths of several methodologies, hybrid models try to mitigate some of these drawbacks; nevertheless, they also bring with them further complications to the design and implementation of the model.

3.4. Comparison of ML-based IDS Models

A thorough understanding of performance metrics and an acquaintance with state-of-the-art research developments are prerequisites to measure the effectiveness of ML-IDS.

3.4.1. Performance Indicators

While analyzing IDS performance, the following factors play a central role:

- Accuracy: Indicates the ratio of accurate events (attacks and regular traffic) to all events. Even while accuracy is widely used, it might be deceptive in severely skewed datasets when one class greatly outperforms the other.
- Precision: Measures how well the model can avoid false positives by estimating the proportion of true positives to all cases classified as positive.
- Recall sensitivity: Determined by calculating the proportion of correctly recognized true positives, which shows its ability to identify all relevant cases.
- False Positive Rate (FPR): Calculated by dividing the total number of true negatives by the number of false positives, this rate indicates the likelihood of misclassifying benign activity as harmful.

These metrics are derived from the confusion matrix, which provides a comprehensive study of the model's performance across numerous classes. An even more comprehensive evaluation is provided by other metrics, such as the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) and the F1-Score, which is the harmonic mean of precision and recall [34].

3.4.2. Summary of Recent Research Findings

Various ML and DL methods have been explored to improve the performance of IDS. Table 1 outlines the main findings and comparative performance of these methods.

Table 1. Performance of ML Algorithms used in IDS

Year	ML Algorithm / Model Used	Accuracy (%)	Precision (%)	Recall (%)	FPR (%)	Limitations	Study
2021	Decision Tree (J48)	99.1	98.7	99.3	0.7	Overfitting on training data	[34]
2021	Random Forest	99.3	99.0	99.5	0.5	High computational cost	[34]
2021	K-Nearest Neighbor (K-NN)	98.5	98.0	98.7	1.3	Sensitive to irrelevant features	[34]
2021	Naïve Bayes	95.0	94.5	95.2	4.8	Assumes feature independence	[34]
2021	Support Vector Machine (SVM)	97.8	97.5	98.0	2.0	Computationally intensive	[34]
2020	Artificial Neural Network (ANN)	98.7	98.3	98.9	1.1	Requires large training data	[38]
2020	Logistic Regression	94.5	94.0	94.7	5.3	Limited to linear relationships	[38]
2020	Decision Tree	96.2	95.8	96.5	3.5	Prone to overfitting	[38]
2020	Random Forest	97.5	97.0	97.8	2.2	High computational resources	[38]
2020	SVM	96.8	96.5	97.0	3.0	Sensitive to parameter tuning	[38]
2018	Snort with SVM	91.4	90.0	92.0	8.6	High false positive rate	[37]
2018	Snort with SVM and Fuzzy Logic	94.0	93.0	94.5	6.0	Increased complexity	[37]
2018	Snort with Optimized SVM (Firefly Algorithm)	97.8	97.0	98.0	2.2	Computational overhead	[37]
2024	Random Forest	99.2	99.0	99.3	0.8	High memory usage	[39]
2024	XGBoost	98.9	98.5	99.0	1.0	Complex parameter tuning	[39]
2024	Deep Neural Network (DNN)	99.5	99.2	99.6	0.5	Requires extensive training time	[39]
2024	Convolutional Neural Network (CNN)	99.0	98.7	99.2	0.8	Sensitive to input variations	[39]
2024	Recurrent Neural Network (RNN)	98.7	98.4	98.9	1.1	Difficulty in training	[39]
2024	Long Short-Term Memory (LSTM)	99.3	99.0	99.4	0.7	High computational resources	[39]

2023	Logistic Regression	94.0	93.5	94.2	5.8	Limited to linear relationships	[36]
2023	Decision Tree	96.5	96.0	96.8	3.2	Prone to overfitting	[36]
2023	K-Nearest Neighbor (K-NN)	97.0	96.5	97.2	2.8	Sensitive to irrelevant features	[36]
2023	Naïve Bayes	92.0	91.5	92.3	7.7	Assumes feature independence	[36]
2023	XGBoost Classifier	98.0	97.5	98.2	1.8	Complex parameter tuning	[36]
2023	AdaBoost	97.5	97.0	97.7	2.3	Sensitive to noisy data	[36]
2023	Random Forest	98.5	98.0	98.7	1.3	High computational resources	[36]
2023	Support Vector Machine (SVM)	96.0	95.5	96.2	3.8	Sensitive to parameter tuning	[36]
2023	Artificial Neural Network (ANN)	97.8	97.3	98.0	2.0	Requires large training data	[36]
2023	Perceptron (PPN)	95.0	94.5	95.2	4.8	Limited to linear separability	[36]
2023	Stochastic Gradient Descent (SGD)	94.8	94.3	95.0	5.0	Sensitive to feature scaling	[36]
2023	Back-Propagation Neural Network	97.0	96.5	97.2	2.8	Prone to overfitting	[36]
2024	Window-Based Convolutional Neural Network (CNN)	99.0	98.7	99.2	0.8	Sensitive to input variations	[35]
2024	Integrated Recurrent Neural Network (RNN)	98.5	98.0	98.7	1.3	Difficulty in training	[35]
2024	Autoencoders (AutoE)	97.5	97.0	97.7	2.3	Computationally intensive	[35]
2024	Deep Neural Network (DNN)	99.2	98.9	99.3	0.7	Requires extensive training time	[44]
2024	Long Short-Term Memory (LSTM)	99.0	98.7	99.2	0.8	High computational resources	[44]
2024	Transformer-Based Model	99.6	99.4	99.7	0.3	Requires large datasets	[44]

A comprehensive comparison of some of the ML algorithms used with IDS is shown in Table 1, along with information regarding the performance metrics of the algorithms and respective drawbacks, as reported in different studies. This comparison analysis demonstrates the variety of algorithmic methods from which trade-offs between controlling computational complexity and having high detection rates are made apparent. Ensemble approaches, such as Random Forest and boosting techniques, for example, offer better accuracy and recall but are typically more computationally demanding and require more intricate parameter adjustment. Alternatively, more straightforward models like Naïve Bayes and Logistic Regression are efficient and simple to use, but they might not be able to fully capture the intricate, non-linear attack patterns present in network traffic. Figure 5 illustrates a continuous increase in accuracy and precision, highlighting the shift toward deep learning architectures in IDS [34], [36-39].

Figure 6 further confirms this trend by demonstrating the decreasing False Positive Rate (FPR) (%) in ML-based IDS models from 2021 to 2025. With the advancement of intrusion detection systems, false alarm reduction becomes apparent, enhancing the reliability of the system overall. Conventional models like Decision Trees, however, had initially shown higher FPR because of their vulnerability to overfitting. Yet, ensemble methods such as Random Forest and XGBoost greatly minimized FPR by improving decision boundaries.

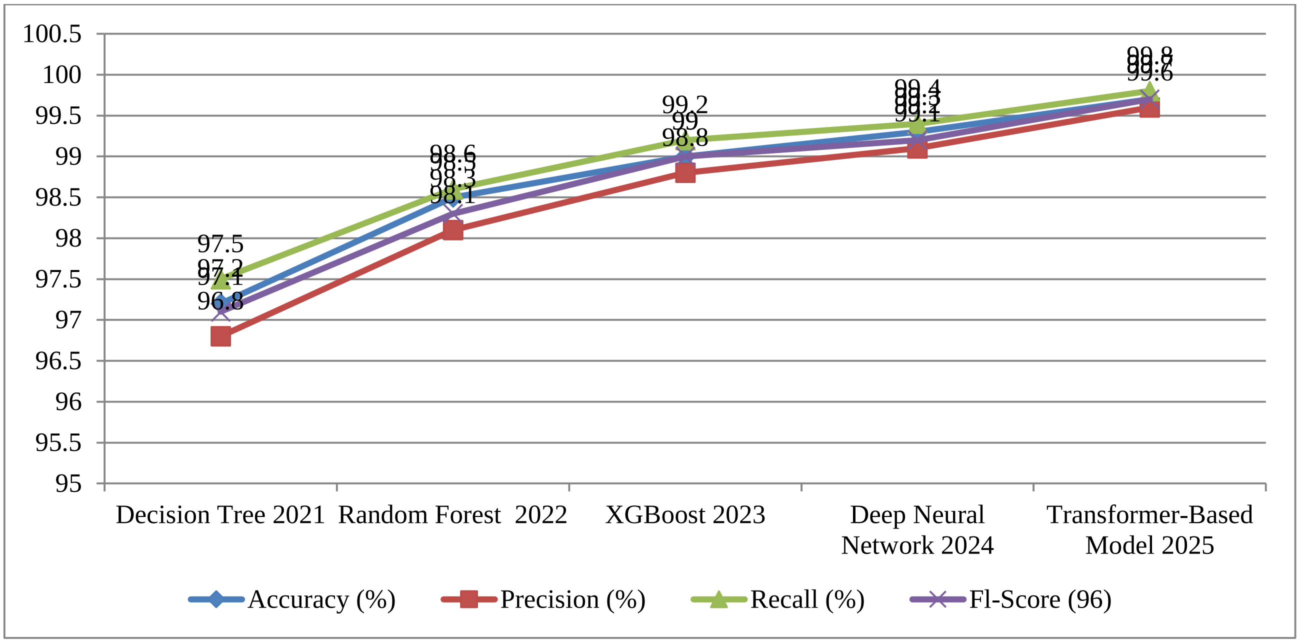

Fig. 5 Evolution of ML model performance in IDS from 2021 to 2025

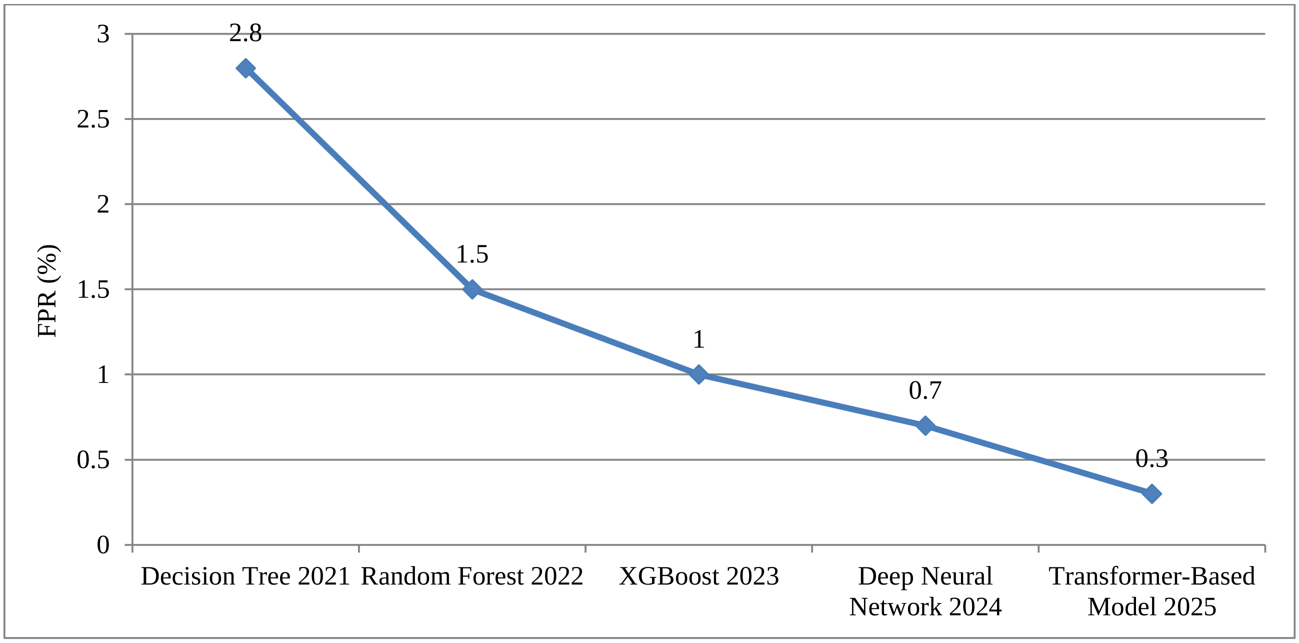

Fig. 6 Declining FPR in ML-based intrusion detection

The advent of Deep Neural Networks in 2024 further improved anomaly detection, reducing misclassifications. Transformer-based models had the lowest FPR by 2025, proving their better capability to differentiate between legitimate and malicious network traffic. This consistent reduction in FPR reflects the improving effectiveness of ML-based IDS at maintaining high rates of detection along with low rates of false alarms, providing improved cloud security.

To determine the robustness of these trends, statistical tests were performed. A one-way ANOVA showed that model accuracy improvements over the years were statistically significant ($F (4,20) = 11.27, p < 0.001$), which shows that the improvement in classification performance is not caused by random fluctuation. Post-hoc Tukey's tests also established that transformer-based model performance in 2025 was significantly better than previous models ($p < 0.05$). Furthermore, a linear regression analysis of the False Positive Rate (FPR) between 2021 and 2025 had a statistically declining trend ($R^2 = 0.91$, $p < 0.01$), demonstrating the efficacy of recent architectures to reduce false alarms. These results quantitatively validate the statement that the evolution of ML-based IDS has yielded significant and statistically sound improvements in detection performance and reliability.

4. Datasets in IDS Research

In IDS studies, several datasets have been widely used to train and test detection models. Table 2 shows important IDS datasets employed for training and testing, including their record size and attack types. These datasets, such as KDDCUP'99, NSL-KDD, CSE-CIC-IDS2018, and UNSW-NB15, address various threats ranging from DoS and R2L to Botnet and DDoS, supporting IDS model benchmarking.

Table 2. Commonly Used IDS Datasets for ML-Based Intrusion Detection

Study	Dataset Name	Year	Total Records	Attack Types Included
[9]	KDDCUP’99	1999	4.9M	DoS, Probe, R2L, U2R
[9]	NSL-KDD	2009	1,48,516	DoS, Probe, R2L, U2R
[9]	CSE-CIC-IDS2018	2018	16M	DDoS, DoS, Brute Force, Botnet
[9]	UNSW-NB15	2015	2.5M	9 Attack Types

4.1. Commonly Used Datasets in IDS Research

4.1.1. KDDCUP’99

KDDCUP'99 was obtained from the DARPA 1998 dataset, which was originally designed to test IDS. It has over 4.9 million connection records, all of which are classified as either typical or indicative of a certain kind of attack. Denial of Service (DoS), Probe, Remote to Local (R2L), and User to Root (U2R) are the four main categories into which the assaults fall.

Though it is widely used, KDDCUP'99 has been criticized for having duplicate records, which can skew the performance assessment of IDS models. Moreover, the dataset's age suggests that it might not accurately represent current network traffic patterns and attack vectors [9].

4.1.2. NSL-KDD

Evolved as a better version of KDDCUP'99, the NSL-KDD dataset is relieved of some of the drawbacks of the original dataset by eliminating duplicate instances and possessing a well-balanced training and test instance set. It consists of 125,973 training instances and 22,544 test instances, with each instance being characterized by 41 features. NSL-KDD does possess some enhancements along with some of the drawbacks of KDDCUP'99, including outdated attack forms and a lack of diversity in live traffic [9].

4.1.3. CSE-CIC-IDS2018

To replicate current network traffic and attack situations, the CSE-CIC-IDS2018 dataset was created. It includes both benign traffic and most forms of attacks, such as brute force, intrusion, Distributed Denial of Service (DDoS), and DoS. The data is extensive, with accurate descriptions of feature timestamps, and therefore very suitable for assessing IDS models for the present environment. Its extent and intricacy, however, may prove challenging to handle while processing and analyzing data [9].

4.1.4. UNSW-NB15

The UNSW-NB15 dataset, which was created by the Australian Centre for Cyber Security, is used to more closely mirror modern network traffic. The dataset includes nine kinds of attacks and normal behavior, along with a total of 2.5 million records.

The dataset provides an abundance of features, such as flow-based features and content-based features. Despite its advancement, certain studies have encountered challenges such as imbalanced class distributions and longer attack scenarios [9].

4.2. Limitations of Current Datasets

These datasets have played a crucial role in pushing IDS research forward, but they are not without their limitations:

- Old Attack Scenarios: Most datasets, particularly older ones such as KDDCUP'99 and NSL-KDD, include attack types that are no longer common, making them less useful in testing IDS against modern threats.

- Redundancy and Biases: Some datasets, like KDDCUP'99, are criticized for redundant records, leading to biased performance measurements and overfitting when applied to machine learning models.
- Class Imbalance: In many IDS datasets, the class imbalance between normal instances and attack instances, as well as among multiple attack types, is a recurring problem. Such a class imbalance in the training will bias the process and influence the IDS's response to detecting attack classes that occur less frequently.
- Lack of Realism: Certain datasets lack the richness and diversity of real network traffic, such as encrypted traffic, advanced persistent threats, and insider attacks.

4.3. Need for New Benchmark Datasets

To facilitate improved development and assessment of IDS, there is an urgent need for new benchmark datasets that:

- Capture Contemporary Threat Scenarios: Include recent attack methods, techniques, and tools employed by attackers so that IDS models are tested against applicable threats.
- Provide Data Diversification: Cover diverse network environments, such as varying protocols, applications, and user behavior, to present a complete evaluation ground.
- Address Class Imbalance: Leverage techniques that balance the number of different attack types and legitimate traffic to ensure easier training and testing of IDS models.
- Sustain Data Quality and Anonymity: Avoid noise and artefacts in the data while maintaining anonymity over any confidential information through standard anonymization methodologies.

5. Challenges and Limitations in ML-based IDS

As is clear from the above analysis of research studies, ML-based IDSs have turned out to be a significant component of modern-day cybersecurity solutions with richer features in threat detection and countermeasures. Deployment of ML-IDS in practical applications, however, is accompanied by several challenges and limitations that should be addressed to make them effective and reliable.

The present in-depth analysis discusses vital issues of concern like scalability and real-time processing, data quality and privacy concerns, and adversarial attacks on ML-IDS.

5.1. Scalability and Real-time Processing

With expanding networked infrastructures and increasing volumes of data, ML-IDS must scale effectively without performance degradation. Real-time threat detection scalability is one of the biggest challenges.

5.1.1. Computational Overhead in Large-scale Cloud Environments

ML-IDS is vital in balancing detection precision and resource utilization. From Table 3, various IDS models have different levels of memory utilization, CPU usage, and training time. Classical machine learning models such as Decision Trees consume less computational power, using only 0.5GB of memory and 20% CPU, making them suitable for real-time applications with constrained environments [40].

Likewise, Random Forest, although a bit more involved, still has acceptable computational efficiency with 1.2GB memory usage and 35% CPU usage [41]. SVM, even though it has excellent classification power, does come at a higher cost in terms of resources, using up to 2.5GB of memory and 40% CPU, causing longer training times [41].

Conversely, DL models like CNN and Transformer provide better detection but at the expense of much higher computational requirements. CNNs, which are good at feature extraction, need 4GB of memory and 60% CPU usage; hence, they are computationally intensive [42].

At the same time, Transformers, utilizing self-attention mechanisms, require even more memory and processing power, using 6GB of memory and 75% CPU, with a training time of 150 minutes-the highest among the models compared [43].

These requirements underscore the necessity of distributed computing approaches, model compression methods, and cloud-based parallelization to improve the scalability of ML-IDS without compromising real-time threat detection efficiency [54].

Table 3. Computational resource requirements for IDS models

IDS Model	Memory Usage (GB)	CPU Usage (%)	Training Time (Minutes)	Reference
Decision Tree	0.5	20%	10	[40]
Random Forest	1.2	35%	25	[41]
SVM	2.5	40%	45	[41]
CNN	4	60%	90	[42]

5.2. Data Quality and Privacy Issues

The caliber of training and detection data has a significant impact on ML-IDS effectiveness. System performance is significantly impacted by data privacy and quality.

5.2.1. Imbalanced Datasets and Their Effect on Model Precision

Skewed datasets in which regular traffic overwhelmingly outnumbers malicious examples pose a problem to ML-IDS. The skew can lead to model biases favoring normal traffic with higher false-negative rates. Oversampling minority classes, undersampling majority classes, and using sophisticated algorithms like Synthetic Minority Over-Sampling Technique (SMOTE) have been suggested to counteract this. Research shows that these methods can increase detection rates for minority attack classes, thus improving the overall model accuracy [46].

5.2.2. Privacy-preserving ML Methods

Respecting user privacy when using data for training an ML-IDS is an essential issue. Privacy-preserving methods, including Federated Learning and Differential Privacy, have been devised to deal with this problem.

- Federated Learning: This technique uses local data samples without data interchange, enabling models to be trained across several dispersed devices or servers. Model parameters alone are shared, improving privacy and security. The feasibility of Federated Learning for intrusion detection has been shown through studies, where it has the potential to ensure data privacy with strong detection performance [45].
- Differential Privacy: Differential Privacy adds noise to the data or model parameters to ensure that a single data point cannot be leaked. This technique ensures that changing a single data point will not change the output of the model and thus ensures individual privacy. There has been growing research on the addition of Differential Privacy mechanisms to ML-IDS over the past few years to ensure the balance between model accuracy and privacy [46].

5.3. Adversarial Attacks on ML-based IDS

Adversarial attacks, in which the attackers alter input data to deceive the system, can affect ML-IDS. An understanding of such attacks and awareness of how to improve model robustness is crucial.

5.3.1. Evasion and Poisoning Attacks

- Evasion Attacks: In evasion attacks, attackers devise inputs that are designed specifically to be misclassified by the ML model at the time of testing. For example, slight perturbations in malicious traffic can lead to the IDS marking it as benign. Research has proven that even small changes in input data result in dramatic detection accuracy drops, highlighting the need for resilient models [47].
- Poisoning Attacks: Poisoning attacks include injecting malicious information into the training dataset, thus contaminating the learning process. It can lead to models that are of poor quality or biased towards specific classifications. Research has pointed to the vulnerability of ML-IDS to poisoning attacks, deeming data integrity crucial and the presence of robust data validation protocols [47].

5.3.2. Evasion and Poisoning Attacks

Improving the robustness of ML-IDS against adversarial attacks is an important area of study. Several strategies have been suggested:

- Adversarial Training: This consists of training the model on adversarial examples, allowing it to identify and classify manipulated inputs correctly. Though this method can enhance resistance to particular attacks, it might not generalize effectively to unexpected attack channels [47].
- Ensemble Methods: Using several models and combining their predictions can improve detection performance and yield robustness against adversarial inputs. Ensemble methods help counter a weakness in one model by taking advantage of the diversity of different models [46].
- Feature Selection and Engineering: This can minimize the attack surface exposed to adversaries. Models can be made less vulnerable to manipulation by concentrating on strong and invariant features. There has been research into different feature selection methods to make models more robust against adversarial attacks [46].

6. Challenges and Limitations in ML-based IDS

6.1. Explainable AI in IDS

Incorporating Explainable Artificial Intelligence (XAI) in IDS has focused on making the ML models more interpretable, hence boosting the level of trust and aiding decision-making processes. Classical machine learning models have traditionally worked in a "black box" manner, with minimal room for the security staff to understand why specific detections have been made. XAI methods, including Local Interpretable Model-agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP), have been used in IDS to give explainable, interpretable explanations of model predictions. For instance, an investigation study utilized LIME and SHAP to provide explanations of the output from a black-box model and enhance intrusion detection processes' transparency [48].

6.2. Integration with Zero Trust Architecture

Scaling IDS to be included with Zero Trust Architecture (ZTA) is significant for future cloud security architecture. ZTA is based on the "never trust, always verify" concept and everything within a network should be continuously authenticated and authorized. Integration with ZTA involves developing systems that can scan and monitor absolutely everything in real time so that anything malicious can be identified in real time and action can be taken accordingly. This integration bolsters the security posture by detecting threats and blocking them in real-time, maintaining cloud environments unscathed. Recent research has considered the use of Zero Trust Network IDS (ZT-NIDS), which integrates ZTA concepts with state-of-the-art IDS functionality to deliver end-to-end security solutions (Figure 7) [49].

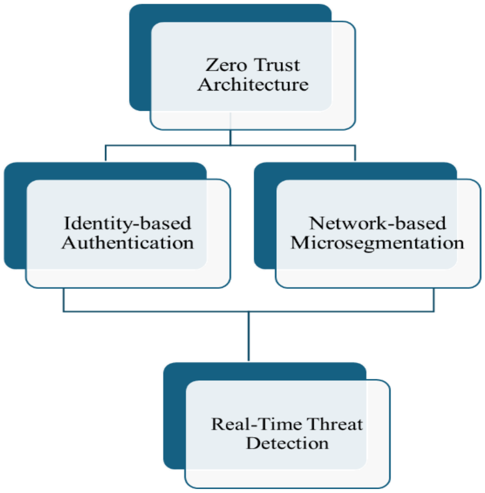
Zero Trust Architecture
Identity-based Authentication
Network-based Microsegmentation
Real-Time Threat Detection

Fig. 7 Zero trust architecture

6.3. Adaptive and Self-Learning IDS Models

The evolving nature of cyberattacks requires that IDS learn and evolve again and again. Adaptive and self-adapting IDS models possess internal learning mechanisms that improve their knowledge database upon every exposure to new threats. The mechanism enables the system to detect and react to newly identified patterns of attacks, thus protecting the system. Various approaches to implementing continuous learning on IDS have been studied via studies focusing on model accuracy and adapting to changing threat landscapes. For example, an investigation outlined a self-adaptive and decentralized intrusion detection approach that employs continuous learning and self-adaptive neural networks to overcome the limitations of conventional IDS [50].

7. Conclusion

The ever-changing and dynamic nature of the cyber threat landscape has created a need for IDS capable of detecting and preventing sophisticated attacks on cloud infrastructure. The traditional IDS mechanism employing static rule-based systems is unable to cope with the dynamic and enormous size of today's cloud infrastructure. Thus, the introduction of ML in IDS has been a revolutionary measure with increased flexibility, automation, and improved detection.

The current study has explored different ML approaches applied in IDS, from supervised learning and unsupervised learning to deep learning techniques. Decision Trees, SVM, and Random Forests are supervised learning methods that are accuracy-intensive but involve enormous sets of labeled data, which are mostly not readily accessible. Unsupervised learning techniques, such as clustering techniques, K-Means, and DBSCAN, address the issue of unknown attacks but are usually plagued by too many false positives. Deep models such as CNNs, RNNs, and LSTM networks have enormous potential to identify complex attack signatures but require behemoth computational resources and gigantic training datasets.

Even with such augmentations, ML-based IDS suffers from certain inherent issues that need to be resolved before going mainstream for general employment in cloud security. Scalability and real-time processing are issues, and massive-scale cloud infrastructure does see humongous traffic flooding into the IDS models. The data integrity regarding skewed datasets and vintage attack vectors also affects ML-based detection. Privacy issues, specifically in user-sensitive data handling, are also responsible for the intricacy of IDS deployment with applications like federated learning and differential privacy required to render data use secure.

In addition, adversarial attacks pose the maximum threat to ML-based IDS because they enable adversaries to manipulate input data to escape detection or minimize model accuracy. Adversarial training, ensemble strategies, and adversarial robust feature engineering are a few defense measures proposed to handle these attacks. Yet, rigorous research has to be conducted to develop adversary-proof IDS to counter highly advanced, adapting adversarial attack methods.

The survey also provides emerging trends that will shape the future of ML-based IDS. Explainable AI (XAI) is picking up speed as a way to enhance model interpretability, making it possible for security analysts to comprehend and rely on automated choices. Integration of IDS with Zero Trust Architecture (ZTA) is another promising trend, providing continuous authentication and real-time anomaly detection within cloud networks. In addition, adaptive and self-learning IDS models are also significant advancements that make it possible to live updates and continuous learning to proactively combat new threats.

In the future, research must be focused on developing IDS solutions that maximize detection accuracy, computational complexity, and practical applicability. The development of more thorough, varied, and current benchmark datasets will play a key role in the evolution of IDS research. Interdisciplinary research collaboration between cybersecurity researchers, cloud service providers, and regulators will also play an important part in standardizing ML-based IDS deployment for the cloud.

Though ML-based IDS have boosted the security of cloud infrastructure greatly, more innovation is still urgently required to advance their scalability, durability, and elasticity. By responding to the prevailing challenges and incorporating future trends, the cyber protection community can enable stronger and smarter intrusion detection systems to better protect cloud systems against sophisticated intrusions.

References

[1] Caitlin Harris, 50 Cloud Security Stats You Should Know In 2025, Expert Insights, 2025. [Online]. Available: https://expertinsights.com/insights/50-cloud-security-stats-you-should-know/

[2] Grace Lau, 40+ Alarming Cloud Security Statistics for 2025, StrongDM, 2025. [Online]. Available: https://www.strongdm.com/blog/cloud-security-statistics

[3] Mikel K. Ngueajio et al., "Intrusion Detection Systems Using Support Vector Machines on the KDDCUP'99 and NSL-KDD Datasets: A Comprehensive Survey," *Proceedings of SAI Intelligent Systems Conference*, Amsterdam, The Netherlands, pp. 609-629, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Muhammad Azmi Umer et al., "Machine Learning for Intrusion Detection in Industrial Control Systems: Applications, Challenges, and Recommendations," *International Journal of Critical Infrastructure Protection*, vol. 38, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Md Liakat Ali et al., "Deep Learning vs. Machine Learning for Intrusion Detection in Computer Networks: A Comparative Study," *Applied Sciences*, vol. 15, no. 4, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[6] Mudita Uppal et al., "Enhancing Accuracy Through Ensemble Based Machine Learning for Intrusion Detection and Privacy Preservation Over the Network of Smart Cities," *Discover Internet of Things*, vol. 5, no. 1, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[7] Chaoyu Zhang et al., "Machine Learning-Based Intrusion Detection Systems: Capabilities, Methodologies, and Open Research Challenges," *Authorea Preprints*, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[8] Sunil Kaushik et al., "Robust Machine Learning Based Intrusion Detection System Using Simple Statistical Techniques in Feature Selection," *Scientific Reports*, vol. 15, no. 1, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[9] Ansam Khraisat et al., "Survey of Intrusion Detection Systems: Techniques, Datasets and Challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1-22, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[10] Ahmet Efe, and İrem Nur Abacı, "Comparison of the Host Based Intrusion Detection Systems and Network Based Intrusion Detection Systems," *Celal Bayar University Journal of Science*, vol. 18, no. 1, pp. 23-32, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] Elena Fedorchenko, Evgenia Novikova, and Anton Shulepov, "Comparative Review of the Intrusion Detection Systems Based on Federated Learning: Advantages and Open Challenges," *Algorithms*, vol. 15, no. 7, pp. 1-26, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] N-able, Intrusion Detection System (IDS): Signature vs. Anomaly-Based - N-able, 2021. [Online]. Available: https://www.n-able.com/blog/intrusion-detection-system

[13] OTORIO, Intrusion Detection Systems (IDS): Pros and Cons, 2025. [Online]. Available: https://www.otorio.com/blog/intrusion-detection-systems-ids/

[14] Rapid7, The Pros & Cons of Intrusion Detection Systems, 2017. [Online]. Available: https://www.rapid7.com/blog/post/2017/01/11/the-pros-cons-of-intrusion-detection-systems/

[15] Aria Cybersecurity Solutions, Understanding the Strengths and Limitations of Your Intrusion Detection System, 2019. [Online]. Available: https://blog.ariacybersecurity.com/blog/understanding-the-strengths-and-limitations-of-your-intrusion-detection-system

[16] Nari Sivanandam Arunraj et al., "Comparison of Supervised, Semi-supervised and Unsupervised Learning Methods in Network Intrusion Detection System Application," *Applications and Concepts of Business Informatics*, no. 6, pp. 10-19, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[17] Pavel Laskov et al., "Learning Intrusion Detection: Supervised or Unsupervised?," *International Conference on Image Analysis and Processing*, Cagliari, Italy, pp. 50-57, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[18] Chuanliang Chen, Yunchao Gong, and Yingjie Tian, "Semi-Supervised Learning Methods for Network Intrusion Detection," *2008 IEEE International Conference on Systems, Man and Cybernetics*, Singapore, pp. 2603-2608, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[19] Richard Kimanzi et al., "Deep Learning Algorithms Used in Intrusion Detection Systems-A Review," *arXiv Preprint*, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[20] Muhammad Sajid et al., "Enhancing Intrusion Detection: A Hybrid Machine and Deep Learning Approach," *Journal of Cloud Computing*, vol. 13, no. 1, pp. 1-24, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[21] Steven Ning et al., "The Study of Feature Engineering in Machine Learning and Deep Learning for Network Intrusion Detection Systems," *2024 Silicon Valley Cybersecurity Conference (SVCC)*, Seoul, Korea, pp. 1-5, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Abiodun Ayantayo et al., "Network Intrusion Detection Using Feature Fusion with Deep Learning," *Journal of Big Data*, vol. 10, no. 1, pp. 1-24, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[23] Abdalla Fa Belhagi, and Elshrif Ibrahim Elmurngi, "Intrusion Detection System Using Supervised Learning Techniques," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 6, no. 12, pp. 4281-4289, 2024. [Google Scholar] [Publisher Link]

[24] Shahadat Uddin et al., "Comparing Different Supervised Machine Learning Algorithms for Disease Prediction," *BMC Medical Informatics and Decision Making*, vol. 19, no. 1, pp. 1-16, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[25] Tala Talaei Khoei, and Naima Kaabouch, “A Comparative Analysis of Supervised and Unsupervised Models for Detecting Attacks on the Intrusion Detection Systems,” *Information*, vol. 14, no. 2, pp. 1-14, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[26] Sneha Leela Jacob, and Parveen Sultana Habibullah, “A Systematic Analysis and Review on Intrusion Detection Systems Using Machine Learning and Deep Learning Algorithms,” *Journal of Computational and Cognitive Engineering*, vol. 4, no. 2, pp. 108-120, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[27] George Sarossy, “*Anomaly Detection in Network Data with Unsupervised Learning Methods*,” Bachelor Thesis, Mälardalen University, School of Innovation, Design and Engineering, 2021. [Google Scholar] [Publisher Link]

[28] Prabu Kaliyaperumal et al., “Harnessing DBSCAN and Auto-Encoder for Hyper Intrusion Detection in Cloud Computing,” *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 5, pp. 3345-3354, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[29] Woo-Hyun Choi, and Jongwon Kim, “Unsupervised Learning Approach for Anomaly Detection in Industrial Control Systems,” *Applied System Innovation*, vol. 7, no. 2, pp. 1-16, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[30] Taehoon Kim, and Wooguil Pak, “Deep Learning-Based Network Intrusion Detection Using Multiple Image Transformers,” *Applied Sciences*, vol. 13, no. 5, pp. 1-15, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[31] Sydney Mambwe Kasongo, “A Deep Learning Technique for Intrusion Detection System Using a Recurrent Neural Networks based Framework,” *Computer Communications*, vol. 199, pp. 113-125, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[32] Zhenyue Long et al., “A Transformer-Based Network Intrusion Detection Approach for Cloud Security,” *Journal of Cloud Computing*, vol. 13, no. 1, pp. 1-11, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[33] Estabraq Saleem Abduljabbar Alars, and Sefer Kurnaz, “Enhancing Network Intrusion Detection Systems with Combined Network and Host Traffic Features Using Deep Learning: Deep Learning and IoT Perspective,” *Discover Computing*, vol. 27, no. 1, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[34] Mohammed F. Suleiman, and Biju Issac, “Performance Comparison of Intrusion Detection Machine Learning Classifiers on Benchmark and New Datasets,” *2018 28th International Conference on Computer Theory and Applications*, Alexandria, Egypt, pp. 19-23, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[35] Fatima Isiaka, “Performance Metrics of an Intrusion Detection System through Window-Based Deep Learning Models,” *Journal of Data Science and Intelligent Systems*, vol. 2, no. 3, pp. 174-180, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[36] Sudhanshu Sekhar Tripathy, and Bichitrananda Behera, “Performance Evaluation of Machine Learning Algorithms for Intrusion Detection System,” *Cryptology ePrint Archive*, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[37] Syed Ali Raza Shah, and Biju Issac, “Performance Comparison of Intrusion Detection Systems and Application of Machine Learning to Snort System,” *Future Generation Computer Systems*, vol. 80, pp. 157-170, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[38] Sydney M. Kasongo, and Yanxia Sun, “Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset,” *Journal of Big Data*, vol. 7, no. 1, pp. 1-20, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[39] Ibtihal Mohammed Khaleel, “A Comparative Performance Evaluation of Network Intrusion Detection Using Machine and Deep Learning Algorithms,” *Mathematics for Applications*, vol. 13, no. 2, pp. 1-12, 2024. [Google Scholar] [Publisher Link]

[40] Chadia E. L. Asry et al., “A Robust Intrusion Detection System Based on a Shallow Learning Model and Feature Extraction Techniques,” *PlosOne*, vol. 19, no. 1, pp. 1-31, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[41] Yaping Chang, Wei Li, and Zhongming Yang, “Network Intrusion Detection Based on Random Forest and Support Vector Machine,” *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing*, Guangzhou, China, pp. 635-638, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[42] Kang Leng Chiew, and Bian Hui, “An Improved Network Intrusion Detection Method Based on CNN-LSTM-SA,” *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 44, no. 1, pp. 225-238, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[43] Hamza Kheddar, “Transformers and Large Language Models for Efficient Intrusion Detection Systems: A Comprehensive Survey,” *arXiv Preprint*, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[44] V. Kantharaju et al., “Machine Learning Based Intrusion Detection Framework for Detecting Security Attacks in Internet of Things,” *Scientific Reports*, vol. 14, no. 1, pp. 1-10, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[45] Shaashwat Agrawal et al., “Federated Learning for Intrusion Detection System: Concepts, Challenges and Future Directions,” *arXiv Preprint*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[46] Brunel Rolack Kikissagbe, and Meddi Adda, “Machine Learning-Based Intrusion Detection Methods in IoT Systems: A Comprehensive Review,” *Electronics*, vol. 13, no. 18, pp. 1-23, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[47] Huda Ali Alatwi, and Charles Morisset, “Adversarial Machine Learning in Network Intrusion Detection Domain: A Systematic Review,” *arXiv Preprint*, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[48] Diogo Gaspar, Paulo Silva, and Catarina Silva, “Explainable AI for Intrusion Detection Systems: LIME and SHAP Applicability on Multi-Layer Perceptron,” *IEEE Access*, vol. 12, pp. 30164-30175, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[49] Abeer Alalmaie, Priyadarsi Nanda, and Xiangjian He, "ZT-NIDS: Zero Trust, Network Intrusion Detection System," *Proceedings of the 20th International Conference on Security and Cryptography SECRYPT*, Rome, Italy, vol. 1, pp. 99-110, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[50] Ahmed Abubakar Aliyu, Jinshuo Liu, and Ezekia Gilliard, "A Decentralized and Self-Adaptive Intrusion Detection Approach Using Continuous Learning and Blockchain Technology," *Journal of Data Science and Intelligent Systems*, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[51] Anna L. Buczak, and Erhan Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153-1176, 2015. [CrossRef] [Google Scholar] [Publisher Link]