International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

# Intrusion Detection Systems: Development, Advancements, and Limitations

**Ms. Amisha Jain**
Assistant Professor, Swami Vivekanand College of Engineering Indore Madhya Pradesh
**amishajain0822@gmail.com**
**Ms. Srashtika Gupta**
Assistant Professor, Swami Vivekanand College of Engineering Indore Madhya Pradesh
**Srashtikaagupta@gmail.com**

**Abstract**
Intrusion Detection Systems (IDS) have evolved significantly over the past decades in response to the increasing complexity and frequency of cyber threats. Early IDS technologies were primarily signature-based, focusing on known attack patterns, which limited their ability to detect novel and sophisticated intrusions. To overcome these constraints, anomaly-based and specification-based IDS models were introduced, enabling improved detection of unknown attacks by identifying deviations from normal system behavior. With the advancement of Internet of Things (IoT), Industrial Control Systems (ICS), and Cyber–Physical Systems (CPS), IDS technologies have further incorporated machine learning and deep learning techniques to handle high-dimensional data and dynamic network environments. Despite these advancements, IDS solutions still face several limitations, including high false-positive rates, computational overhead, scalability challenges, real-time deployment constraints, and privacy concerns in distributed systems. This section critically examines the evolutionary phases of IDS technologies and highlights their inherent limitations, emphasizing the need for adaptive, lightweight, and privacy-preserving IDS frameworks for modern network infrastructures.

**Keywords-**Intrusion Detection System; Signature-Based IDS; Anomaly-Based IDS; Machine Learning; Deep Learning; IoT Security; Cyber–Physical Systems; False Positives; Network Security

## I INTRODUCTION

Evolution and limitations of IDS technologies.

In the last five years, network security has permeated nearly every aspect of people's life. Tools and strategies used to defend and secure networks are now the main focus of computer security discourse.[1] Discovering, evaluating, and reporting hostile activity has its roots in the past, and this article aims to trace those roots as well as present and future directions. In addition, they will delve into some of the numerous methods and tools that are being employed in network defense. In order to protect and keep tabs on computer networks, there are a number of technologies that

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

deliver a certain amount of peace of mind with tolerable risks. Protection during Depth refers to all the things needed to accomplish this goal, including thorough analyst training, strategically placed hardware, and a solid security policy.[2] We can accomplish this using the resources we have at our disposal every day. Network devices, the host computer, security software, virus scanners, and an intrusion detection system—a tool specifically designed to detect known attacks—contribute to the data aggregation. Actively seeking out evidence of intrusive actions is what it is all about. Computer and network intrusion detection covers a lot more ground than just that. There are a lot of steps involved in finding out whether someone is using a network or computer equipment without permission. All of this is made possible by purpose-built software that can identify suspicious or out-of-the-ordinary behavior.[3-4]

**The beginning**

James P. Anderson detailed the United States Air Force's (USAF) "becoming increasingly aware of computer security problems" in a document issued in October 1972. Every part of the United States Air Force's operations and administration suffered the effects of this crisis. The United States Air Force faced a significant issue at the time in maintaining a need-to know environment for its classified computer systems, which allowed users with different levels of clearance to access and use the systems. That caused a major issue that has persisted for the past 30 years. The essential question is: how can they ensure the security of distinct classification domains on a same network? Improve computer security auditing and monitoring at client sites with the help of a study by James P. Anderson, released in 1980. His work on "How to use accounting audit files to detect unauthorized access" is widely considered to be the inspiration for automatic ID. Mainframe systems now have a new tool for detecting abuse according to this ID study.[5] They needed to identify the dangers before we could do anything else. They need to know what kinds of attacks and threats can be launched against computer systems and how to spot them in audit data before you can build an intrusion detection system. Actually, he was likely making a reference to the necessity of a risk assessment strategy in order to comprehend the danger (i.e., the strengths and weaknesses of the system, potential attack vectors, and penetration methods), and subsequently to develop a security policy to safeguard the existing systems. In 1984 and 1986, Dorothy Denning and Peter Neumann conducted research and produced the initial concept of real-time intrusion detection systems. They called this first version the Intrusion Detection Expert System (IDES). At one point, this IDES was a trained expert system that relied on rules to identify specifically malicious actions. The current term for this upgraded system is the Next-Generation Intrusion Detection Expert System, or NIDES.[6-7] For the most part, IDES research in the '80s and '90s was based on James P. Anderson's published report and IDES work. Most of this study's funding came from the federal government of the United States over this timespan.

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

Network Audit Director and Intrusion Reporter, Discovery, Haystack, MIDAS, and NADIR are just a few of the many systems developed for detecting and notifying hackers.[8]

**Misuse detection model:** The intrusion detection system is alert to potentially malicious actions and actively seeks out any instances of policy violations. Searching for known harmful or undesirable behavior is also part of it. Actually, its effectiveness and relatively low false alarm rate are its primary characteristics. The ID field has experienced significant growth in recent years, leading to the development of numerous IDS to meet specific demands. Today, abuse detection solutions are the market leaders, although the original ID systems were designed to detect anomalies. Identification is now crucial due to the ever-increasing quantity of interconnected computer systems. Items marketed to the general public first appeared in the mid-1990s. In the mid-1990s, RealSecure from Internet Security Systems and Netranger from Wheelgroup were two of the most widely used intrusion detection systems (IDS). The two businesses both began with IDS that was based on their networks. In October 1995, Wheelgroup was established with the purpose of commercializing Netranger, a security solution that had been prototyped by the United States Air Force at the time. This tool analyzes network data for signs of abuse, alerting users in real-time and providing information about sneaky assaults. After being acquired by Cisco in February 1998, Wheel group would later become an essential component of Cisco's security architecture. With the initial version of the Internet Scanner designed and released by Christopher Klauss in April 1994, Thomas Noonan and Klauss established Internet Security Systems, Inc. (ISS). Released on December 9, 1996, RealSecure was ISS's solution to enhance network security with real-time threat identification. A new commercial breakthrough, RealSecure 1.0 for Windows NT 4.0, was introduced on August 19, 1997. It was their first intrusion detection system. Keep in mind that the majority of systems on the market today are knowledge-based, which means they detect and respond to known threats by comparing signatures of these attacks to changes in systems or packet streams on a network. On the other hand, one of their biggest flaws is that they aren't always up to snuff when it comes to new types of assaults; as a result, they need to be updated often with information on attack signatures. While these false positives are typical of behavior-based intrusion detection systems, so is their capacity to uncover attacks that have gone undetected.[9-14]

**Types of IDS Detection**

There are five types of IDS: network-based, host-based, protocol-based, application protocol-based and hybrid.

**The two most common types of IDS are:**

**Network-based intrusion detection system (NIDS)**

The interconnection of Network security systems keeps tabs on an entire protected network. Its deployment across the infrastructure occurs at key junctures, like the most susceptible

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

subnets. From packet contents and information, the NIDS deduces what devices are connecting to and from the network.

**Host-based intrusion detection system (HIDS)**

An intrusion detection system (IDS) that is installed on a host computer keeps tabs on that computer. In simpler terms, it is used to safeguard a particular endpoint from both internal and external dangers. For this purpose, the IDS analyzes incoming and outgoing data, keeps track of any suspicious behavior, and alerts the appropriate parties.

**Protocol-based (PIDS)** Most often, a web server will have an intrusion detection system installed on it that is protocol-based. A user's or device's protocol traffic to and from the server is tracked and analyzed by it. To keep tabs on the protocol's status and activity, a PIDS often sits at the server's front end.[15]

**Application protocol-based (APIDS)**

Server parties often have APIDSs, which are systems or agents, installed within. The system monitors and deciphers messages sent over protocols designed for certain applications. To illustrate, this would keep an eye on the SQL protocol as it communicates with the middleware and the web server.[16]

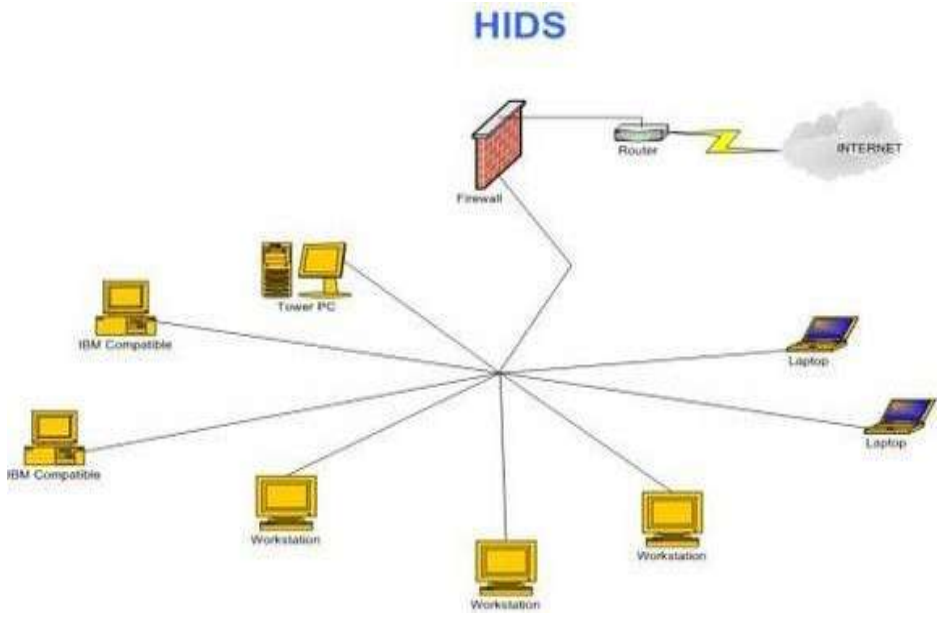


Figure 1 Host Based intrusion detection

Machine Learning in Cyber security

There are three steps in the security lifecycle: detection, reaction, and prevention [17]. It is acknowledged that entirely preventing cyber threats is an unrealistic undertaking, whereas the

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

reaction phase operates under the assumption that harm has already occurred. Consequently, threat detection is the primary goal of most security methods, including those that rely on ML. For example, while it's impossible to stop the construction of phishing webpages, you can stop people from falling for them by detecting compromised webpages and notifying them before they fall for a phishing "hook."

Two separate methods exist for identifying cyber threats: anomaly based and misuse based. The first type, sometimes called signature or rule based, works by assuming that future threats would display the same patterns that were defined for a particular threat. The second set of rules assumes that security incidents are associated with events that deviate from the norm and seeks to identify them. This approach requires establishing a concept of "normality" as its starting point. Both of these methods of detection work well together: Anomaly-based methods are more likely to produce false alarms but have a better possibility of detecting new assaults, in contrast to misuse-based methods, which are extremely accurate but can only identify known threats.[18]

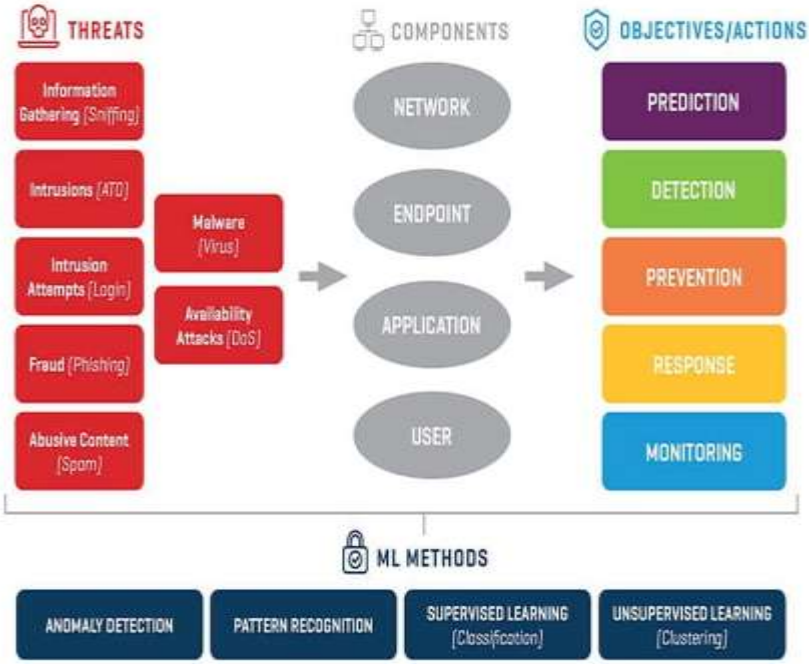


Figure 2 Machine learning approach to cyber threats

Prior to the development of ML, each approach's detection mechanisms—whether they were based on misuse or anomalies—required the manual description of all the relevant parts. Not only were such endeavors inefficient and prone to mistakes, they also couldn't keep up with the exponential expansion of today's settings. With the advancement of data analytics techniques, detection systems started to use data-driven solutions, including ML. Not only did these

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

alternatives reduce the need for human intervention, but they occasionally even beat more conventional methods of handwriting recognition [19]. When applied to ML, this improved performance is a result of the algorithm's inherent capacity to learn "weak" signals—those that go unnoticed by human operators—in the processed data and incorporate them into an improved detection system.[20]

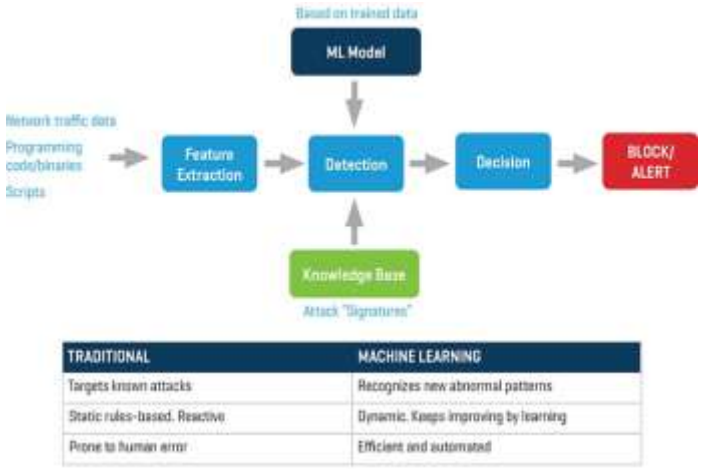


Figure 3 machine learning vs. traditional methods in cyber security

Table summarizing the key points related to machine learning in cyber security
Table 1 the performance of anomaly-based Network Intrusion Detection Systems (NIDS):[21]

| Aspect | Description | Advantages | Challenges |
|---|---|---|---|
| Security Lifecycle | Cybersecurity consists of three steps: detection, reaction, and prevention. Detection is the primary goal of most security methods, especially with ML. | Focus on detecting threats in real-time before causing harm. | Prevention is unrealistic, and reaction occurs after harm has occurred. |
| Detection Methods | Two primary methods for detecting threats: anomaly-based and misuse-based (signature/rule-based). | Combination of both methods yields better overall results. | Anomaly-based methods generate more false alarms, while misuse-based methods can only identify known threats. |
| Anomaly- | This method detects | Effective at detecting | High false alarm rates |

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

| | | | |
|---|---|---|---|
| based Detection | abnormal activities deviating from the "normal" baseline and identifies previously unknown threats. | new, unseen threats. | due to misidentification of benign activities as threats. |
| Misuse-based Detection | Detects known threats by recognizing pre-defined patterns or signatures of attacks. | High accuracy in detecting known threats. | Can only detect previously identified threats, leaving unknown threats undetected. |
| Manual Approach Pre-ML | Traditional methods involved manually defining all patterns or rules for detection, both for anomaly and misuse. | Detailed and domain-specific understanding of attack patterns. | Inefficient, prone to errors, and unable to scale with the exponential growth of data. |
| ML in Threat Detection | ML automates and improves threat detection by analyzing data for hidden signals that humans may miss, leading to more accurate and less labor-intensive systems. | Reduces human intervention, enhances performance by recognizing subtle "weak signals," and adapts to changes. | Requires large datasets and extensive training for effective performance. |
| Impact of Data Analytics | With data-driven solutions, particularly through ML, threat detection becomes more scalable and faster, handling large-scale data more efficiently. | Enables faster, real-time detection of both known and unknown threats. | May struggle with adapting to very new, evolving threats unless trained with up-to-date datasets. |
| Performance of ML Systems | ML has been shown to outperform traditional handwriting recognition and conventional methods in certain cases by learning from data and adapting continuously. | Can outperform traditional methods due to continuous learning and adapting. | Performance depends heavily on the quality and quantity of training data. |

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

**Overview of machine learning techniques for IDS.**

Interactions between various components are characteristic of Control Systems (CS) networks. A malicious actor well-versed in computer systems, operating systems, and networks could use these vulnerabilities to gain access to the control system without authorization and carry out damaging acts within it. Among the various sorts of attacks that a network may face, three significant risks warrant consideration: DOS, Spoofing, and Eavesdropping. In a DOS attack, the assailant overwhelms the network with a flood of either legitimate or illegitimate communications, therefore compromising the availability of network resources. Spoofing is a method of mimicking a real user in order to get access to protected services and information on a network. Data transmitted over a network is vulnerable to eavesdropping and other forms of data breach. Unauthorized parties pose a significant threat of eavesdropping in wireless networks, as adversaries can capture transmissions from a considerable distance, extending beyond the organization's grounds. The continuous struggle between attackers and IDS has prompted considerable progress in security protocols. Nonetheless, it has also engendered progressively nuanced and challenging-to-detect attack methodologies. Here are few significant advantages conferred by machine learning in the realm of network cybersecurity:[22]

**Threat detection**: To better monitor communication networks for suspicious activity or cyber threats, prediction models trained using machine learning can be developed. Algorithms like this may sift through mountains of data in real time, uncovering patterns and outliers associated with harmful actions in things such as metadata, user actions, and packet data.

**Automation of attack responses**: Automation is crucial for the security of communication networks. Automated assault responses made possible by machine learning techniques provide quick and efficient responses. Without human interaction, machine learning systems can be trained to identify specific types of assaults and execute corrective measures, such as removing infected devices from circulation or upgrading security policies.

**Detect new types of attacks**: Every time cyber dangers change, new kinds of cyberattacks pop up. Perhaps the new dangers are too complicated for the old signature-based method to identify. When new types of attacks emerge, machine learning algorithms can identify them by looking for unusual patterns or behaviors, even if no obvious warning indicators are present.

**Reduce False Positives**: Conventional security systems frequently produce several false positives, erroneously indicating typical activity as an assault. This may result in the squander of time and important resources in addressing irrelevant reports. With the use of machine learning models, security operations can become more efficient while true threats can be more accurately identified, all while reducing the number of false positives.

**Adaptation and continuous learning** When new dangers or circumstances arise in communication networks, machine learning models may be instantly updated to reflect these

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

changes. Through ongoing learning, models can enhance over time, acquiring deeper insights into risks and their variations.[23]

Table 2 machine learning techniques for IDS.

| Parameter | Description | Advantages in IDS |
|---|---|---|
| Threat Detection | Utilizes machine learning algorithms to sift through large volumes of network data to identify suspicious patterns and activities. | Enhances the ability to detect cyber threats like DoS, spoofing, and eavesdropping in real time. |
| Automation of Attack Responses | Machine learning enables automated responses to detected threats, without requiring human intervention, such as isolating compromised devices or updating security policies. | Increases efficiency and speed in mitigating attacks, reducing manual oversight. |
| Detection of New Attack Types | Machine learning algorithms are capable of identifying new, unseen attack patterns that may not be detected by signature-based methods. | Enables IDS to adapt to evolving cyber threats and recognize unknown attack strategies. |
| False Positive Reduction | Machine learning improves the identification of true threats and reduces false alarms, distinguishing between genuine threats and normal activities. | Reduces wasted resources and false alerts, increasing the overall efficiency of security systems. |
| Adaptation and Continuous Learning | Machine learning models are able to continuously evolve and adapt to changing attack patterns and new cybersecurity strategies. | Enhances the accuracy and flexibility of IDS as it adjusts to new and dynamic network conditions. |

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

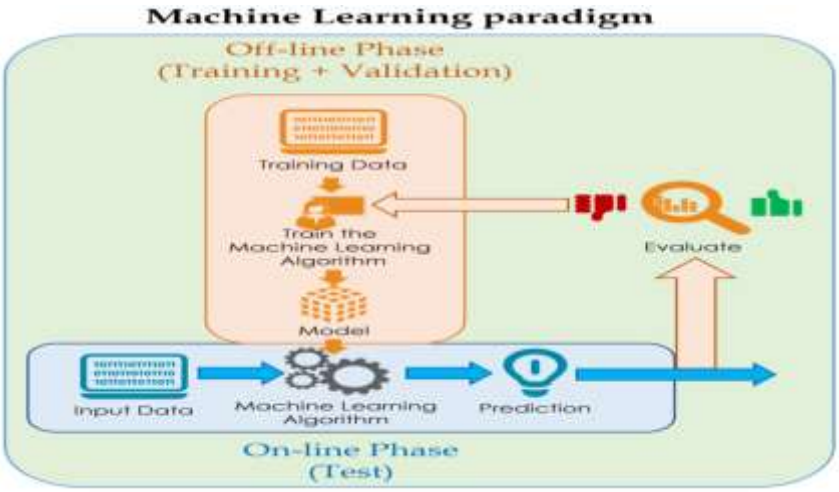


**Fig. 2.4 Schematic representation of the machine learning workflow.**

Ultimately, employing Artificial intelligence for securing communication networks has several advantages, such as real-time threat detection, automated responses, identification of novel attack vectors, and a reduction in false positives. These advantages enhance overall network security and safeguard fundamental data and assets.[24]

As illustrated in Figure 1, the machine learning paradigm comprises the following primary steps:

**Data Collection**: One of the initial steps is to gather training data. These datasets include you can find labelled instances, which are sets of inputs and outputs that match. For example, while building a model to recognize cat photos, the dataset will include both "cat" images and other images that are labeled as "not cat."

**Data preparation**: This phase entails the cleansing, normalization, and transformation of the training data to render it appropriate for processing by the machine learning model. This may involve removing absent data, addressing categorizing variables, and standardizing numerical values.

**Model selection and training**: The data must be cleaned, normalized, and transformed before the machine learning model can be trained. This is done in this phase. Eliminating missing data, dealing with variable categorization, and standardizing numerical values are all possible steps in this direction. Finding the right machine learning model to solve the problem is what this phase is all about. Training the model on this set of data allows it to eventually recognize the underlying correlations and patterns. The training process involves iteratively refining the model to align its predictions more closely with the corresponding output labels in the training dataset.

**Model Evaluation**: Following training, the model is assessed with new, unrelated test data. This lets us see how effectively the model handles pattern recognition in fresh data. Precision,

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

accuracy, and area under the ROC curve are only a few of the metrics used to assess a model's performance.

**Model Usage**: The next step is to use the trained and evaluated model to make predictions using new input data. In order to forecast future input instances, the model makes use of the connections learned throughout training.

**Data Collection Procedure and Dataset Description**

This study utilized the CICIDS2017 dataset, developed by the Canadian Institute for Cyber security, which encompasses various scenarios of network attacks. Due to its large scale and detailed structure, CICIDS2017 is a key resource for designing and evaluating new models and algorithms aimed at mitigating network intrusions. The dataset includes eight distinct files, covering five days of both normal and attack-related network traffic data, provided by the Canadian Institute for Cybersecurity. In total, the dataset contains 2,830,743 records, each described by 79 different features (25]Evaluation dataset for intrusion detection (CIC-IDS2017)

When it comes to protecting networks from the increasingly complex and numerous intrusion attempts, intrusion prevention systems and intrusion detection systems are the best bets. Performance improvements in anomaly-based intrusion detection systems have been steady and accurate, but this has been hindered by a dearth of trustworthy test and validation datasets. Based on our analyses, the majority of the eleven datasets that have been available since 1998 are inaccurate and out of date. A number of these datasets have issues, such as insufficient traffic diversity and volumes, incomplete coverage of known attacks, or anonymization of packet payload data, making them unable to reflect current trends. Not all of them have the necessary metadata and feature set.

The CICIDS2017 dataset is representative of actual real-world data (PCAPs) since it includes both benign and recently-discovered frequent threats. Included as well are the CSV files with the findings of the network traffic analysis performed using CICFlowMeter. The flows have been categorized according to timestamps, source and destination IPs, ports, protocols, and attacks. When we were building this dataset, producing realistic background traffic was our number one aim. They have created realistic, harmless background traffic by utilizing my suggested B-Profile method to profile the abstract behavior of human interactions (Sharafaldin, et al. 2016). In this dataset, we constructed the abstract behavior of 25 users using protocols such as HTTP, HTTPS, FTP, SSH, and email. [26]

For a total of five days, beginning at nine in the morning on Monday, July 3, 2017, and ending at five in the afternoon on Friday, July 7, 2017, data was collected. On Monday, everything is back to normal, including the mild traffic. Web Attack, Infiltration, Brute Force FTP, Brute Force SSH, Denial of Service, Heartbleed, Botnet, and Distributed Denial of Service are some

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

of the attacks that have been put into action. They were put to death on Tuesday, Wednesday, Thursday, and Friday at both the morning and afternoon sessions. Eleven requirements for constructing a trustworthy benchmark dataset have been outlined in our most current dataset evaluation methodology [27]Up till now, no IDS dataset has been able to meet all eleven requirements. What follows is a synopsis of these requirements:

**Complete Network configuration:** The existence of multiple operating systems, including Windows, Ubuntu, and Mac OS X, as well as a modem, firewall, switches, and routers, constitute a comprehensive network topology.

- **Complete Traffic:** With the use of an agent that profiles users, twelve separate computers in the Victim-Network, and actual assaults in the Attack-Network.
- **Labelled Dataset:** The benign and assault designations for each day are displayed in Section 4 and Table 2, respectively. The dataset document will also provide the specifics of when the assault occurred.
- **Complete Interaction:** Having two separate networks and Internet connectivity allowed us to cover both within and between the internal LAN, as shown in Figure 1.
- **Complete Capture:** The usage of the mirror port, also known as the tapping system, allowed us to record and collects all communication on the storage server.
- **Available Protocols:** Ensured the availability of all widely used protocols, including HTTP, HTTPS, FTP, SSH, and email.
- **Attack Diversity:** This dataset includes the most prevalent assaults as reported in the 2016 McAfee study. These attacks include web-based, brute-force, denial-of-service, distributed denial-of-service, infiltration, heartbleed, bot, and scan.
- **Heterogeneity:** Recorded system calls, memory dumps, and network activity from the affected workstations as well as the primary switch during the attack's execution.
- **Feature Set:** Using CICFlowMeter, they were able to extract 80+ network flow features from the generated network traffic and provide the dataset in a CSV format. Take a look at their CSV generator and PCAP analyzer.
- **MetaData:** Provided a detailed explanation of the dataset in the published publication, covering time, attacks, flows, and labels.[28]

Table 3 **CIC-IDS-2017 dataset's classes (attack types)**

| Attack Category | Specific Attack Types |
|---|---|
| Benign (Normal) | Normal traffic (no attack) |
| Brute Force | FTP-Patator, SSH-Patator |
| DoS (Denial of Service) | DoS-Slowloris, DoS-GoldenEye, DoS-Hulk, DoS-Slowhttptest |
| DDoS (Distributed DoS) | DDoS attack |

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

| | |
|---|---|
| Botnet | Bot attack traffic |
| Web Attacks | Web Brute Force, Web XSS, Web SQL Injection |
| Infiltration | Infiltrating the network |
| Port Scanning | PortScan |
| Heartbleed | Exploiting Heartbleed vulnerability |

Table 4 **Features of CIC-IDS-2017 Dataset**

| Category | Feature Name |
|---|---|
| Basic Features | Destination IP, Source IP, Source Port, Destination Port, Timestamp |
| Flow-based Features | Flow Duration, Flow Bytes/s, Flow Packets/s |
| Packet-level Stats | Information on forward packets, backward packets, total lengths of forward packets and backward packets, minimum and maximum packet lengths, average and standard packet lengths, and packet length variance |
| Time-based Features | future iat min, future iat max, future iat mean, future iat std, backward iat min, backward iat max, backward iat mean, and backward iat std |
| Flag-based Features | the following flags are being advanced: fwd psh, bwd psh, fwd urg, fwd fin, fwd syn, bwd syn, fwd rst, and bwd rst. |
| TCP/IP Header Features | Header Length (forward and backward), Number of packets (forward and backward), Average packet size, Minimum packet length, Maximum packet length, and all of the above |
| Content-based Features | The following values are provided: forward packet length (Fwd), forward packet length (Min), forward packet length (Max), forward packet length (Mean), forward packet length (Std), and backward packet length (Bwd). |
| Miscellaneous Features | Idle Min, Idle Max, Idle Mean, and Idle Std are the variables that contribute to the overall level of inactivity. |
| Subflow Features | Subflow Fwd Packets, Subflow Bwd Packets, Subflow Fwd Bytes, Subflow Bwd Bytes |
| TCP Window Features | Init_Win_bytes_fwd, Init_Win_bytes_bwd, Fwd Act Data Pkts, Fwd Seg Size Min |
| Attack Label | Label (Benign or Attack Category) |

Table 5 **Attack Category Distribution in CIC-IDS-2017**

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

| Attack Category | Attack Type | Number of Samples |
|---|---|---|
| Benign (Normal) | Normal traffic | 2,273,097 |
| Brute Force | FTP-Patator, SSH-Patator | 193,360 |
| Denial of Service (DoS) | DoS-Slowloris, DoS-GoldenEye, DoS-Hulk, DoS-Slowhttptest | 323,515 |
| Distributed DoS (DDoS) | DDoS attack | 128,027 |
| Botnet | Bot attack traffic | 196,616 |
| Web Attacks | Web Brute Force, Web XSS, Web SQL Injection | 21,712 |
| Infiltration | Infiltrating the network | 16,429 |
| Port Scanning | PortScan | 158,930 |
| Heartbleed | Exploiting Heartbleed vulnerability | 11 |

KDD 1999

Featuring 25,192 TCP/IP connections (observations), the KDD 99 dataset is derived from a simulated LAN environment that imitates a baseline US Air Force configuration. A diversified dataset was produced by intentionally subjecting this network to a variety of attacks in an effort to make it reflect real-life events. Here, "connection" is the exchange of TCP packets between two IP addresses, one from the source and one from the destination, in accordance with a predetermined protocol. The data transmitted, the time it takes to start and end the connection, and other characteristics characterize these links. It classifies connections as "normal" or "anomalous" based on how they behave. Roughly 100 bytes of data is associated with each reported connection. With 38 numerical features and 3 qualitative attributes, every TCP/IP connection yields a total of 41 features. "Normal" and "Anomalous" are the two possible values for the class variable that determines whether a connection is considered an intrusion. A comprehensive overview of the dataset's features is provided in Tables 1 and 2, along with brief explanations of each. "Basic," "Content-based," "Time-based and "Connection-based" are the categories into which the qualities fall. The tables further detail the categorization of each feature as either "Continuous" (C) or "Discrete" (D).[29-30]

Types of features and their significance

The KDD99 dataset is a widely used benchmark for evaluating intrusion detection systems. It contains network connection records labeled as normal or belonging to specific types of attacks (e.g., denial-of-service, probe, remote-to-local, and user-to-root). Each connection is represented by 41 features, which are classified into three main categories based on their nature and

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

contribution to identifying network intrusions. Here's an overview of these feature types and their significance:

Table 1.4 Features Description KDD 99

| Feature Name | Type | Description |
|---|---|---|
| Duration | C | Length of the connection |
| Protocol-type | D | Type of protocol |
| Service | D | Network service at the destination |
| Flag | D | Normal or error status of the connection |
| Src-bytes | C | Number of data bytes from source to destination |
| Dst-bytes | C | Number of data bytes from destination to source |
| Land | D | 1 if connection is from/to the same host/port; 0 otherwise |
| Wrong fragment | C | Number of "wrong" fragments |
| Urgen | C | Number of urgent packets |

Table 6 Content-based Features:

| Feature Name | Type | Description |
|---|---|---|
| **Hot** | C | Number of “hot” indicators |
| **Num-failed-logins** | C | Number of failed login attempts |
| **Logged-in** | D | 1 if successfully logged in; 0 otherwise |
| **Num-compromised** | C | Number of compromised conditions |
| **Root-shell** | D | 1 if root-shell is obtained; 0 otherwise |
| **Su-attempted** | D | 1 if "su root" command attempted |
| **Num-root** | C | Number of “root” accesses |
| **Num-file-creations** | C | Number of file creation operations |
| **Num-shells** | C | Number of shell prompts |
| **Num-access-files** | C | Number of operations on access control files |
| **Num-outbound-cmds** | C | Number of outbound commands in an FTP session |
| **Is-host-login** | D | 1 if login belongs to the “hot” list; 0 otherwise |

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

| | | |
|---|---|---|
| **Is-guest-login** | D | 1 if the login is a "guest" login; 0 otherwise |

Table **7 Time-based Feature**

| Feature Name | Type | Description |
|---|---|---|
| **Count** | C | Number of connections to the same host as the current connection in the past 2 seconds |
| **Srv-count** | C | Number of connections to the same service as the current connection in the past 2 seconds |
| **Serror-rate** | C | Percentage of connections that have SYN errors (same-host connections) |
| **Srv-error-rate** | C | Percentage of connections that have SYN errors (same-service connections) |
| **Rerror-rate** | C | Percentage of connections that have REJ errors (same-host connections) |
| **Srv-error-rate** | C | Percentage of connections that have REJ errors (same-service connections) |
| **Same-srv-rate** | C | Percentage of connections to the same service (same-host connections) |
| **Diff-srv-rate** | C | Percentage of connections to different services (same-host connections) |
| **Srv-diff-host-rate** | C | Percentage of connections to different hosts (same-service connections) |

Table 8 host based Feature

| Feature Name | Type | Description |
|---|---|---|
| **Dst-host-count** | C | Count of destination hosts |
| **Dst-host-srv-count** | C | Service count for destination host |
| **Dst-host-same-srv-rate** | C | Same service rate for destination host |
| **Dst-host-diff-srv-rate** | C | Different service rate for destination host |
| **Dst-host-same-src-port-rate** | C | Same source port rate for destination host |
| **Dst-host-srv-diff-host-rate** | C | Different host rate for destination host with same service |
| **Dst-host-error-rate** | C | SYN error rate for destination host |
| **Dst-host-srv-error-rate** | C | SYN error rate for destination host with same service |
| **Dst-host-r error-rate** | C | REJ error rate for destination host |
| **Dst-host-srv-r error-rate** | C | REJ error rate for destination host with same service |

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

## CONCLUSION

The evolution of Intrusion Detection System technologies reflects the continuous effort to address emerging and increasingly sophisticated cyber threats. While signature-based IDS provided an effective foundation for detecting known attacks, their inability to recognize zero-day threats necessitated the development of anomaly-based and hybrid detection mechanisms. The integration of machine learning and deep learning techniques has significantly enhanced detection accuracy and adaptability, particularly in complex environments such as IoT-enabled ICS and CPS. However, these advanced IDS models often demand substantial computational resources and large volumes of labeled data, making them less suitable for resource-constrained systems. Additionally, challenges such as false alarms, scalability, lack of explainability, and privacy risks remain unresolved. Therefore, future IDS research must focus on developing efficient, scalable, and interpretable detection mechanisms that balance accuracy with resource efficiency and ensure robust security for next-generation networks.

## REFERENCES

1. D. Aldous, "The continuum random tree. I," The Annals of Probability, pp. 1–28, 1991.
2. D. Ruck, S. Rogers, M. Kabrisky, M. Oxley, and B. Suter, "The multilayer perceptron as an approximation to a Bayes optima ldiscriminant function," IEEE Transactions on Neural Networks, vol. 1, no. 4, pp. 296–298, 1990 digital forensics. Digit. Investig. 22, 3–13.
3. Anderson, James P. "Computer Security Threat Monitoring and Surveillance", 15 April 1980 http://seclab.cs.ucdavis.edu/projects/history/papers/ande80.pdf 107120.
4. N. Mhawi,AmmarAldallal,Soukeana Hassan (2022) "Advanced Feature-Selection-Based Hybrid Ensemble Learning Algorithms for Network Intrusion Detection Systems" 2022, 14(7), 1461; https://doi.org/10.3390/sym14071461, 17 July 2022
5. Cao, Y., et al., 2019. A topology-aware access control model for collaborative cyber–physical spaces: Specification and verification. Comput. Secur. 87,101478.
6. Chen, H.; Jiang, B.; Ding, S.X.; Huang, B. Data-driven fault diagnosis for traction systems in high-speed trains: A survey, challenges, and perspectives. IEEE Trans. Intell. Transp. Syst. **2020**, 23, 1700–1716.
7. ChunyingZhang,WenjieWang,LuLiu,JingRen,Liya Wang (2022) "Three-Branch Random Forest Intrusion Detection Model" 2022, 10(23), 4460; https://doi.org/10.3390/math10234460, 26 November 2022
8. Doaa N. Mhawi,AmmarAldallal,Soukeana Hassan (2022) "Advanced Feature-Selection-Based Hybrid Ensemble Learning Algorithms for Network Intrusion Detection Systems" 2022, 14(7), 1461; https://doi.org/10.3390/sym14071461, 17 July 2022

International Journal of Engineering, Science and Humanities
An international peer reviewed, refereed, open-access journal
Impact Factor 8.3 www.ijesh.com ISSN: 2250-3552

9. G. Di Crescenzo, A. Ghosh, and R. Talpade, "Towards a theory of intrusion detection," Lecture notes in computer science, vol. 3679, p. 267, 2005.
10. G. Gu, P. Fogla, D. Dagon, W. Lee, and B. Skoric, "Measuring ´ intrusion detection capability: An information-theoretic approach," in Proceedings of ACM Symposium on Information, computer and communications security (ASIACCS06), pp. 90–101, ACM New York, NY, USA, 2006.
11. G. John and P. Langley, "Estimating continuous distributions in Bayesian classifiers," in Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, pp. 338–345, 1995.
12. Hsiao-Chung Lin,PingWang,Kuo-Ming Chao,Wen-Hui Lin,Zong-Yu Yan (2021) "Ensemble Learning for Threat Classification in Network Intrusion Detection on a Security Monitoring System for Renewable Energy " 2021, 11(23), 11283; https://doi.org/10.3390/app112311283, 29 November 2021
13. Htun, H.H.; Biehl, M.; Petkov, N. Survey of feature selection and extraction techniques for stock market prediction. Financ. Innov.**2023**, 9, 26.
14. M. Shyu, S. Chen, K. Sarinnapakorn, and L. Chang, "A novel anomaly detection scheme based on principal component classifier," Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop, in conjunction with the Third IEEE International Conference on Data Mining (ICDM03), pp. 172–179, 2003.
15. Mohammad, R.M.A.; Salah, K. Detecting malicious URLs using machine learning techniques: Review and research directions. IEEE Access **2022**, 10, 121395–121417.
16. Cao, J., et al., 2021. Hybrid-triggered-based security controller design for networked control system under multiple cyber attacks. Inform. Sci. 548,
17. J. Gaffney Jr and J. Ulvila, "Evaluation of intrusion detectors: A decision theory approach," in Proceedings of IEEE Symposium on Security and Privacy, (S&P), pp. 50–61, 2001.
18. J. McHugh, "Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory," ACM Transactions on Information and System Security, vol. 3, no. 4, pp. 262–294, 2000.
19. J. Quinlan, C4.5: Programs for Machine Learning. Morgan Kaufmann,1993.
20. Jabed Al Faysal,Sk Tahmid Mostafa ,Jannatul Sultana Tamanna, Khondoker Mirazul Mumenin,Md. Mashrur Arifin,Md. Abdul Awal,Atanu Shome,Sheikh Shanawaz Mostafa (2021) "XGB-RF: A Hybrid Machine Learning Approach for IoT Intrusion Detection" 2022, 3(1), 52-69; https://doi.org/10.3390/telecom3010003, 4 January 2022
21. K. Leung and C. Leckie, "Unsupervised anomaly detection in network intrusion detection using clusters," Proceedings of the Twenty-eighth Australasian conference on Computer Science-Volume 38, pp. 333– 342, 2005.

22. L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
23. L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion detection with unlabeled data using clustering," Proceedings of ACM CSS Workshop on Data Mining Applied to Security, Philadelphia, PA, November, 2001.
24. Liu, J.; Dong, Y.; Zha, L.; Tian, E.; Xie, X. Event-based security tracking control for networked control systems against stochasticcyber-attacks. Inf. Sci. **2022**, 612, 306–321.
25. M. Mahoney and P. Chan, "An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection," LECTURE NOTES IN COMPUTER SCIENCE, pp. 220–238, 2003.
26. M. Shyu, S. Chen, K. Sarinnapakorn, and L. Chang, "A novel anomaly detection scheme based on principal component classifier," Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop, in conjunction with the Third IEEE International Conference on Data Mining (ICDM03), pp. 172–179, 2003.
27. Mohammad, R.M.A.; Salah, K. Detecting malicious URLs using machine learning techniques: Review and research directions. IEEE Access **2022**, 10, 121395–121417.
28. Satish Kumar, Sunanda, and Sakshi Arora (2020) A Statistical Analysis on KDD Cup'99 Dataset for the Network Intrusion Detection System DOI: 10.1007/978-981-15-3852-0_9
29. Ambusaidi MA, He X, Nanda P, Tan Z (2016) Building an intrusion detection system using a filter-based feature selection algorithm. IEEE Trans Comput 65(10):2986–2998
30. KDD Cup 1999. Available on: http://kdd.ics.uci.edu/databases/kddcup 99/kddcup99.html, Ocotber 2007.