International Journal of Basic and Applied Sciences, 14 (SI-1) (2025) 501-509

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS
https://doi.org/10.14419/9t3jm248
Research paper

A Comprehensive Overview of IDPS and Using The TII-SSRC-2023 Dataset for Implementing Machine Learning Techniques for Enhancing Network Security

Ashwathi Chandrasekharan [1] *, Dr. Jubilant J Kizhakkethottam [2] *

[1] MTech CS & SE, Department of Computer Science, Saintgits College of Engineering, Kottayam, India
[2] Professor, School of Computing, SRM Institute of Science and Technology, Trichy, India.
**Corresponding author E-mail: ashwathi.se2325@saintgits.org*

Received: June 18, 2025, Accepted: June 20, 2025, Published: November 1, 2025

Abstract

The growing number of connected devices and individuals, coupled with the rise in AI-driven attacks, has made robust network security essential. The dynamic nature of networks, the increasing variety of attack types, and the exponential growth of Artificial Intelligence in attacks have rendered traditional Intrusion Detection techniques obsolete. Therefore, it is necessary to create an effective Intrusion Detection and Prevention System (IDPS). Only a warning or alert message is sent by the Intrusion Detection System (IDS); it does not stop an intrusion from occurring within the network. Therefore, countermeasures can prevent, detect, and limit such attacks that could have a significant impact on the services that these networks offer. This paper begins with an explanation of the distinctions between IDS and IDPS. Additionally, it describes several intrusion detection mechanisms and their corresponding tools, such as SNORT, OSSEC-HIDS, and KISMET, continuing with the different IDPS types and how they operate. Next, for intrusion detection methods in large-scale networks, we delve deeper into some machine learning-based approaches in this research, including SVM, Decision Trees, MLP, XGBoost, and others. The TII-SSRC-23 dataset was utilized for evaluation, and the outcomes were compared to well-known machine learning-based methods for network intrusion detection. Additionally, this paper provides a detailed explanation of various recent cyberattacks documented in the TII-SSRC-23 dataset, including DDoS assaults, Brute-Force attacks, Mirai attacks, and information gathering.

Keywords: *Machine-learning-based Intrusion Detection; IDPS; TII-SSRC-23 Dataset.*

1. Introduction

Numerous applications, including e-commerce, banking, healthcare, smart homes, and educational institutions, enhance the quality of living and facilitate the widespread networking of devices and networks for communication. Cyber-attacks are a significant threat to most organizations. Some of the primary problems associated with cyberattacks include stealing confidential information, such as customer private data, trade secrets, and company plans, as well as stealing money and causing reputational damage to individuals or organizations. Businesses and governments are investing considerable effort in preventing the theft of private data. Since firewall and anti-virus software have limited capabilities for identifying and categorizing unusual behavior, they are not the only tools used to handle and control the majority of cyberthreats and attacks. Consequently, IDS and IDPS become essential in such a scenario.
Moreover, dealing with malicious software variations that cause network security breaches and significant errors poses a considerable security risk in detecting intrusions [30] [35] [36]. More than 90% of cyberattacks fall under one of these categories: malware, DDoS, Phishing, or man-in-the-middle. This document provides a concise summary of some of the most recent attacks identified in the TII-SSRC-23 dataset. Due to the development of sophisticated evasion techniques that steal vital data and evade IDS or IDPS detection, cyberattacks are becoming increasingly difficult and complex, particularly in the cases of unknown malware attacks and Denial of Service (DoS) attacks [31] [1].
As a result, this article provides a concise summary of some of the significant recent attacks and a thorough explanation of how IDPS is responding to them. IDS is used as a real-time monitoring system with the ability to spot unusual activity and issue alerts in the event of malicious attacks or unauthorized access. When an attack penetrates a network and is not blocked, intrusion detection sets off an alarm [37] [5] [19]. IDSs currently on the market have several drawbacks, including a lack of scalability and adaptability [32]. An early warning system against security breaches is the Intrusion Detection and Prevention System (IDPS), which works by monitoring network traffic patterns and identifying anomalous activity in stored data records (signature) [34] [38] [26]. As soon as the attack is identified, IDPS

Copyright © C. Ashwathi, Dr. Jubilant J Kizhakkethottam. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

blocks the offending data. This study is significant because machine learning (ML) is becoming increasingly important in the cybersecurity space [2] [8] [7] [27].
In cybersecurity, machine learning (ML) has the potential to reduce the burden of security experts, automate processes, and improve the effectiveness of security measures [4] [6] [14] [9]. Among the most effective datasets, TII-SSRC-23 includes 26 different types of recent attacks. This paper will also cover the features and other aspects of the TII-SSRC-23 dataset. Lastly, this research also discusses the outcomes of other Machine learning models applied to this dataset [33] [25].
The paper's primary contributions include:

- A review of the various intrusion detection systems (IDSs) and machine learning models employed in them is presented in this study [10] [3] [26].
- A thorough understanding of the various threats, including those that manipulate communications over protected networks, such as Mirai attacks, brute-force attacks, and Information Gathering, as well as infiltration and other types of DoS and DDOS attacks [24].
- A comprehensive analysis, categorization, and security impact assessment of the threats mentioned above.
- A comprehensive analysis of several machine learning techniques, including SVM, Decision Trees, MLP, XGBoost, and others.
- A review of prospective issues, difficulties, and research strategies for the future.
- AI-driven cyberattacks have increased at an alarming rate nowadays. Let us discuss some machine learning models in the next section. We then discuss IDPS and its types, as well as IDPS evaluation, in the subsequent sections [12].

2. Intrusion Detection and Prevention System (IDPS)

In this section, we will discuss what IDPS is and its various types. Real-time packet inspection is performed by the IDPS, which thoroughly examines each packet as it passes through the network. To prevent a similar intrusion from occurring again, intrusion correction activities aim to restore operations to a baseline condition and identify the origin and method of the intrusion. Like an intruder, Intrusion Detection Systems (IDSs) identify breaches and trigger an alert. A breach can be identified by an Intrusion Detection and Prevention System (IDPS), which can then initiate an ongoing counterattack. The IDPS will take one of the following measures if it finds any suspicious or malicious packets:

2.1. Terminate the compromised TCP connection

- Prevent the violation of the originating IP address or user profile by accessing any application, intended host, or other illegal online resource.
- Modify or reconfigure the firewall to prevent the same threat from happening again.
- If anything, malicious remains on the network following a breach, change, or remove it. And any infected attachments are removed from file or email servers.

Figure 1 compares IDS with IDPS, as we have shown. Both IDPS and IDS are similar in their ability to detect intrusions. The procedure was designed to detect and identify potential security breaches within the internal network. However, the primary method of preventing intrusions employs signature mechanisms to detect activity in network traffic and on hosts, whereby inward and outbound packets are identified [11] [17]. The idea is to block that activity before it does damage and allows access to network resources [18]. An Intrusion Detection and Prevention System (IDPS) is an inline device designed to detect and block malicious network activities in real time [16] [13] [20-23].

2.2. Types of intrusion detection techniques

IDPS Intrusion Detection and Prevention Systems (IDPS) are a combination of both Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) to offer more reliable network security. Whereas IDS only identifies and notifies administrators in case of potential threats, IDPS takes a step further to prevent or minimize attacks in real-time. The major distinction between them is based on the capabilities of the responses, in which IDS will provide warnings. In contrast, IDPS will act, which can include shutting down the compromised connection, denying access to suspicious IPs, and adjusting firewalls to avoid future attacks. An IDPS architecture consists of several key components: sensors to collect network information, a management server to analyze the information and identify potential intrusions, and a response and control server that provides countermeasures in real-time. The system is designed to operate seamlessly between the internal network and external traffic, enabling it to proactively detect and block attacks. By combining signature-based detections, anomaly detection, and real-time responses, IDPS provides a more comprehensive defense compared to conventional IDS. The modular design of the system provides a dynamic defense mechanism, as the management server analyzes the risk level and the response server implements remedial steps where needed, resulting in a powerful and flexible solution for network security.

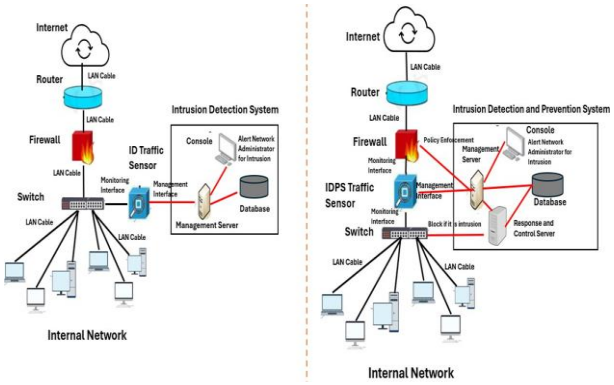


Fig. 1: Architecture of Network-Based (A) IDS vs (B) IDPS.

Machine learning-based Intrusion Detection: Machine learning (ML) enhances intrusion detection and prevention systems (IDPS) by improving threat identification precision, reducing false alarms, and providing customizable safeguards. Unlike traditional signature-based methods, machine learning algorithms identify patterns and anomalies to detect evolving cyber threats.
Policy-Based Intrusion Detection: By highlighting violations of a specified security policy, this method ensures that all users adhere to the established set of rules and standards.
Stateful Protocol Analysis-based Intrusion Detection: Anomaly detection utilizes network or host-specific profiles to identify suspicious behaviors. Going one step further, stateful protocol analysis employs each protocol state's set of standards to look for deviations. Vendor-driven protocol definitions play a major role in stateful protocol analysis. Due to its granularity, it requires a significant number of resources and valuable bandwidth to track multiple sessions simultaneously.

2.3. Some of the open-source tools for detecting intrusion

Many security-related objectives are addressed by the intrusion detection system currently on offer. Some of the tools for security are;
SNORT: SNORT is a flexible, freely available program. Snort analyzes the traffic from an IP address using an adaptable rule-based language. It achieves this by utilizing multiple types of preprocessors, protocol analysis, and content searches to record the packet in a format that humans can easily read. Snort identifies various infections, port scans, vulnerability exploit attempts, and other suspicious activities.
OSSEC-HIDS: Open-source security, or OSSEC and HIDS, is a Host-Based Intrusion Detection System, which is software that is available for free. It features a client-server architecture and will function on popular operating systems. The server could receive OS logs from OSSEC for data storage and analysis. It gets used by data centers, ISPs, and many potent log analysis engines. HIDS monitors and analyzes authentication logs and firewalls.
KISMET: WIDS (Wireless Intrusion Detection System) uses KISMET as a guide. WIDS has accessed the packet payload and WIDS events. It'll locate the access point used by burglars.

3. Types of IDPS Based on Architecture

3.1. Host-based IDPS (HBIDPS)

Host-based Intrusion Detection and Prevention solutions are deployed on individual systems, in contrast to other Intrusion Detection and Prevention Systems. Hosting facilities are essential servers that may serve as gateways to internal networks and store private information or be accessible to anyone with internet access. The HIDPS monitors all traffic entering and leaving the host machine by tracking active processes, network traffic, log files, application activity, and configuration updates. HIDPS is a feature that many anti-malware applications include in their product offerings. HIDPS can also prevent users from accessing private data on the host, thereby limiting the potential harm that Trojan horses and rootkits could cause.

3.2. Network-based IDPS (NIDPS)

Network-Based IPS: This type of network communication monitors and tracks gadgets or network data flow. After that, it examines app usage and network connection activity to identify questionable activity. With virtual private network (VPN) servers, remote access servers, routers, firewalls, and wireless networks, it is most utilized. NIDPS is designed to identify approved hosts, applications, and operating systems that are often used across a network by gathering data from a host console and the network itself once it is deployed. To identify any suspicious network changes, they also examine log data related to typical network traffic patterns.

3.3. Network behavior analysis (NBA)

The main technique used in this strategy is anomaly-based detection. While anomaly-based detection is excellent at spotting new threats, issues may arise if the network is compromised during the training period while the profile is being generated, as hostile activity may appear to be legitimate. Furthermore, a significant number of false positives are generated through anomaly-based detection due to unnoticed behavior that was overlooked during the initial training phase. NIDPS examines anomalies in network activity, whereas network behavior analysis systems (NBAs) search for unusual traffic flows to identify attacks. Such trends are usually the result of rule infringements, malware-infected assaults, or distributed denial-of-service (DDoS) attacks. NBA systems are set up within companies' networks, encompassing both internal and external networks, where they converge.

3.4. Wireless IDPS (WIDPS)

A Wireless Intrusion Detection and Prevention System (WIDPS) functions at the data link layer of the Open Systems Interconnection architecture. It monitors traffic over its wireless networks and evaluates its protocol usage to identify any questionable behaviors that may be impacting the protocols themselves. Wireless network transmission cannot detect suspicious activity from the application or higher-layer network protocols (like TCP and UDP) that it is transmitting. Through network reconfiguration and scanning for denial-of-service and other types of attacks, WIDPS may identify rogue or misconfigured devices and prevent them from functioning on wireless enterprise networks.

4. Components of IDPS

IDPS Architecture and Components, An Intrusion Detection and Prevention System (IDPS). IDPS functions in real-time and in inline mode, which means that it not only detects but also prevents security breaches. The system consists of several components. The sensors collect network traffic and monitor for potential intrusions, reporting this data to the management server for analysis and review. The management server compares the captured data to predefined signatures or anomaly profiles, detecting suspicious activity. After an intrusion is identified, the response and control server immediately responds, for example, by terminating compromised connections, blocking malicious IP addresses, or reconfiguring firewalls to prevent further intrusion. The architecture of this system also enables dynamism and adaptability in

defense, ensuring that the network remains secure as its response mechanisms are continually updated. The signature-based detection, anomaly detection, and real-time responses that IDPS can integrate enable it to take active measures in preventing attacks, providing a more comprehensive security solution than traditional IDS.

There are basically three stages in IDPS [refer fig 1]. They are the Detection stage, the evaluation stage, the response stage, and the recovery stage. The detection stage involves sensor and intrusion detection performed by the Management server. The evaluation stage includes the Management server, and the response and recovery phase encompasses countermeasures, Frontend (User interface, Command & control), Response and Control server, etc. The primary component of an IDPS for detecting intrusions on a computer or network is a sensor, as illustrated in Figure 1. To carry out detection operations, it captures a packet. The packet is sent to the Management server by it.

Depending on the requirements, the Intrusion Detection Module in the Management Server can utilize hybrid detection mechanisms or techniques based on machine learning, anomaly detection, or signature detection for intrusion detection, as discussed above. Suppose an intrusion is detected in the upcoming packet. In that case, the Management Server will pass on alerts to both the Response and Control servers, as well as the Network Administrator, through the console. The packet then proceeds to the evaluation stage, where it calculates the risk levels. Event logging that is identified is the focus of the IDPS evaluation stage, which is done by the Management server. Once the evaluation is complete, the Response and Control server initiates the active response, depending on the risk level evaluated during the evaluation stage. The Response and Control Server can notify the administrator regularly via email, block connections, reset TCP connections, record events in the database (as discussed in more detail below), and display the status in the IDPS user interface through Console. The IDPS setup, signature database enhancements, behavioral detection, and events observed by the sensor are all visible to the network administrator. A comprehensive catalog of known attack patterns, including byte sequences, protocol abnormalities, and characteristics of malicious payloads, is stored in the database. The Management server analyzes network traffic and identifies potential risks by comparing each packet to the signatures stored in the database.

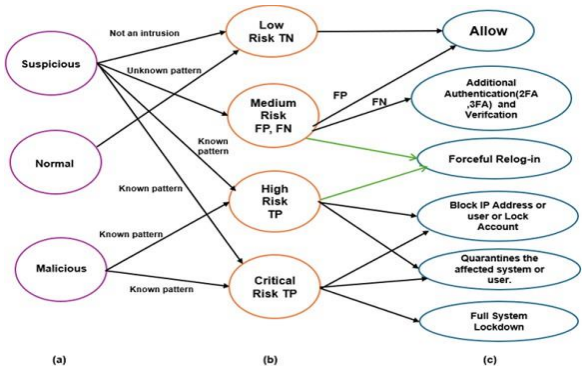


Fig. 2: Relationship between (a)Types of packets, (b) Risk Levels, and (c)Mitigation Strategy.

In Figure 2, the relationship between the type of packet, the level of risk, and the mitigation strategy is critical when considering an Intrusion Detection and Prevention System (IDPS). In this scheme, packets will be initially categorized into one of several types, e.g., normal traffic or possibly malicious packets (e.g., packets from DoS attacks, brute-force attacks, and so on). The system examines the attributes of these packets to determine the magnitude of their risks. In one instance, normal traffic is classified as low-risk, whereas suspicious or malicious packets are assigned to higher levels of risk based on their behavioral patterns and content. After the level of risk is assigned, the IDPS implements appropriate mitigation strategies. Packets with low risk may be recorded for later reference, while medium-risk packets may trigger further security inspections or temporary measures. The critical or high-risk packets, i.e., continuous data breaches or mass DDoS attacks, necessitate immediate measures to block IP addresses, cancel connections, or reconfigure firewalls. This distributed methodology enables the IDPS to prioritize responses based on the severity of the threat, providing an efficient and timely defense mechanism. When the type of packets is matched with risk levels and mitigation measures, Figure 2 indicates that the IDPS design is dynamic, allowing the system to adjust and respond to different cyber threats.

To keep pace with evolving threats, security professionals frequently update the database with new attack signatures. Information on security policies, such as what to do if a particular threat is detected, may also be kept in the database. Let's now discuss the classification and analysis of the attack by the Response and Control server. For the prevention mechanism to be activated and to distinguish between malicious and suspicious packets, the system must be proactive in risk level analysis and rating. An inline solution called IDPS aims to promptly identify and stop threats. Considering this, we propose a mapping approach for evaluating risk levels, which will be used in the subsequent section to assess the degree of threat and distinguish between suspicious and malicious incidents.

4.1. Accuracy alarm

Reducing false positive alerts is the primary goal of accuracy alarms, which measure the percentage of detection and failure based on the data kept for every assault in databases. As we know, there are four alerts which also explain the confusion matrix in the Table below:

- True negative (TN)- Volume of regular traffic
- True positive (TP) - Assault network traffic
- False negative (FN)- Traffic that is regarded as typical user traffic but contains assault.
- False positive (FP)- A user's typical activity that is interpreted as an assault.

4.2. Risk levels

In four scenarios, the risk level is determined for the following solutions overall, as shown in Figure 2:

Case 1: If an event appears suspicious but does not show any more assault shifts, it is considered to have a low-risk rate. For later, it is considered ordinary traffic. Examples include making a single unsuccessful effort to sign in, using a new gadget for the first time, etc.

Case 2: An event is classified as medium risk if it exhibits suspicion and fits an unknown pattern. Therefore, dependent on the type of attack, it is either allowed (FP) or sometimes Forceful Re-login is instructed, or additional authentication like 2FA

3FA (FN) is required. Could temporarily restrict access until verification. Notify the network administrator through alerts. Logs all activities. Ex: Unusual but not clearly malicious network traffic, Suspicious file uploads/downloads, unusual access behavior (e.g., logging in from a different country), etc.
Case 3: In the case that a known signature pattern identifies the event as suspicious or malicious. It is considered to have a high-risk rate. It is therefore blocked by the IP address/user/account. Isolate the affected system or user from others. Depending on the attack variant, the user may occasionally be prompted to log back in to verify their authentication. Ex: Multiple failed logins attempt from different locations (brute-force attack), Large-scale DDoS attack detected, Malware or ransomware behavior detected, etc.
Case 4: In the case that a known signature pattern identifies the event as suspicious or malicious. It is regarded as having a Critical-risk rate. It is therefore blocked by the IP address/user/account. Immediate system lockdown or isolation based on the type of attack. Ex: Active data exfiltration (data breach in progress), Compromised administrator account, Unauthorized privilege escalation, etc.

4.3. Activity response

Before active response stages, the network threat level is quantified using a risk rating system, as shown in Fig. 2. The active response can be divided into two categories:

1) Eactive response, which is initiated and carried out as soon as an intrusion is discovered; and
2) Proactive response, which refers to early prevention systems and aims to take preventative measures before an assault is planned. Responses typically consist of relogging in, updating the model database, and reporting to the administrator, as explained above, based on the risk ratings.

Table 1: TII-SSRC-23 Dataset Description

Dataset	TII-SSRC-23
Dataset Type	Multi class
Year	2023
Traffic Source	Emulated
Total features	86
Attacks	26
Total records	Around 8.6 million records
Normal records	Approximately 1,300 records (0.015%)
Intrusion records	Around 8,598,700 (99.985%)
Access Links	https://www.kaggle.com/datasets/daniaherzalla/tii-ssrc-23

4.4. Using the TII- SSRC-23 dataset for detecting intrusion

First, the old dataset is no longer applicable to the current network traffic due to the increasing prevalence of AI-driven cyberattacks. Furthermore, we have noticed that TII-SSRC-23 [Technology Innovation Institute - Secure Systems Research Center - 23] is more recent, has more features, and contains more instances than the previous version. One example of such a website is the TII-SSRC-23 dataset available on the Kaggle website, and the link is given below. The TII-SSRC-23 dataset was made available in 2023 by the Technology Innovation Institute's Secure Systems Research Center in Abu Dhabi, United Arab Emirates. The 27.5 GB dataset comprises eight distinct types of network flow and is categorized into two main groups: harmful and benign. Thirty-two traffic subtypes are created from these types, with 26 being malignant and six being benign. The section that follows offers more details on these subtypes.
The dataset comprises both the raw network usage data, recorded as Packet Capture (PCAP) files, and the extracted characteristics, presented as Comma-Separated Values (CSV) files. The TII-SSRC-23 dataset shares similarities with actual data, as it contains benign (normal) and common network attacks, such as Denial of Service (DoS) attacks, brute-force attacks, information-gathering techniques, and traffic, with an emphasis on the Mirai botnet. In the massive dataset TII-SSRC-23, nearly 8.6 million network flows are scattered over several files.

4.5. Types of attacks in Tii-Ssrc-23 dataset

The TII-SSRC-23 data set contains actual and harmful dissimilar attacks on networks, which are detrimental to network security. Some of these include Denial of Service (DoS) attacks, Brute-force attacks, AI-based attacks, and Information Gathering activities. They are such attacks and are elaborated on below:

Table 2: Attack Categories in TII-SSRC-23 Dataset

Category	Types of Attacks
Denial of Service (DoS)	HTTP, ICMP, MAC, UDP, TCP (ACK, CWR, ECN, FIN, PSH, RST, SYN, URG)
Brute-Force Attack	DNS, FTP, HTTP, SSH, Telnet
Information Gathering	Information Gathering
Mirai	DDoS ACK, DDoS DNS, DDoS GREETH, DDoS GREIP, DDoS HTTP, DDoS SYN, DDoS UDP, Scan & Brute-Force

4.6. Denial of service attacks (DoS)

A Denial-of-Service (DoS) attack is a type of cyberattack in which the attacker aims to temporarily or permanently prevent a host connected to the Internet from operating normally, rendering a computer or other network unavailable to authorized users. DoS attacks, to put it simply, are intended to prevent actual users from accessing a resource, such as a web page, a network, or electronic mail, or to make it extremely difficult to access. DoS attacks typically target Web servers as their target.
Denial of Service (DoS) Attacks: A DoS attack is designed to saturate a network or server, rendering it unavailable to existing users. It penetrates resources, such as web servers, routers, or communication systems, and exploits vulnerabilities to consume system resources. Citing an example, a DDoS attack, which caused major websites to go offline in 2016, including Twitter, Reddit, and CNN, was

implemented with the help of the Mirai botnet. The attack demonstrated the disastrous consequences of applying IoT devices in botnet attacks and the vulnerability of unprotected devices (Hou et al., 2024).

4.7. Brute-force attacks

A Brute-Force attack is a hacking method that uses trial and error to crack passwords, login credentials, and encryption keys. It is any effort to exploit a weakness in a digital system's user authorization. It is a simple yet reliable tactic for gaining unauthorized access to individual accounts and the systems and networks of organizations.
Brute-Force Attacks: Brute-force attacks target systems by trying every possible combination of passwords or encryption keys until the system identifies the correct one. This type of attack is common with services such as SSH, Telnet, and FTP. The Mirai botnet, used in brute-force attacks against IoT devices, exploits weak or default credentials to gain unauthorized access (Arslan et al., 2023).

4.8. Information gathering

Information gathering, also known as reconnaissance, is the initial phase in both ethical hacking and malicious cyber- attacks, in which a malicious actor actively col- lects data about a target system, network, or individual, usually as the initial step in a larger hacking operation, to gain crucial details about potential vulnerabilities and weaknesses before launching a more targeted attack; essentially, it's the process of gathering intelligence about a target before exploiting it. This process is crucial for identifying potential vulnerabilities and planning subsequent actions.
Information Gathering: Information gathering may be perceived as the reconnaissance phase of an attack, as attackers gather information to identify custom points that can be exploited to trigger more targeted attacks. Such means as port scanning and network mapping are used to gather information on potential vulnerabilities.
These are the types of attacks reported in the TII-SSRC-23 dataset, which are representative of the dynamic nature of network security threats. In particular, the Mirai botnet demonstrates that IoT devices are susceptible, and the growing use of such devices in DDoS attacks and brute force attacks is a great concern to network security experts.

4.9. Mirai attacks

The Mirai attack leverages a botnet called Mirai, which is a collection of hacked devices, including cameras, routers, and other networked devices. The main purpose of the Mirai botnet was to initiate large-scale DDoS attacks. There are numerous variations of Mirai, all referred to by different names, and they evolve as attackers modify the original code to suit their requirements or evade detection.
Mirai-based attacks: The Mirai botnet is a notorious example of a large-scale botnet attack primarily designed to target IoT devices. The Mirai botnet, which infected vulnerable IoT devices such as security cameras and routers in 2016, is one of the largest DDoS attacks of all time. The Mirai botnet had the potential to generate massive amounts of traffic, demonstrating the seriousness of the security threat posed by vulnerable IoT devices. Researchers, such as Chowdhury et al. (2024)and Al-Omari et al. (2024), highlight the drastic impacts of using hijacked computers in stage-to-stage attacks that the victim does not notice.

Table 3: TII-SSRC-23 Dataset Binary Classification Results [5]

Reference Research Paper	DL-Model	Accuracy (%)	Precision (%)	F1- Score (%)
[5]	Support Vector Machine (SVM)	99.84	100	57.87
[5]	Multi-Layer Perceptron (MLP)	99.99	100	89.48
[5]	Decision Tree	100	100	96.87
[5]	Random Forest	100	100	98.01
[5]	Extra Trees	100	100	98.60
[5]	XGBoost	100	100	98.79

Table 5: TII-SSRC-23 Dataset Multi-Class Classification Results [5]

Reference Research Paper	DL-Model	Accuracy (%)	Precision (%)	F1- Score (%)
[5]	Support Vector Machine (SVM)	97.73	72.44	61.66
[5]	Multi-Layer Perceptron (MLP)	99.94	82.62	75.60
[5]	Decision Tree	99.98	93.21	94.84
[5]	Random Forest	99.98	97.66	97.28
[5]	Extra Trees	99.98	97.46	96.71
[5]	XGBoost	99.99	98.34	97.31

4.10. Mirai-based scan & brute-force

Scanning Mechanism: Mirai-infected devices actively scan the internet for other IoT devices by sending TCP SYN probes to randomly generated IPv4 addresses, specifically targeting Telnet ports 23 and 2323. This rapid, stateless scanning process allows Mirai to efficiently locate potential targets while excluding certain IP address ranges, such as private networks and addresses allocated to specific organizations
Brute-Forcing Technique: Upon detecting an open Telnet port, Mirai attempts to establish a connection using a predefined list of over 60 common default usernames and passwords. These credentials are often factory-set faults that many users neglect to change, making their devices susceptible to unauthorized access. If Mirai successfully logs in, it infects the device, adding it to its botnet. The compromised device continues to function normally but may exhibit occasional sluggishness and increased bandwidth usage due to its participation in the botnet's activities.

5. Critical Findings and Analysis

Studies for supervised classification are conducted, considering a binary classification to distinguish between adverse and normal traffic, and a multiclass classification for identifying various traffic types as mentioned in the TII-SSRC-23 Dataset. We used Random Forest (RF),

Decision Tree (DT), Extra Trees (ET), Multilayer Perceptron (MLP), Support Vector Machine (SVM), and XGBoost as our classifiers in the supervised study [5].

The average outcome statistics from three independent runs of every model and three independent runs of each approach are shown in Table III All the models performed well in binary classification findings for differentiating between malicious and normal traffic, with XGBoost having the greatest F1 score of 98.79 (AUROC 100) and SVM having the lowest F1 score of 57.87 (accuracy 99.84). XGBoost performed similarly in multiclass classification for distinct types of traffic (F1 score: 97.31, AUROC: 99.80), while SVM had the lowest F1 score of 61.66 (accuracy: 97.73). The accuracy of MLP, DT, ET, and RF was higher than 99.94.

The results demonstrate that the selected classifiers generally performed well in our dataset's binary and multiclass classifications. However, the performance was not uniform across all models in binary tasks, with SVM and MLP classifiers yielding less satisfactory F1 scores. Conversely, the XGBoost and ET classifiers excelled in all experiments, proficiently classifying benign and malicious traffic and differentiating various traffic types and subtypes.

Here, we provide a detailed discussion of the performance variation in the machine learning models applied in the research, specifically the binary classification task, where XGBoost had an impressive F1 score of 98.79% when compared to the lower F1 score of 57.87% of SVM. It is important to understand the factors that cause this difference in performance, so that the strengths and weaknesses of these models in the context of network intrusion detection can be better understood.

5.1. Effect of dataset characteristics

One reason that leads to the difference in performance between the SVM and XGBoost can be attributed to the unbalanced TII-SSRC-23 dataset. The dataset has around 99.985 percent of intrusion records and only 0.015 percent of normal traffic records. This imbalance poses a problem to most traditional machine learning algorithms, especially those that are sensitive to the minority (i.e., normal traffic) class, such as SVM, which may not be able to find the minority class when the dataset is highly imbalanced.

The Support Vector Machine (SVM), with its high accuracy of 99.84, had a disadvantage of a lower F1 score since it is more inclined to emphasize accuracy at the expense of balanced precision and recall. With skewed databases, SVM tends to give the majority (intrusion) a greater weight and overlooks the proper categorization of the minority (benign traffic) class. Because of this, the model ends up categorizing most cases as intrusion (false positives), decreasing the recall of the minority classification and giving the model a poor F1 score.

XGBoost, which is more advanced and more capable, on the other hand, is more efficient with imbalanced datasets. XGBoost also relies on the concept of gradient boosting to create a series of decision trees, with each successive tree being trained on the misclassified cases of the last one. This is an adaptive method that enables XGBoost to make more constructive decisions by distinguishing between two classes (malicious and benign traffic) in an optimal way. Also, XGBoost has default options of dealing with imbalance in classes, like weighted loss functions and early stopping to avoid overfitting, which leads to increased precision, recall, and an eventual high F1 score of this dataset.

5.2. Simple model complexity and feature selection

The variation in the performance can also be explained by the complexity of the models. As a tree-based ensemble algorithm, XGBoost has the advantage of being able to model non-linear relationships and can model high-dimensional feature spaces with minimal preprocessing. It consists of 86 features (TII-SSRC-23) with varying levels of relevance, and XGBoost is particularly efficient in automatically choosing the most relevant features during the training process because of tree splits that help it outperform its classification rates.

SVM, on the other hand, is a model that is more complicated to tune to achieve the best performance regarding the choice of the correct kernel, regularization parameters, and feature interactions. SVM is sensitive to manual feature engineering or feature selection methods to enhance its performance, and unless it is properly tuned, it might not adequately characterize the relationships between the features and the target class, resulting in an all-purpose performance in this study.

5.3. Further background on more recent research

To further support this, Ahmed et al. (2025) in their research on the signature-based intrusion detection systems have shown that XGBoost is better than the classical classifiers, such as SVM, in deeply unbalanced and complicated intrusion detection jobs. They discovered that XGBoost could better manage noisy data, which is a natural property of real-life cyberattacks due to its capability to optimize across multiple rounds of boosting. In addition, XGBoost has more regularization mechanisms that regulate model complexity, improving its generalization ability, minimizing overfitting relative to SVM.

Conversely, SVM did not perform well in the study by Ahmed et al. because it is not as flexible as it needs to be to represent the complicated decision boundaries in those types of datasets, and therefore, it obtains lower recall with the minority population as well as lower F1 scores in an imbalanced setting.

6. Future Scope and Challenges

In addition to failing to prevent insider intrusion threats, several of the IDPS previously reviewed struggle to identify unknown patterns and manage zero-day attacks. According to recent studies, using a hybrid model for detection and prevention leads to improved performance. Future research projects must focus on creating a hybrid IDPS that is auto-immune to AI-driven cyberattacks and has early detection (based on historical and present packet traffic). The Hybrid Intrusion Detection System (IDPS) aims to avoid both insider and outside attacks by establishing a sophisticated baseline through the implementation of multiple intrusion detection techniques, such as anomaly-based and signature-based detection, using ensemble neural networks. It is necessary to do research to learn more about this IDPS mitigation strategies section and to employ robotic process automation techniques that can handle intrusions concurrently, efficiently handle false positives, optimize resource consumption, and secure all IDPS System components.

Regretfully, the current IDPS did not consider decreasing the number of false alarms and categorizing for user identification and recognition. Thus, in the future, a hybrid IDPS will be used to identify dangers by employing user-based learning behavior and a dependable, readily available system that has log file backups and up-to-date information on cyberthreats. The only topic of current research articles is network-based preventative systems. However, it was not designed to handle other types of intrusions, such as those that originate from hardware

or external sources (Keyloggers, for example). Periodic network scanning or monitoring is to prevent other sources of intrusions from penetrating the network, which is the other IDPS section that was not covered.

6.1. Perspectives and future problems

Although the implementation of machine learning driven IDPS models has achieved success based on the success that has been documented, several challenges have been realized. The focus areas of the research that the future research must investigate are as follows:
Hybrid IDPS Models: The hybrid IDPS that incorporates the use of several types of detection would be more effective. The ensemble neural network could be a combination of models, and it could consider the Convolutional Neural Networks (CNNs) to detect the features and the Recurrent Neural Networks (RNNs) to compute sequences to improve their accuracy. Transfer learning can also be applied to fine-tune the available models to intrusion detection tasks and, more precisely, where unlabeled data is not available. As the results of the studies, among them [Al-Omari and Al-Haija, 2024], suggest, an algorithm combination has the merit of raising the detection accuracy in the scenario when the data is complex.
False Positive Reduction: False positives are also of interest and need to be minimized to facilitate actual real-world use. The alternative methods to multi-layered anomaly detection (supervised (e.g., XGBoost) and unsupervised (e.g., autoencoders)) can be utilized to detect benign anomalies and actual attacks. Data fusion techniques would also boost the accuracy of detection and reduce false alarms because the information from various sources in the network would be combined to provide a larger scope.
Scalability in IoT Networks: The increased number of IoT devices has become a security issue, as well as a scalability issue for the IDPS. As the solution, it is recommended that the next generation of research should resort to the alternative of considering distributed IDPS models and edge computing as a method of data processing closer to the source, which would be less latent and expensive to process. The IDPS will also require lightweight models that can be trimmed down into massive IoT networks with pruning of neural networks.
On-the-Fly and On-the-Fly: Adaptive and real-time intrusion detection is required so that new threats can be acted on. Online learning would also improve flexibility as the models would be operating twenty-four hours, with the incoming data updated. One possible example of this is that XGBoost can be trained to enable incremental learning, i.e., it does not require retraining to be updated.
It aims at overcoming the challenges in the realm of false positive reduction, scalability, and real-time detection by designing stronger, scalable, and adjustable IDPS systems.

7. Conclusion

The main objective of IDPS is to detect and prevent network intrusions before they have a chance to do more harm to the targeted system. This research examined the distinctions between an IDS and an IDPS, as well as the different kinds of IDPS. A comprehensive analysis of multiple Machine Learning techniques and the operation of an intrusion detection mechanism based on Machine Learning has also been reported. Additionally, this paper discusses the performance and outcomes of several Machine Learning-based methods for intrusion detection and provides a detailed description of the TII-SSRC-23 23 dataset. In our upcoming work, we will evaluate a hybrid-based intrusion detection system [that includes early detection and auto immunity to AI-driven cyber-attacks] on various datasets and deep learning architectures. Additionally, we will investigate IDPS behavior- based on user research studies, appropriately classifying insider and outsider attacks, with up-to-date information about cyber threats, backups for log files, etc., and implementing our approach on countermeasures to be taken other than allowing block and relog when an intrusion is detected.

Conflict of Interest

The authors declare that none of the work reported in this publication appears to have been influenced by any known competing financial interests or personal connections.

Data Availability

Anyone can get the datasets used in this paper at https://www.kaggle.com/datasets/daniaherzalla/tii- ssrc-23/data. They are freely available (retrieved March 31, 2025).

Acknowledgment

The RD team of the Department of Computer Science at Saintgits College of Engineering in Kottayam, Kerala, India, is aiding in this research. The assistance in making this research a reality has been immeasurable.

References

[1] Abdulganiyu, O. H., Tchakoucht, T. A., & Saheed, Y. K. (2024). Towards an efficient model for network intrusion detection system (IDS): Systematic literature review. *Wireless Networks, 30*, 453–482. https://doi.org/10.1007/s11276-023-03495-2.
[2] Prayaga, L., Devulapalli, K., & Prayaga, C. (2022). *Research anthology on machine learning techniques, methods, and applications* (pp. 1023–1037). https://doi.org/10.4018/978-1-6684-6291-1.ch053.
[3] ·Sowmya, T., & Mary Anita, E. A. (2023). A comprehensive review of an AI-based intrusion detection system. *Measurement: Sensors, 28*. https://doi.org/10.1016/j.measen.2023.100827.
[4] Macfadyen, L. P., & Doff, S. (2006). *Encyclopedia of human-computer interaction* (pp. 396–403). https://doi.org/10.4018/978-1-59140-562-7.ch060.
[5] Herzalla, D., Lunardi, W. T., & Andreoni, M. (2023). TII-SSRC-23 dataset: Typological exploration of diverse traffic patterns for intrusion detection. *IEEE Access, 11*, 118577–118594. https://doi.org/10.1109/ACCESS.2023.3319213.
[6] Burrell, D. N. (2018). *International Journal of Hyperconnectivity and the Internet of Things*, 52–67. https://doi.org/10.4018/IJHIoT.2018010105.
[7] Ahmed, U., Nazir, M., Sarwar, A., et al. (2025). Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering. *Scientific Reports, 15*, Article 1726. https://doi.org/10.1038/s41598-025-85866-7.
[8] Chandrasekaran, M., & Wahida Banu, R. S. D. (2009). *Breakthrough perspectives in network and data communications security, design, and applications* (pp. 78–94).

[9] Hossain, M. S., et al. (2023). Performance evaluation of an intrusion detection system using machine learning and deep learning algorithms. *4th International Conference on Big Data Analytics and Practices (IBDAP)*, 1–6. https://doi.org/10.1109/IBDAP58581.2023.10271964.

[10] Chowdary, P. B. K., Udayakumar, R., Jadhav, C., Mohanraj, B., & Vimal, V. R. (2024). An efficient intrusion detection solution for cloud computing environments using integrated machine learning methodologies. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 15*(2), 14–26. https://doi.org/10.58346/JOWUA.2024.I2.002.

[11] Abdulganiyu, O. H., Tchakoucht, T. A., & Saheed, Y. K. (2023). A systematic literature review for network intrusion detection systems (IDS). *International Journal of Information Security, 22*, 1125–1162. https://doi.org/10.1007/s10207-023-00682-2.

[12] Al-Omari, M., & Al-Haija, Q. A. (2024). Towards robust IDSs: An integrated approach of hybrid feature selection and machine learning. *Journal of Internet Services and Information Security, 14*(2), 47–67. https://doi.org/10.58346/JISIS.2024.I2.004.

[13] Gupta, N., Jindal, V., & Bedi, P. (2023). A survey on intrusion detection and prevention systems. *SN Computer Science, 4*, Article 439. https://doi.org/10.1007/s42979-023-01926-7.

[14] Krishnan, G., Krishnan, N., Karim, S. S., Yuvarajan, G., & Priya, M. R. (2020). Cyber security in data mining to data driven security. *International Journal of Advances in Engineering and Emerging Technology, 11*(1), 71–76.

[15] Utarbayeva, M., & Mukanova, M. (2024). Integrated computer network security system: Intrusion detection and threat prediction using machine learning algorithms. *IEEE 4th International Conference on Smart Information Systems and Technologies (SIST)*, 565–570. https://doi.org/10.1109/SIST61555.2024.10629410.

[16] Ariunaa, K., & Tudevdagva, U. (2025). Generative adversarial network-based damage simulation model for reinforced concrete structures. *International Academic Journal of Innovative Research, 12*(2), 43–53. https://doi.org/10.71086/IAJIR/V12I2/IAJIR1216.

[17] Sharma, Y., Chaudhary, J., & Malhotra, V. (2023). Intrusion prevention system for website attacks. *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), 3*(7). https://ijarsct.co.in/Paper9492.pdf.

[18] Hussain, I., & Khanna, S. (2025). Development of a chaos theory-based digital image encryption algorithm for enhanced security in modern applications. *International Academic Journal of Science and Engineering, 12*(2), 1–5. https://doi.org/10.71086/IAJSE/V12I2/IAJSE1210.

[19] Diez, I. J. B., & Teleron, J. I. (2025). Enhancing cybersecurity: A comprehensive study of intrusion detection and prevention systems. *International Journal of Advanced Research in Arts, Science, Engineering & Management (IJARASEM), 12*(1).

[20] Talukder, M. A., Islam, M. M., Uddin, M. A., et al. (2024). Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. *Journal of Big Data, 11*, Article 33. https://doi.org/10.1186/s40537-024-00886-w.

[21] Balamurugan, M., Varanasi, U., Mangai, R. A., Vinayagam, P., Karuppaiah, S., & Sayyed, H. (2024). Deep learning-powered intrusion detection systems: Enhancing efficiency in network security. *International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, 1–7. https://doi.org/10.1109/ACCAI61061.2024.10602010.

[22] Telang, S., & Ranawat, R. (2024). Enhancing network security with deep learning-based intrusion detection systems. *Journal of Computer Analysis and Applications (JoCAAA), 33*(7), 1003–1013. https://eudoxuspress.com/index.php/pub/article/view/1163.

[23] Sharath, T., & Muthukumaravel, A. (2024). Deep learning-powered intrusion detection systems networks using LSTM. In S. S. Rajest et al. (Eds.), *Advancing intelligent networks through distributed optimization* (pp. 105–126). IGI Global. https://doi.org/10.4018/979-8-3693-3739-4.ch006.

[24] Al-Doori, M. B., & Komotskiy, E. I. (2024). Intrusion detection and prevention system AI based features with random forest. *IEEE Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)*, 326–328. https://doi.org/10.1109/USBEREIT61901.2024.10584056.

[25] Rakesh, L., Upadhyay, L., Reddy, P. M. (2023). Evaluation of network intrusion detection with machine learning and deep learning using ensemble methods on CICIDS-2017 dataset. *5th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 1429–1433. https://doi.org/10.1109/ICAC3N60023.2023.10541488.

[26] Issa, M. M., Aljanabi, M., & Muhialdeen, H. M. (2024). Systematic literature review on intrusion detection systems: Research trends, algorithms, methods, datasets, and limitations. *Journal of Intelligent Systems, 33*(1), Article 20230248. https://doi.org/10.1515/jisys-2023-0248.

[27] Khanan, A., Mohamed, Y. A., Mohamed, A. H. H. M., & Bashir, M. (2024). From bytes to insights: A systematic literature review on unraveling IDS datasets for enhanced cybersecurity understanding. *IEEE Access, 12*, 59289–59317. https://doi.org/10.1109/ACCESS.2024.3392338.

[28] Kauhsik, B., Nandanwar, H., & Katarya, R. (2023). IoT security: A deep learning-based approach for intrusion detection and prevention. *International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT)*, 1–7. https://doi.org/10.1109/EASCT59475.2023.10392490.

[29] Panchal, R. K., Snehkunj, R., & Panchal, V. V. (2024). A survey on network-based intrusion detection system using learning techniques. *5th International Conference on Image Processing and Capsule Networks (ICIPCN)*, 740–747. https://doi.org/10.1109/ICIPCN63822.2024.00128.

[30] Kizza, J. M. (2024). System intrusion detection and prevention. In *Guide to computer network security* (pp. 233–256). Springer. https://doi.org/10.1007/978-3-031-47549-8_13.

[31] Sharma, V., & Kumar, M. (2025). Comparative analysis of machine learning models for intrusion detection systems. *Panamerican Mathematical Journal, 35*(3s). https://doi.org/10.52783/pmj.v35.i3s.3891.

[32] Singh, R., & Ujjwal, R. L. (2024). Intrusion detection and prevention system for smart IoT network. In *Adaptive Intelligence. InCITe 2024. Lecture Notes in Electrical Engineering* (Vol. 1280). Springer. https://doi.org/10.1007/978-981-97-9045-6.

[33] Kukartsev, V., Kravtsov, K., Stefanenko, O., Podanyov, N., & Bezvorotnykh, A. (2024). Using machine learning techniques to simulate network intrusion detection. *International Conference on Intelligent Systems for Cybersecurity (ISCS)*, 1–4. https://doi.org/10.1109/ISCS61804.2024.10581097.

[34] Möller, D. P. F. (2023). Intrusion detection and prevention. In *Guide to cybersecurity in digital transformation* (Advances in Information Security, Vol. 103). Springer. https://doi.org/10.1007/978-3-031-26845-8_3.

[35] Guntoro, G., & Omar, M. N. B. (2023). A systematic literature review of intrusion detection system in network security. In *Computing and Informatics. ICOCI 2023. Communications in Computer and Information Science* (Vol. 2001). Springer. https://doi.org/10.1007/978-981-99-9589-9_8.

[36] Rysbekov, S., Aitbanov, A., Abdiakhmetova, Z., & Kartbayev, A. (2025). Advancing network security: A comparative research of machine learning techniques for intrusion detection. *International Journal of Electrical and Computer Engineering (IJECE), 15*(2), 2271–2281. https://doi.org/10.11591/ijece.v15i2.

[37] Yogesh, & Goyal, L. M. (2023). A systematic literature review of network intrusion detection system models. In *Proceedings of the International Conference on Paradigms of Communication, Computing and Data Analytics (PCCDA)*. Springer. https://doi.org/10.1007/978-981-99-4626-6_38.

[38] Rege, P. R., Kalnawat, A., Dhablia, A., Sharma, R., Kaldoke, R. S., & Ashtagi, R. (2024). Exploring machine learning's role in intrusion detection systems for network security. *International Conference on Emerging Smart Computing and Informatics (ESCI)*, 1–6. https://doi.org/10.1109/ESCI59607.2024.10497357.

[39] Lucena, K., Luedeke, H. J., & Wirth, T. (2025). The evolution of embedded systems in smart wearable devices: Design and implementation. SCCTS Journal of Embedded Systems Design and Applications, 2(1), 23–35.

[40] Megha, N., Shetty, P., Kudtarkar, R. R., Naik, S. U., & Abhilash, A. L. (2024). Design and VLSI Implementation of SAR Analog to Digital Converter Using Analog Mixed Signal. Journal of VLSI Circuits and Systems, 6(1), 55–60. https://doi.org/10.31838/jvcs/06.01.09.

[41] Madhanraj. (2025). Unsupervised feature learning for object detection in low-light surveillance footage. National Journal of Signal and Image Processing, 1(1), 34–43.

[42] Sathish Kumar, T. M. (2024). Measurement and modeling of RF propagation in forested terrains for emergency communication. National Journal of RF Circuits and Wireless Systems, 1(2), 7–15.

[43] Rahim, R. (2025). Lightweight speaker identification framework using deep embeddings for real-time voice biometrics. National Journal of Speech and Audio Processing, 1(1), 15–21.

[44] Uvarajan, K. P. (2025). Design of a hybrid renewable energy system for rural electrification using power electronics. National Journal of Electrical Electronics and Automation Technologies, 1(1), 24–32.