IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 16, 2021, accepted November 13, 2021, date of publication November 18, 2021,
date of current version December 3, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3129336*

# A Comprehensive Systematic Literature Review on Intrusion Detection Systems

**MERVE OZKAN-OKAY**[1], **REFIK SAMET**[1], **ÖMER ASLAN**[2], **AND DEEPTI GUPTA**[3]
[1]Department of Computer Engineering, Ankara University, 06560 Ankara, Turkey
[2]Department of Computer Technologies, Bandırma 17 Eylül University, 10200 Bandırma, Turkey
[3]Department of Computer Science, The University of Texas at San Antonio, San Antonio, TX 78249, USA

Corresponding author: Merve Ozkan-Okay (merveozkan@ankara.edu.tr)

This work was supported by the 2211-E National Scholarship Program for Ph.D. students.

**ABSTRACT** Effectively detecting intrusions in the computer networks still remains problematic. This is because cyber attackers are changing packet contents to disguise the intrusion detection system (IDS) recently. Besides, everyday a lot of new devices are added to the computer networks. These new devices are also raising security issues in the computer networks. To effectively manage the computer network flows and provide the security in advance; the components of the IDSs, the approaches and technologies that are used, the nature of the attacks, and the tools that are used needs to be examined deeply. This paper discusses intrusion detection technologies, methodologies, and approaches and also investigates new attack types, protection mechanisms, and recent scientific studies that have been made in this area. In addition, available datasets, well-known IDS tools, and advantages and disadvantages of particular IDSs are explained deeply. We believe that this scientific review study presents a road map for researchers and industry employees who focus on IDSs.

**INDEX TERMS** Intrusion detection system, IDS technologies, IDS methodologies, IDS approaches, datasets, IDS tools.

## I. INTRODUCTION

The Internet has become a part of daily life and an indispensable tool. It takes place in human life in many areas such as business, education and entertainment. It is used as an important component of business life [1]. In other words, with the advancement of technology, network usage emerges in every aspect of our lives. This popularity of network usage brings with it the risks of attack against the network.

Computer network security has become one of the most important issues in recent times. The most powerful mechanism for securing a network is the use of a robust security system. Firewall is one of the mechanisms used, but it is not very capable of protecting the network from attacks because the firewall can only detect attacks from outside the network. In recent years, the number of attacks on networks has increased rapidly. Therefore, interest in IDSs, which is an alternative security method, has increased among researchers [2]. IDSs are software that monitors computer networks for malicious activities, such as stealing information, censoring or breaking network protocols.

IDSs are widely used to detect both known and unknown attacks on networks from internal and external attackers. Most techniques used in today's IDS cannot cope with the dynamic and complex nature of cyber-attacks in computer networks. Due to the development of these malicious attacks every day, currently used network security systems remain insufficient to protect computer systems. For this reason, it has become necessary to develop new methods and improve current technologies in this respect. The aim of this study is to analyze IDSs, current development methods, available datasets, and remaining problems in detail. For this purpose, intrusion detection technologies, methodologies, commonly used tools and leading methods in the literature [3]–[10] are examined thoroughly.

This paper presents a detailed literature review to investigate and examine the current state of IDSs. First of all, information about what this system is, and in general, the basic features that should be in an IDSs are mentioned. Afterwards, IDSs are categorized according to the way they monitor the network traffic, record flow data, detect attacks, and report warnings. All IDS technologies, methodologies and approaches within this scope have been examined in detail. Their strengths and weaknesses are

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Huang.

mentioned, and a comprehensive summary of the work done in each area is given. Then, the datasets that are widely used in the testing and evaluation phase of the developed intrusion detection systems were examined and detailed information was given about these datasets. Finally, common intrusion detection tools used by individuals, institutions and organizations to recognize attacks are mentioned. The intrusion detection method used by each of the intrusion detection tools, their advantages and disadvantages are reviewed.

This review paper is different from the previous survey papers in many aspects. Previous studies are mainly focused on only one or two subjects such as intrusion detection methodologies or datasets that have been used. However, in this study, the various aspects of the IDSs are discussed. Besides, several suggestions are being made for each subject. The paper also makes contributions not only for researchers but also private companies which want to utilize IDSs more effectively. The contributions of this study are summarized below:

- The current status and deficiencies of intrusion detection systems and new technological developments in this context are explained.
- Intrusion detection technologies, methodologies, approaches are explained and a summary of current studies in these areas is presented.
- Commonly used datasets in intrusion detection systems are described.
- A summary of known and widely used intrusion detection tools is presented.
- Existing challenges and problems are discussed and new assumptions for intrusion detection systems are proposed.
- Provides a systematic overview of intrusion detection systems and methods for further studies.

To understand the paper language more precisely and to follow the paper structure efficiently the most used phrases are abbreviated in Table 1.

The rest of the paper is organized as follows. Section II gives basic information about IDS systems. In Section III, intrusion detection technologies and studies in this field are explained and evaluated. Intrusion detection methodologies are given in Section IV and intrusion detection approaches are explained in Section V. In addition, current studies are evaluated in Section V as well. In Section VI, commonly used datasets are examined. In Section VII, well-known current IDS tools are reviewed. In Section VIII, general evaluation is made and comparisons of IDSs are given. Finally, in Section IX, conclusion and future research directions are given.

**TABLE 1. Most used terms in the paper and their acronyms.**

| Term | Acronym |
|---|---|
| Advanced Persistent Threats | APT |
| Aegean Wi-Fi Intrusion Dataset | AWID |
| Artificial Neural Network | ANN |
| Australian Defence Force Academy | ADFA |
| Behavior based Network Intrusion Detection | BNID |
| Convolutional Neural Network | CNN |
| Correlation based Partial Decision Tree | CPDT |
| Data Mining | DM |
| Decision Stump | DS |
| Decision Tree Splitting | DTS |
| Deep Learning | DL |
| Deep Neural Network | DNN |
| Defense Advanced Research Project Agency | DARPA |
| Denial of Service | DoS |
| Discrete Hessian Eigenmap | DHE |
| Distributed Denial of Service | DDoS |
| Domain Name System | DNS |
| Feed Forward Deep Neural Networks | FFDNNs |
| File Transfer Protocol | FTP |
| Gated Repetitive Units | GRUs |
| Generalized Suffix Tree | GST |
| Hypertext Transfer Protocol | HTTP |
| Hypertext Transfer Protocol Secure | HTTPS |
| Host-based Intrusion Detection System | HIDS |
| Internet Protocol | IP |
| Intrusion Detection System | IDS |
| Intrusion Detection Prevention System | IDPS |
| Knowledge Discovery and Data Mining | KDD '99 |
| K-Nearest Neighbor | KNN |
| Long Short-Term Memory | LSTM |
| Machine Learning | ML |
| Meta Pagging | MP |
| Naive Bayes | NB |
| Naive Bayes Classifier | NBC |
| Network-based Intrusion Detection System | NIDS |
| Network Behavior Analysis | NBA |
| Neural Network | NN |
| Open Source Security | OSSEC |
| Operating System | OS |
| Open Source Wireless IPS | OpenWIPS-NG |
| Pattern based Intrusion Detection | PID |
| Random Tree | RT |
| Reduced Error Pruning tree | REPT |
| Remote to User | R2L |
| Secure Shell | SHH |
| Security Event Manager | SEM |
| Service Set Identifier | SSID |
| Signal to Noise Ratio | SNR |
| Simple Mail Transfer Protocol | SMTP |
| Simple Network Management Protocol | SNMP |
| Statistical Based Intrusion Detection | SBID |
| Support Vector Machine | SVM |
| Transmission Control Protocol | TCP |
| Tenant Virtual Machine | TVM |
| University of New Mexico | UNM |
| University of New South Wales | UNSW |
| User Datagram Protocol | UDP |
| User to Root | U2R |
| Wrapper based Feature Extraction Unit | WFEU |
| Wireless Intrusion Detection System | WIDS |

## II. INTRUSION DETECTION SYSTEMS

Intrusion detection is the operation of monitoring events occurring on a network/computer system and analyzing them for warnings of potential events, such as threats or violations of usage policies or standard security practices. IDS mainly focus on detecting potential events, recording information about these events, and reporting recorded information to security administrators. In addition, IDSs are used for other aims such as detecting issues on security policies, reporting

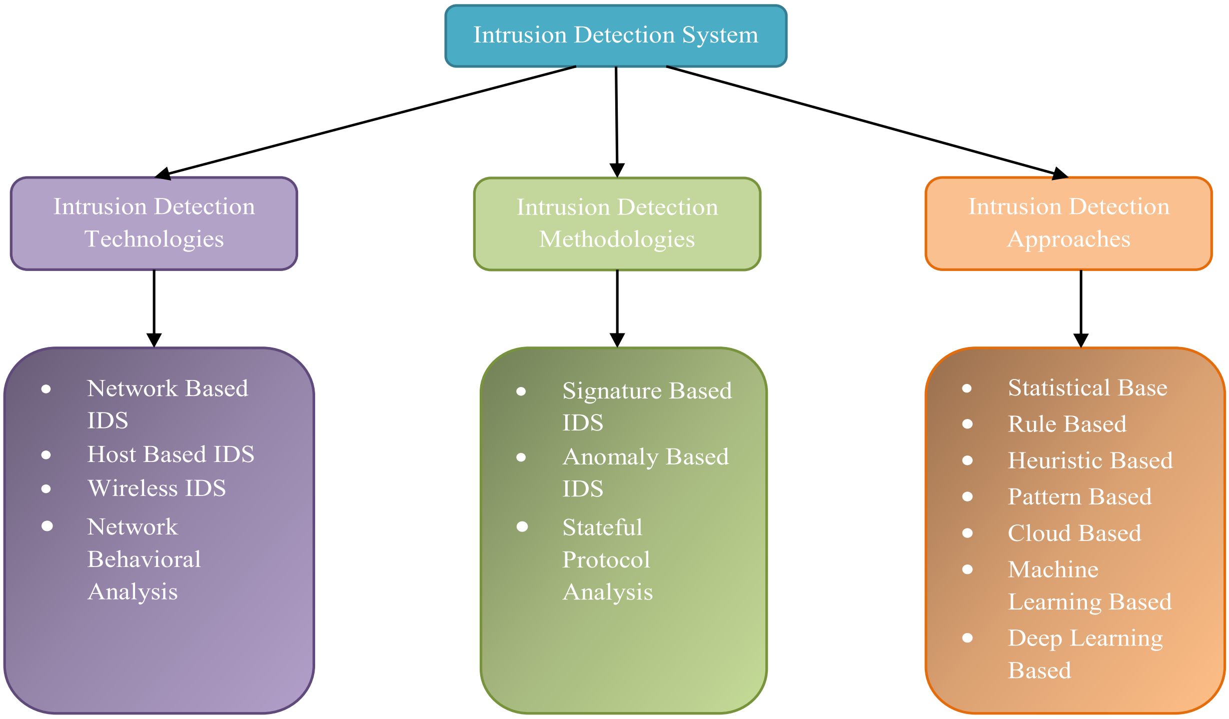

**FIGURE 1.** Classification of intrusion detection system.

present threats, and discouraging individuals from security attacks.

Generally, for an efficient and effective IDS where components must be properly secured. IDS consist of various components including users, sensors, database servers, management servers and networks. Securing IDSs components is crucial because they are targeted by attackers who want to prevent IDSs from accessing important information, known vulnerabilities or attack detection. The operating systems and applications of all components must be up-to-date, and all software-based IDS components must be protected against threats. It may also be an option to use multiple IDS technologies for comprehensive and high-accuracy detection of attacks. There are various IDS technologies being used such as network-based, wireless, and host-based. Each of them offers fundamentally different information gathering, recording, detection and prevention capabilities. Furthermore, each technology offers advantages such as detecting certain events more efficiently, or detecting with higher accuracy. For example, host-based and network-based IDSs can be integrated to provide an efficient solution. In other words, when choosing IDS technologies, different features and advantages of each technology should be considered. The most common technologies, approaches and methodologies of intrusion detection systems in the literature are given in Figure 1.

In summary, IDSs have become a necessary system for the security of almost every person, institution and organization due to the increasing dependence on technology and information systems, the spread of attacks, and their potentially damaging effects.

### A. PRINCIPLES OF IDSs

Intrusion detection is the process of observing events occurring in a computer system or network, and analyzing these events to determine intrusions. There are various threats including malware, DoS-DDoS attacks, unauthorized access, escalation of privileges or probe attack. Although many events that appear to be harmful on the system are indeed attacks, there are some exceptions; for example, the user may mistype the computer's address or unknowingly connect to the wrong system. The system must correctly separate intrusions from the normal network traffic. In conclusion, an IDS is software that simplifies and automates the process of detecting attacks.

There are some important factors for an effective attack resolution when applying IDS technologies:

- System durability/reliability;
- Fast detection;
- Minimal false positives;
- Maximum detection rate;
- Usage minimum software/hardware;
- Ability to accurately detect the location of intrusion;
- Ability to work with other technologies.

In summary, an IDS must provide the above-mentioned features for high accuracy and timely detection of attacks.

**TABLE 2. Confusion matrix.**

|  | Predicted |  |  |
|---|---|---|---|
|  |  | Positive | Negative |
| Actual Positive |  | TP | FN |
| Actual Negative |  | FP | TN |

## B. BASIC FUNCTIONS OF IDSs

First of all, there are many different IDS technologies according to the types of attacks they can recognize and the method they use to detect attacks. In addition to the ability to observe and analyze events to detect undesirable events, all types of IDS must provide the following mentioned functionalities.

### 1) RECORDING INFORMATION

Information is usually saved locally for comparison or to create profiles that are normally set. In addition, the recorded information is sent separately to central recording servers, information security solutions and management systems.

### 2) IDENTIFICATION OF IMPORTANT EVENTS

It is necessary to quickly and accurately identify a situation that occurs outside the information that is recorded regularly and that is seen as normal.

### 3) NOTIFICATION OF IDENTIFIED IMPORTANT EVENTS

These notifications, called alerts, are carried out using various methods such as e-mails, messages in the user interface of the system. A message usually contains basic information about suspicious events that have occurred. System users need to access the IDS to learn more.

### 4) GENERATING REPORTS

The generated system reports summarize observed events or provide detailed information about notable events. For example, if suspicious activity is detected in the session, IDS can collect more detailed information. Additionally, it can change settings such as when alerts should be issued after a threat is detected.

The basic common feature of IDS types is that they cannot provide a completely accurate detection. A false positive occurs when an IDS identifies a normal activity as an attack. A false negative occurs if it can't see and detect a malicious activity as normal. It is not possible to completely eliminate all these false positives and negatives. In fact, in most cases, reducing one causes the other to increase. Many IDS developers prefer to reduce the false negative rate even if the false positive rate increases.

## C. EVALUATION METRICS OF IDSs

To evaluate developed IDS models and compare their performance, metrics such as recall, false positive, false negative, precision, f-measure, and accuracy are used generally. These values are calculated using the confusion matrix (Table 2).

$$\text{Recall} = TP/(TP + FN) \tag{1}$$

$$\text{Precision} = TP/(TP + FP) \tag{2}$$

$$\text{F-Measure} = (2 * \text{precision} * \text{recall})/(\text{precision} + \text{recall}) \tag{3}$$

$$\text{Accuracy} = TP + TN/(TP + TN + FP + FN) \tag{4}$$

True Positive (TP) is a correct prediction of the positive class (prediction and actual both are positive). True Negative (TN) is a correct prediction of the negative class (prediction and actual both are negative). False Positive (FP) is the wrong prediction of the negative class (predicted-positive, actual-negative). False Negative (FN) is a wrong prediction of the positive class (predicted-negative, actual-positive). Precision (also called positive predictive value) is the ratio of relevant samples among the taken samples; recall (also known as sensitivity) is the ratio of relevant samples taken. f-measure is the harmonic mean of precision and recall Accuracy is the measure that gives how much of the data was classified correctly.

## D. CHALLENGE OF IDSs

Intrusion detection systems can be defined as security systems that monitor computer systems and network traffic and use this information to identify external attacks, system abuses, or internal attacks [11], [12]. Today, IDSs are seen as one of the basic security products that should be used in corporate systems. IDSs can be used as a layered security architecture when used with other security products. For example, many use IDSs alongside firewalls and anti-virus software. In this way, IDSs can be used to detect attacks that other security products cannot detect.

IDSs detect attacks with different methods and techniques. Anomaly detection studies from system calls have been going on for many years. However, although a lot of work has been done in this area to produce universal datasets, there are still deficiencies in datasets that should theoretically model all normal behaviors. At the same time, anomaly-based approaches can detect unknown attacks as well as known attacks to a certain extent, while they can also identify normal behaviors as attacks. End users or system administrators should examine the behavior detected by IDS as an attack. Thus, it is possible to extract the correct signature for the application, which was detected through anomaly detection systems and determined to be an attack after analysis. Signature-based systems, on the other hand, can directly detect attacks with their signature, but they cannot detect unknown attacks. Machine Learning techniques have recently received wide attention in the field of intrusion detection. There are many classification techniques that have proven to be effective in solving a wide variety of problems such as pattern recognition, image processing and cyber security, especially in the field of intrusion detection. However, ML techniques are more useful for estimating between two possible outcomes, such as normal or abnormal, for a given network traffic. The Software Defined Networking (SDN) architecture is based on a centralized control, separating the data plane from the control or management plane, thus providing the ability to program the network. All network devices can be monitored and managed from a central location. Centralized control of SDN can be leveraged to save and improve storage and processing as well. However, there are not any standardized security protocols for SDN. Even though there are some third party service providers, still there exists a security concern. In summary, existing IDSs cannot cope with the dynamic nature of the currently developing attack types [13]–[19].

In the studies to be carried out on these research areas, the development of new methods that will contribute to the literature, the generation of new datasets and the application of new technologies should be included. Another issue is that hybrid IDSs should be created to combine the strengths of IDS types to cover each other's weaknesses, and these systems should be used in real environments. In this study, detailed IDS examination and analysis were made for IDS types, strengths and deficiencies in order to contribute to new technologies that can be developed.

## III. INTRUSION DETECTION TECHNOLOGIES

### A. NETWORK-BASED IDSs

A network-based IDS (NIDS) monitors network traffic for the security of the network devices and analyzes the protocols (network, application, transport, etc.) that have been used to detect suspicious activities [20], [21]. TCP/IP is widely used to provide network communication. TCP/IP consists of four layers which work together. When a user wants to transfer data, the data is passed from the highest layer to the lowest layer and more information is added in each layer. The lowest layer transmits the collected data over the physical network; then the data is transmitted from the layers to the destination.

The four TCP/IP layers work together to transfer data between hosts. In network-based IDSs, most of the analysis usually takes place at the application layer. Some network-based IDSs also perform limited analysis at the hardware layer.

In general, network-based IDSs consist of sensors, one or more management servers, database servers, and multiple consoles. All of the components mentioned, except the sensors, are similar in other IDS technologies as well. The network-based IDS sensors, monitor and analyze the network activities.

#### 1) SECURITY FEATURES OF NIDS

Network-based IDSs offer a wide range of security capabilities. Common security features, which are broadly divided into three categories, are described in detail at the below: information collection, logging, and detection.

##### a: INFORMATION COLLECTION

Network-based IDSs have limited capability to gather information from the communication networks. The collected information is generally collected about related hosts and network activities. Some of the collected information features can be listed as follows:

- Identifying Hosts: An IDS can create a list of network hosts.
- Identification of Operating System: Operating systems and versions used by hosts can be identified. Knowing the operating system version used is helpful in identifying vulnerable hosts.
- Identification of Applications: An IDS sensor is able to identify application versions by monitoring ports in use and monitoring application communication. This information is used to identify potentially vulnerable applications and their unauthorized use.
- Determining Network Characterization: General information about some IDS sensors, network configuration and traffic is collected. Thanks to this information, any changes in the network configuration are easily detected.

##### b: LOGGING

Network-based IDSs logs comprehensive data on detected events. This data is used to validate alerts, investigate and correlate events. Data types commonly logged by network-based IDSs are as follows:

- Date and time;
- Number of connections;
- Event type;
- Protocols;
- Source and destination IP addresses;
- Number of transmitted packets;
- Application requests and responses.

##### c: DETECTION

Network-based IDSs offer broad detection capability. Many network-based IDS integrate signature-based method and anomaly-based method to perform detailed analysis and increase the detection rate. When the anomaly-based method examines anomalous activities, it parses it into requests and responses that are examined and compared with the signatures of known attacks. That is, the implementation of the methods is nested.

#### 2) RELATED WORK

Network-based IDSs offer extensive detection capability. Most studies use a combination of different attack detection techniques to obtain a high accuracy rate in attack detection in addition to NIDS. That is, intrusion detection methods often overlap with each other. Some of the studies conducted in this area are summarized at the below in Table 3.

Wattanapongsakorn *et al.* [22] proposed a network-based Intrusion Detection and Prevention System (IDPS). The purpose of this system is to effectively detect known attack types and to take immediate action against attacks. The proposed approach can be used with different machine learning techniques and tested on an online network environment. The results show that the proposed IDPS can recognize normal

**TABLE 3.** Summary of network-based intrusion detection methods.

| Paper | Proposed Method | Goal/Success | Year |
|---|---|---|---|
| Wattanapongsakorn *et al.* [22] | A network-based Intrusion Detection and Prevention System. | It can detect normal activities from attack types with high accuracy and automatically block the victim's computer network against attacks.<br>C4.5 Decision Tree algorithm with proposed approach detects unknown attack types. | 2012 |
| Amaral *et al.* [23] | A network-based intrusion detection system for IPv6-enabled wireless sensor networks. | With the proposed system, possible misbehaviors can be detected instead of detecting predefined attacks. | 2014 |
| Kumar *et al.* [24] | A Machine Learning based model for Network Based Intrusion Detection Systems. | The proposed model was able to detect known and unknown attacks with up to 99.4% accuracy. | 2016 |
| Qassim *et al.* [25] | Anomaly and network based intrusion detection system. | The proposed system based on machine learning algorithms is effective and efficient in terms of classifying malicious activities. | 2016 |
| Karatas and Sahingoz [26] | Comparing 7 different network training functions in a multilayered artificial neural network. | "trainlm" function is the best algorithm on the IDS application area which was implemented as a pattern recognition problem.<br>A time-consuming function of "trainr" also produces a very good result in a less number of epochs. | 2018 |
| Larijani *et al.* [27] | A novel Random Neural Network based Intrusion Detection System(RNN-IDS). | In the NSL-KDD dataset, performance evaluation was made by training different numbers of input and hidden layer neurons with learning rates.<br>The results were compared with existing systems and an accuracy of 94.50% was obtained. | 2018 |
| Mazini *et al.* [28] | A new hybrid method for an anomaly network-based IDS. | The proposed method applied on NSL-KDD and ISCXIDS2012 datasets and %98.9 accuracy rate is obtained. | 2019 |
| Meftah *et al.* [29] | An anomaly-based network intrusion detection approach. | Obtained 82.11% accuracy rate with Support Vector Machine.<br>Applying the two-stage hybrid classification improved the accuracy of results by up to 86.04%. | 2019 |
| Devan and Khare [30] | A XGBoost-DNN model which uses the XGBoost technique for feature selection and uses deep neural network (DNN) to classify network intrusion is proposed. | NSL-KDD dataset was used. The results were compared with existing logistic regression, naive bayes, and support vector machine.<br>According to observed results, DNN reveals a level of 97% classification accuracy than existing models. | 2020 |
| Bedi *et al.* [31] | A two-layer Improved Siam-IDS approach for the problem of imbalance class. | Compared to similar studies, I-SiamIDS showed important improvement in recall, accuracy, F1 score, precision and AUC values for both CIDDS-001 and NSL-KDD datasets. | 2021 |

events from attacks within seconds with high accuracy and automatically block the victim's computer network against attacks. Additionally, they applied the C4.5 Decision Tree algorithm with a proposed approach to detect unknown attack types and this algorithm can work effectively when faced with unknown types of network attacks. However, this study can be further improved by developing the approach for the detection of unknown attacks as well as the detection of known attacks.

Amaral *et al.* [23] proposed a network based intrusion detection system for IPv6-enabled wireless sensor networks. The proposed system detects attacks by using traffic signatures and abnormal behaviors. Proposed system consists of two components PPPSniffer and Finger2IPv6. In the proposed system, network nodes selected as observers are located by the intrusion detection system. In this way, packets exchanged in neighbors are observed and possible attack attempts are detected. The observed messages are compared with the rule set created by NIDS. If a match occurs, an alarm is generated and sent to the Event Management System. With this proposed system, possible misbehaviors can be detected instead of detecting predefined attacks. However, the system should be improved by adding new detection rules.

Kumar *et al.* [24] proposed and evaluated Network Based Intrusion Detection Systems based on machine learning to detect threats to the network. In this study, different supervised machine learning classifiers are constructed using datasets including labeled examples of network traffic features created by various benign and malicious applications. The main goal of this study is Android-based malware due to the increase in mobile malware and its popularity among users. For testing the proposed approach, traffic was generated. Several malware examples such as Premium SMS sender, backdoor, spammer, bots, ransomware, information stealing and fake antivirus were used to generate this traffic. According to the obtained results, the proposed approach was able to detect unknown and known attacks up to 99.4% accuracy. This study can be improved by enlarging the created dataset and integrating it into the existing intrusion detection systems mentioned.

According to Qassim *et al.* [25], anomaly-based intrusion detection system (AIDS) can identify the network traffic that is detected as malicious. It raises an alarm each time when it detects an activity that is different from the normal behaviors. Therefore, managing IDS alarms and distinguishing false positives from true alarms becomes a major challenge.

This study proposed an approach consisting of two steps. Firstly, they suggested a set of network traffic features that are supposed to be the most relevant features in detecting anomalies in the network. Secondly, an AIDS alarm classifier proposed to classify activities automatically by a packet header-based anomaly detection system. According to the authors, the proposed system based on machine learning algorithms is effective and efficient in terms of classifying malicious activities. This study can be improved using various machine learning techniques to increase the accuracy rate.

Mazini *et al.* [28] proposes a new hybrid network-based IDS approach to detect anomaly by using AdaBoost and artificial bee colony (ABC) algorithms. Feature selection was made using the ABC algorithm. The AdaBoost algorithm was used to evaluate and classify the selected features. The proposed approach was applied to NSL-KDD and ISCXIDS2012 datasets to evaluate the accuracy of the method. 98.9% accuracy rate is achieved. According to the authors, the proposed method outperformed other IDSs on the same dataset. In the future studies, accuracy can be further improved and performance evaluation can be made on different datasets.

Meftah *et al.* [29] implemented an anomaly-based network intrusion detection approach using the UNSW-NB15 dataset. Their approach consists of two main stages. They use Recursive Feature Elimination and Random Forests among other techniques to select important features for machine learning purposes. Then they perform a binary classification to detect abnormal traffic using different data mining techniques such as Support Vector Machine, Gradient Boost Machine and Logistic Regression. They achieved the highest accuracy result of 82.11% with the Support Vector Machine. They then feed the output of the SVM into a set of polynomial classifiers to increase the accuracy of detecting attack types. In particular, they evaluated the performance of Naive Bayes, Decision Trees and polynomial SVM. The application of the two-stage hybrid classification increased the accuracy of the results up to 86.04%. This work can be further developed on different datasets by developing a new classification algorithm or using deep learning techniques.

NIDSs trained on unstable data incline to offer inaccurate forecasts against small classes of attacks, resulting in undetected or misclassified intrusions. Previous studies have addressed this class imbalance problem using data-level approaches that increase minority-class instances or reduce majority class instances. Although these balancing approaches indirectly improve the performance of NIDSs, they fail to address the underlying problem. In the study of Bedi *et al.* [31], a two-layer Improved Siam-IDS (I-SiamIDS) approach was proposed to address the problem of class imbalance. I-SiamIDS defines both minority and majority classes as algorithms without using any data level balancing technique. The first layer of I-SiamIDS uses a binary ensemble of Siamese Neural Network, eXtreme Gradient Boosting and Deep Neural Network (DNN) for filtering of input samples. After that, these attacks are sent to the second layer to be classified into different attack classes using the multi-class eXtreme Gradient Boosting classifier (m-XGBoost). Compared to similar studies, I-SiamIDS showed important improvement in recall, accuracy, F1 score, precision and AUC values for both CIDDS-001 and NSL-KDD datasets. In order to present the results more clearly, the computational cost analysis of the proposed method is also given. At the same time, this study can be improved by examining the results on different datasets.

### 3) EVALUATION OF NETWORK-BASED IDSS

Network-based IDSs are generally known to have a high rate of false negatives and positives. Most of the early developed network-based IDSs used signature-based detection to detect known simple attacks. Novel technologies have used a combination of detection methods to achieve high accuracy and increase the type of attacks that can be detected. Thus, false positive and negative rates are reduced. Another problem is that they often require a significant amount of tuning and customization to take into account the characteristics of the observed environment.

Although network-based IDSs have extensive detection, they have several important restrictions. The most important of these are analyzing encrypted traffic, handling heavy traffic loads, and countering attacks against the IDSs. NIDSs cannot detect attacks on encrypted network traffic and cannot perform full analysis in case of high load. In addition, IDS sensors may cause several events to not be detected, particularly if stateful protocol analysis is used.

## B. HOST-BASED IDSs

Host-based IDSs (HIDS) observe a host's properties and activities to detect potential threats. A host-based IDS monitors data such as traffic information, logs of system, file access and modification [32], [33].

Most HIDS have detection software known as agents installed on interest hosts. Each agent monitors activity in a single host. Agents forward data to management servers that can use database servers. Consoles are used for management and monitoring. Some host-based IDSs use special devices that run the agent software directly instead of installing it on individual hosts. Each device is positioned to monitor traffic on a particular host. Technically, these devices can be considered network-based IDSs. Each device is specifically designed to protect one of the following:

- Server: In addition to observing the server's operating system, the agent can monitor some applications.
- Client Host: Agents designed to monitor users' hosts often observe the operating system, common applications such as email clients and web browsers.
- Application Service: Some agents are designed only to observe a specific application, such as a web server program or database server program. Such agents are also known as application-based IDSs.

**TABLE 4.** Summary of host-based intrusion detection methods.

| Paper | Proposed Method | Goal/Success | Year |
|---|---|---|---|
| Ou *et al.* [34] | Designed and applied a host-based intrusion detection system with log file analysis and Back Propagation neural network technology. | The proposed system improved the efficiency and accuracy rate of attack detection. | 2010 |
| Creech *et al.* [35] | A new host-based anomaly intrusion detection approach based on a semantic algorithm. | The new semantic based algorithm showed significantly better performance than the current methods on three different dataset. | 2013 |
| Catherine *et al.* [36] | A new Host-based IDS named as CPDT (Correlation based Partial Decision Tree Algorithm). | The algorithm was applied on KDD '99 dataset and 99.9458% accuracy rate was obtained.<br>CPDT gives better results than the existing algorithms. | 2014 |
| Subba et al. [37] | A novel HIDS framework. | It effectively detects intrusions with low false positive rate and high accuracy. | 2017 |
| Chawla *et al.* [38] | A new Host based IDS. | The stacked CNN/GRU is roughly 10 times faster than LSTM due to faster convergence in training.<br>The 100% True Detection Rate and the 60% false alarm rate are obtained with the proposed system. | 2018 |
| Deshpande *et al.*[33] | A host based intrusion detection model for Cloud computing environment with implementation and analysis. | The model provides security as a service (SecaaS) in the infrastructure layer.<br>Implementation result shows 96 % average intrusion detection sensitivity. | 2018 |
| Byrnes *et al.* [39] | Examining the latest Linux kernel 5.7.0-rc1. | It presented certain runtime and memory constraints that must be met in order for the models to run at their intended limits. | 2020 |
| Gassais *et al.*[40] | A novel host-based automated framework for intrusion detection in IoT which combines user space and kernel space information and machine learning techniques. | According to the authors, the presented solution obtained good results and was optimized for quick detection. | 2020 |
| Liu *et al.* [41] | Host-based intrusion detection system with system calls named as SCADS which use Apache Spark in the Google cloud environment. | According to the authors, the experiment results demonstrate that the efficiency of intrusion detection is enhanced.<br>The proposed method can apply to the design of next-generation host-based intrusion detection systems with system calls. | 2021 |
| Park *et al.* [42] | A novel approach proposed and applied on the Leipzig Intrusion Detection Dataset (LID-DS). | According to the accuracy, precision, recall and F1-score indicators, the proposed Siamese-CNN model showed an increase of approximately 6% compared to the vanilla-CNN model. | 2021 |

### 1) SECURITY FEATURES OF HIDS

Host-based IDSs offer different security capabilities. These are logging, detection and other features.

#### *a: LOGING*

Host-based IDSs often log extensive data on detected events. This data can be used to validate alerts, investigate events, and correlate events among other sources. Generally, the data fields recorded by host-based IDSs are:

- Date and time;
- Type of event or alert;
- IP address;
- Port information;
- Application information;
- Filenames/paths and user IDs.

#### *b: DETECTION*

Most host-based IDSs have the ability to detect several types of malicious activity. They often use a combination of signature-based detection techniques to identify known attacks, and a combination of policy or rule sets and anomaly-based detection techniques to identify previously unknown attacks.

### 2) RELATED WORK

The summary of host-based IDSs methods can be seen in Table 4. In Table 4, the main idea of each study and important aspects of the papers are discussed.

Ou *et al.* [34] designed and applied a host-based intrusion detection system, which consists of two detection technologies. These are log file analysis and Back Propagation neural network technology. Log file analysis was used for misuse detection, and BP neural network was used for anomaly detection. The aim of combination of these two detection technologies is the proposed HIDS can effectively enhance the accuracy and efficiency of intrusion detection. Obtained results show that the proposed system improved the efficiency and accuracy rate of attack detection.

Creeach and Hu [35] proposed a new host-based anomaly intrusion detection approach based on a semantic algorithm. The aim of this study is increasing detection rates whilst reducing false alarm rates by using discontinuous system call patterns. The main concept is to apply a semantic structure to kernel-level system calls to help detect abnormal behaviors. This new approach was evaluated on three different datasets. KDD98 dataset and the new ADFA Linux dataset (ADFA-LD) were used for testing core performance, the UNM dataset was used for portability and robustness testing. According to the authors, a new semantic based algorithm showed significantly better performance than the current methods. This research may investigate the novel techniques to reduce the training overhead and enhance the resilience of semantic features.

According to Catherine *et al.* [36], existing host based intrusion detection systems are not fast enough for attack detection because of using the whole feature set. As a solution of the mentioned problem, this paper proposed a new Host-based IDS named as CPDT (Correlation based Partial Decision Tree Algorithm). The CPDT integrates Correlation feature selection for selecting features and Partial Decision Tree for classifying traffic. The algorithm was applied and evaluated on KDD '99 dataset and 99.9458% accuracy rate was obtained. According to the authors CPDT gives better results than the existing algorithms. This work can be developed with a new method to detect the unknown attacks.

Subba *et al.* [37] proposed a novel HIDS framework to reduce computation and be resource intensive. The proposed framework firstly turns the system call traces into n-gram vectors and then reduces the size of the input feature vectors by applying a dimensionality reduction. The reduced feature vectors were analyzed with several classifier models based on machine learning. To evaluate performance of the proposed model, an ADFA-LD dataset was used. The obtained results showed that it effectively detects intrusions with low false positive rate and high accuracy. This study can be enhanced with fine tune various parameters to further increase its performance.

Chawla *et al.* [38] proposed a new Host based IDS to identify normal behavior of a system based on sequences of system calls. This study describes an efficient anomaly based intrusion detection system in terms of computation based on Recurrent Neural Networks. By using Gated Repetitive Units instead of normal LSTM networks, it is possible to achieve important results with decreased training times. Combining GRUs with CNNs enhances anomaly IDS. The proposed technique applied on the ADFA dataset. Obtained results showed that stacked CNN/GRU is roughly 10 times faster than LSTM due to faster convergence in training. Additionally, they achieved the 100% True Detection Rate and the 60% false alarm rate with the proposed system. This study can be improved by increasing the number of training samples or applied on different dataset.

According to Byrnes *et al.* [39], sometimes decades-old datasets remain obsolete for a long time, due to the rapid evolution of operating systems and the resulting underlying complexity. In this study, they aimed to close the gap between theoretical models and application environments by examining the latest Linux kernel 5.7.0-rc1. This environment examines the feasibility of sys call-based HIDS in modern operating systems and the limitations imposed on the HIDS developer. Recent advances to the kernel are examined, and a new approach is proposed to generate data and improve the detection model. It also presented certain runtime and memory constraints that must be met in order for the models to run at their intended limits.

Park *et al.* [42] carried out an experiment on the Leipzig Intrusion Detection Dataset (LID-DS), which is a host-based IDS dataset created in 2018. In addition, an intrusion detection model consisting of a host-based preprocessing, vector-to-image processing, training and testing steps is proposed to improve the performance of the system. In the training and testing steps, a Siamese Convolutional Neural Network (Siamese-CNN) was constructed using a learning technique consisting of several steps with high performance using a small amount of data. Siamese-CNN determines the type of attack based on the similarity score of each attack sample converted to image. Accuracy is calculated using a few shot learning techniques. For performance evaluation, the performance of Vanilla Convolutional Neural Network (Vanilla-CNN) and Siamese-CNN was compared. According to the accuracy, precision, recall and F1-score indicators, the proposed Siamese-CNN model showed an increase of approximately 6% compared to the vanilla-CNN model. The proposed model can be developed by working on increasing the accuracy of intrusion detection for known or unknown cyber-attacks by optimizing the hyper parameter values.

#### 3) EVALUATION OF HOST-BASED IDSS

The types of attacks detected by host-based IDSs vary depending on the detection techniques used in the system. Some host-based IDS products combine these detection techniques, while others focus on several or one. Because host-based IDSs have detailed knowledge of hosts' features and configurations, an IDS agent can often determine whether an attack against a host will succeed if it is not stopped.

As with other IDSs technologies, host-based IDSs often cause false positives and negatives. However, accuracy of detection may be more difficult for host-based IDSs. Because most IDSs do not know the context in which detected events such as log analysis and file system monitoring occur. For instance, a host might reboot or a new application might be installed, and these actions could be malicious activities. The events themselves are accurately detected, but often, without additional information, it cannot be determined whether they are normal or attack. Host-based IDSs using a combination of several detection techniques can generally provide a more accurate detection rate than those using one technique. Since each technique can monitor different aspects of the host, using more techniques allows to gather more information about the activities taking place. This provides a more complete profile of events and may also provide additional information to assist in assessing the intent of events.

### C. WIRELESS IDSs

Wireless IDSs (WIDS) monitor wireless network traffic and analyze wireless network protocols to identify suspicious activity [43], [44]. It cannot identify suspicious activities in the application or in the upper layer network protocols (for example, TCP, UDP) that wireless network traffic is passing through. It is widely deployed in the coverage area of the wireless network for monitoring.

#### 1) SECURITY FEATURES OF WIDS

Wireless IDSs offer a variety of security features. Since Wireless IDS is a relatively new type of intrusion detection system, its features currently vary widely between products. Common security capabilities are defined, which are basically divided into three categories: information collection, logging and detection.

##### *a: INFORMATION COLLECTION*

Most wireless IDSs can collect information about wireless devices. Examples of these information gathering capabilities are as follows:

- Identifying Devices: Most IDS sensors create a chart of their observed devices, including APs, clients. The charts are used as a profile to define new devices and removal of existing devices.
- Identifying Wireless Networks: Most IDS sensors monitor monitored networks by identifying them by their SSID. Each is then labeled as an authorized, normal or rogue network. This information is used to identify new networks and also to prioritize responses to identified events.

##### *b: LOGGING*

Wireless IDSs usually record extensive data on detected events. This data can be used to validate alerts, investigate events, and correlate events between IDS and other logging sources. Data types commonly recorded by wireless IDSs are as follows:

- Date and time
- Activity or alert type
- Priority or importance
- Source MAC address
- Identity of the sensor observing the event

##### *c: DETECTION*

Wireless IDSs can detect WLAN protocol level attacks, misconfigurations, and policy violations by first examining IEEE 802.11a, b, g, and i protocol communication. Wireless IDSs do not examine communications at higher levels (e.g. IP addresses, application payloads). Some products only perform simple signature-based detection, while others use a combination of signature-based detection, anomaly-based detection, and situational protocol analysis techniques.

#### 2) RELATED WORK

The summary of wireless IDSs methods can be seen in Table 5. In Table 5, the main idea of each study and important aspects of the papers are discussed.

Meng and Li [45] develop a trust-based intrusion detection mechanism by using Bayesian model and evaluate it in terms of detecting malicious activities. A map of trust values among different sensor nodes is created thanks to this Bayesian model. To evaluate the performance of the proposed mechanism, they analyzed the impact of a fixed and a dynamic trust threshold and also evaluated in a wireless sensor environment. The experimental results showed that the Bayesian model is promising in detecting malicious activities. This study can be enhanced with using several models.

According to Afzal *et al.* [46], there is no specific and practically implemented open source WIDS solution for detection of deauthentication and the evil twin attack. In this study, they proposed an open source WIDS for detecting these OSI layer attacks. The detailed examination of these two common attacks on the standard is conducted. After that, these attacks are implemented to learn attack behavior. Finally, novel attack signatures and techniques to detect these attacks are designed and implemented as a Wireless Intrusion Detection System (WIDS). In the evaluation of the proposed system, 89% and 93% accuracy rates are obtained for the two attacks. According to authors, proposed attack signatures worked but there is can be improved further.

Kolias *et al.* [47] proposed a system called TermID, a distributed intrusion detection system that is suitable for wireless networks. The system is based on swarm intelligence principles and classification rule induction to obtain an effective training model for intrusion detection without exchanging data. Another feature is that the proposed model is easily readable. These are the main principles of the proposed approach. This approach was tested on AWID dataset. Among the requirements of a modern WIDS, this study achieved the task of building a model in terms of low network footprint and user privacy. These studies can be improved by examining accuracy rate and detecting unknown attack types.

According to Abdulhammed *et al.* [48] feature selection is a key factor for an improved wireless intrusion detection system based on machine learning classifiers. There are two main contributions of this study:

1. selecting effective features,
2. improving a machine learning based wireless intrusion detection system.

This study discusses multiclass classification using four effective feature sets of 5, 7, 10, and 32 features, respectively. The obtained results used the AWID dataset to evaluate the efficiency of seven well-known machine learning classifiers based on the selected set of features. The proposed system used a Random Forest algorithm with 32 features and achieved a maximum accuracy of 99.64%. The obtained results showed that effective feature reduction gives better results in terms of accuracy rate and speed. This study can be improved by examining proposed approach on different datasets.

Kasongo and Sun [50] proposed wireless IDS based on deep learning. The proposed technique uses feed forward deep neural networks (FFDNNs) combined with a filter based feature selection algorithm. The FFDNN IDS is interpreted

**TABLE 5. Summary of wireless intrusion detection methods.**

| Paper | Proposed Method | Goal/Success | Year |
|---|---|---|---|
| Meng and Li [45] | A trust-based intrusion detection mechanism by using Bayesian model. | Analyzing the impact of a fixed and a dynamic trust threshold.<br>The experimental results showed that the Bayesian model is promising in detecting malicious activities. | 2013 |
| Afzal *et al.* [46] | A WIDS solution for detection of deauthentication and the evil twin attack. | In the evaluation of the proposed system, 89% and 93% accuracy rates are obtained for the two attacks. | 2016 |
| Kolias *et al.* [47] | A WIDS system called TermID. | This study achieved the task of building a model in terms of low network footprint and user privacy. | 2017 |
| Abdulhammed *et al.* [48] | Proposed feature selection and multiclass classification. | Random Forest algorithm and 32 features achieves a maximum accuracy of 99.64%.<br>The obtained results showed that effective feature reduction gives better results in terms of accuracy rate and speed. | 2018 |
| Vijayakumar and Ganapathy[49] | The paper made an extensive survey about the role of machine learning techniques to reduce the false alarm rate in WLAN and proposed a filtration technique. | An intelligent agent based false alarm filter is recommended that undergoes a double filtration process to enhance the performance in terms of reduction of false alarm rate.<br>It suggested that to design new machine learning algorithms with the introduction of intelligent agents, soft computing techniques and fuzzy rules for better prediction accuracy on WLAN attacks. | 2018 |
| Kasongo and Sun [50] | A wireless IDS based on deep learning. | The proposed FFDNN system increases accuracy in comparison to other methods. | 2019 |
| Kasongo and Sun [51] | A Feed-Forward Deep Neural Network (FFDNN) wireless IDS system using a Wrapper Based Feature Extraction Unit (WFEU). | In the UNSW-NB15 dataset, 22 attributes were used and obtained 87.10% and 77.16% accuracy rate for the binary and multiclass classification schemes, respectively.<br>In the AWID dataset, 26 attributes were used and obtained overall accuracies of 99.66% and 99.77%. | 2020 |
| Satam and Hariri [52] | A WIDS that uses n-grams for modelling the normal behavior of the Wi-Fi protocol and machine learning models to classify Wi-Fi traffic flows. | The proposed approach detected attacks on Wi-Fi protocols with low false positives (0.0174) and a varying low rate of false negatives for different attacks. | 2020 |
| Riyaz and Ganapathy [53] | A new IDS to provide security in data communication by identifying and detecting the intruders in wireless networks.<br>A new feature selection algorithm called conditional random field and linear correlation coefficient-based feature selection algorithm. | The proposed model achieved better performance in terms of detection accuracy (98.8%), less training (0.57 s) and testing time (0.26 s).<br>The false alarm rate of the proposed model is less than 1% when it is compared with the existing CNN. | 2020 |
| | | | |
| Singh *et al.* [54] | A novel algorithm based on Deep learning technique. | In the NSL-KDD dataset, 97.32% and 97.47% accuracy rates were obtained for binary and multiclass categorization, respectively. | 2021 |

using the well-known data mining (NSL-KDD) dataset and it is compared to the current machine learning methods in the literature such as decision tree, support vector machines, Naïve Bayes and K-Nearest Neighbor. The obtained results showed that the proposed FFDNN system increases accuracy in comparison to other methods. However, this study can be improved with a new algorithm to increase the detection rates of R2L and U2R attacks, which are small attack clusters. Also, the proposed method can be applied to different datasets.

In this research, Kasongo and Sun [51] propose a Feed-Forward Deep Neural Network (FFDNN) wireless IDS system using a Wrapper Based Feature Extraction Unit (WFEU) to protect the various communication infrastructures using an intrusion detection system. The proposed extraction method uses the Extra Trees algorithm in order to create an optimal feature vector. The evaluation of the WFEU-FFDNN was examined by using UNSW-NB15 and the AWID datasets. Furthermore, the WFEU-FFDNN is compared to five different standard machine learning (ML) algorithms in the literature such as Random Forest, Decision Tree, SVM, KNN and Naive Bayes. The obtained results suggested that the proposed approach gives a higher accuracy rate than other approaches. In the UNSW-NB15 dataset, 22 attributes were used and obtained overall accuracies of 87.10% and 77.16% for the binary and multiclass classification schemes, respectively. In the AWID dataset, 26 attributes were used and obtained overall accuracies of 99.66% and 99.77%. This study can be improved by further reducing the number of features used.

Singh *et al.* [54] proposed a novel algorithm based on Deep learning technique to protect the Wireless networks from the attacks and detect any such activity. This proposed algorithm uses the Customized Rotation Forest algorithm for the aim of selecting features. The classification of different attacks is carried out by Gated Recurrent Units (GRU). The presented model was applied on NSL-KDD dataset and a 97.32% and 97.47% accuracy rate were obtained for binary and multiclass categorization, respectively. According to results, the proposed WIDS can be used for detecting attacks in real-time networks. However, this study can be improved by applying the proposed model on different datasets.

#### 3) EVALUATION OF WIRELESS IDSS

Wireless IDSs can detect WLAN protocol level attacks, misconfigurations, and policy violations by first examining IEEE 802.11a, b, g, and i protocol communication. They are also susceptible to denial-of-service attacks and physical attacks. Some Wireless IDS products perform signature-based detection only, while others use an integration of signature-based detection, anomaly-based detection, and situational protocol analysis techniques. Compared to other forms of IDS, wireless IDSs have higher intrusion detection accuracy. Although wireless IDSs offer powerful detection capabilities, they have some limitations: They cannot detect types of attacks against wireless networks, such as attacks involving passive monitoring and offline processing of wireless traffic.

### D. NETWORK BEHAVIORAL ANALYSIS

A network behavioral analysis (NBA) system examines network traffic or statistics on network traffic to identify unusual traffic flows such as distributed denial of service (DDoS) attacks, forms of malware (for example, worms, backdoors), and policy violations [55], [56].

NBA systems often include sensors and consoles; some systems also offer management servers. NBA sensors are usually only available as devices. Some sensors are similar to network-based IDS sensors in that they sniff packets to monitor network activity in one or more network segments. Other NBA sensors do not monitor networks directly, but instead use network flow information provided by routers and other network devices.

#### 1) SECURITY FEATURES OF NBA

NBA systems offer a variety of security capabilities. These are information collecting, logging and detection. Some systems also provide security information and event management capabilities.

##### *a: INFORMATION COLLECTION*

NBA systems offer extensive information-collecting capabilities and require knowledge of the mainframes' characteristics for detection. NBA sensors can create and store lists of hosts communicating on monitored networks. NBA systems obtain the following information for the detection of attacks:

- IP address;
- OS;
- IP protocols;
- TCP and UDP ports;
- Other connected hosts and used services.

NBA sensors constantly monitor network activity for changes in this information. Additional information about each server's stream is also collected on an ongoing basis.

##### *b: LOGGING*

NBA technologies log comprehensive data on detected incidents. This data can be used to validate alerts, investigate events, and correlate events among other sources. Data types frequently recorded by NBA software include:

- Date and time;
- Activity or alert type;
- Protocols;
- Source and destination IP addresses;
- TCP or UDP ports;
- Additional package header fields;
- Number of bytes and packets.

##### *c: DETECTION*

NBA technologies typically have the ability to detect several types of malicious activity. Most products primarily use anomaly-based detection, along with some stateful protocol analysis techniques, to analyze network flows. Most NBA technologies do not offer signature-based detection.

#### 2) RELATED WORK

The summary of network behavior analysis studies can be seen in Table 6. In Table 6, the main idea of each study and important aspects of the papers are discussed.

According to Youssef and Emam [57] study, traditional intrusion detection systems have limited capabilities and in most cases they cannot detect malicious behavior or raise an alarm (false positive) when there is no abnormality in the network. In this study, it is thought that the application of Data Mining (DM) techniques to network traffic data and the use of Network Behavior Analysis system will help develop better intrusion detection systems. Youssef and Emam proposed a hybrid intrusion detection approach. DM and NBA approaches for network intrusion detection are examined and it is claimed that a combination of both approaches can detect network intrusions more effectively.

Nitin *et al.* [58] analyzed and evaluated an IDPS technology called network behavior analysis system. A network behavior analysis system (NBAS) is basically an IDPS (intrusion detection and prevention system) technology that examines network traffic to identify threats that generate unusual traffic flows, such as distributed denial of service (DDoS) attacks, certain types of malware. A detailed evaluation of NBA technologies is presented in this article. First, the main components of NBA technologies and their architectures for deploying components are explained. In addition, the security capabilities of the technologies, including the methodologies used to detect suspicious activity, are examined in detail. The remainder of the section discusses the management capabilities of the technologies, including recommendations for implementation and operation.

In the study of Srivatav *et al.* [59], Principal Component Analysis (PCA) and Network Behavior Analysis with the KDD cup 99 dataset were used to detect new attacks or intrusions as well as existing attacks. Here, the PCA is used

**TABLE 6. Summary of network behavior analysis intrusion detection methods.**

| Paper | Proposed Method | Goal/Success | Year |
|---|---|---|---|
| Youssef and Emam [57] | DM and NBA approaches for network intrusion detection. | The combination of both approaches can detect network intrusions more effectively. | 2011 |
| Nitin *et al*. [58] | Analyzed and evaluated IDPS technology called network behavior analysis system. | The main components of NBA technologies and their architectures for deploying components are explained.<br>The security capabilities of the technologies, including the methodologies used to detect suspicious activity, are examined in detail. | 2012 |
| Srivastav *et al*.[59] | Principal Component Analysis (PCA) and Network Behavior Analysis (NBA) applied on KDD cup 99 dataset. | The proposed method is more promising in terms of detection accuracy and computational efficiency for real-time attack detection compared to previous systems in the literature. | 2013 |
| Moon *et al.* [60] | A decision tree-based intrusion detection system that performs behavioral information analysis to detect changed APT attacks. | The proposed system can quickly respond to APT attacks, detect the first possibility of intrusion and minimize the size of damage. | 2017 |
| Ghanshala *et al*. [61] | A lightweight and adaptive intrusion detection approach called Behavior Based Network Intrusion Detection (BNID). | BNID achieved an accuracy rate of 98.88% with 1.57% false positives. | 2018 |
| Pacheco *et al.* [62] | A methodology to develop an IDS based on anomaly behavior analysis to detect when an IoT network node is being compromised. | The obtained experimental results show that the proposed approach accurately detects known and unknown anomalies due to misuses or cyber-attacks, with high detection rate and low false alarms. | 2019 |
| Ahn *et al*.[63] | A new IDS called Hawkware. | Hawkware is light enough to be deployed and deployed on a Raspberry PI and can detect attacks at a fairly adequate level. | 2020 |
| Fladby *et al.* [64] | Investigated the use of adversarial techniques for adapting the communication patterns in order to evaluate the robustness of an existing NBA solution.<br>Implemented a packet parser that let us extract and edit certain properties of network flows and automated an approach. | Obtained results showed that network flow parameters could indeed be perturbed to ultimately enable evasion of intrusion detection models.<br>Additionally, it demonstrated that it was possible to combine evading detection with techniques for optimization problems that aimed to minimize the magnitude of perturbation to network flows. | 2020 |
| Khan [65] | A convolutional recurrent neural network is used to create a DL-based hybrid IDS framework that predicts and classifies malicious cyberattacks in the network. | According to the obtained results, the proposed HCRNNIDS gives better results than current IDS methodologies, attaining a high malicious attack detection rate accuracy of up to 97.75% for CSE-CIC-IDS2018 data with 10-fold cross-validation. | 2021 |
| Yang [66] | A behavior-based deep learning-based IDS. | The proposed system can effectively detect known or unknown malicious behavior in the existing network environment. | 2021 |

to size the dataset and the NBA to analyze the behavior of the network. KDD cup 99 dataset was used for training and testing of the proposed method. The main purpose of this article is to evaluate the malicious dataset and detect intrusions. According to the evaluation results, the proposed method is more promising in terms of detection accuracy and computational efficiency for real-time attack detection compared to previous systems in the literature. The method is also very effective in detecting new attacks and most known attacks. This work can be extended by using known datasets as well as online datasets.

APT (advanced persistent threats) attacks do simple attacks such as spear phishing during the initial intrusion, but in the long run after the initial intrusion, they leak information, creating a backdoor and analyzing the network to transmit malicious code. In this paper, a decision tree-based intrusion detection system that performs behavioral information analysis to detect changed APT attacks after a system intrusion is proposed by Moon *et al.* [60]. The proposed system first analyzes the behavior of malicious code and then determines the rule through the decision tree. According to the evaluation results, the proposed system can quickly respond to APT attacks, detect the first possibility of intrusion and minimize the size of damage. This study can be developed with different studies on distributed APT security methods in the network and system by standardizing the main behaviors created in the process.

In this article, Ghansahala *et al.* [61] propose a lightweight and adaptive intrusion detection approach called Behavior Based Network Intrusion Detection (BNID) at the cloud network layer. Traffic behavior analysis is performed on the Cloud Network Node (CNN) to detect intrusions. A security framework has been proposed for deploying BNID in the cloud. This eliminates the need to deploy the IDS to each tenant virtual machine (TVM). BNID uses feature selection and statistical learning techniques for traffic behavior analysis. The Information Technologies Operations Center (ITOC) attack dataset was used to evaluate the proposed approach. BNID achieved an accuracy rate of 98.88% with 1.57% false positives. This study can be developed with the proposed approach to detect unseen attacks.

In another study, Ahn *et al.* [63] propose a new IDS called Hawkware, an ANN-based distributed IDS that runs on an IoT device and analyzes the device's runtime behavior along with network traffic. The Hawkware system is designed to detect sophisticated attacks by analyzing device behavior, as opposed to the expensive, deep data analysis traditionally used. According to the evaluations, it is seen that Hawkware is light enough to be deployed and deployed on a Raspberry PI and can detect attacks at a fairly adequate level. This work can be developed not only for IoT devices, but also to work in different systems.

Yang [66] proposes a behavior-based intrusion detection system that uses deep learning for timely detection of malicious behavior in the network. The proposed system implements the Bidirectional Long Short Term Memory architecture and uses the UNSW-NB15 dataset for testing the proposed system. According to the authors, as a result of the experimental tests, the proposed system can effectively detect known or unknown malicious behavior in the existing network environment. This work can be further improved by solving the unbalanced dataset classification problem.

#### 3) EVALUATION OF NETWORK-BASED ANALYSIS

NBA systems basically work by detecting significant deviations from normal behavior, giving high accuracy in detecting attacks that generate large amounts of network activity in a short time and attacks with unusual activity. NBA technologies offer powerful detection capabilities for certain types of attacks, but they also come with significant limitations. One of them is that NBA systems are not very effective at detecting small-scale attacks. Especially if the attack is implemented slowly and does not violate the determined policies, the detection rate is not high. This is because NBA technologies use anomaly-based detection methods; they cannot detect many attacks unless their effectiveness is significantly different from what was expected. If a DoS attack starts slowly and builds up over time, it is likely to be detected but will be detected late. If the sensors are configured to be more sensitive to abnormal activity, alerts are generated faster when attacks occur, but this time the false positive rate may increase.

Another important limitation is the delay in detecting attacks. When NBA systems use streaming data from routers and other network devices, they are delayed in detecting attacks because of their data source. This data is transmitted to the intrusion detection system in batches over a period of one minute to several hours. If the attack happens quickly during this time, the attack has already corrupted the systems or may go undetected until the system is damaged. Therefore, it needs more powerful systems to monitor directly instead of using streaming data.

## IV. INTRUSION DETECTION METHODOLOGIES

Intrusion detection methodologies mainly divided into three distinct categories including:

1. Signature-based model;
2. Anomaly-based model;
3. Stateful protocol analysis;

Each IDS methodology uses a different technique to identify network attacks. For known attack types, signature-based detection is quite fast and effective, but it fails to recognize zero-day attacks. Anomaly-based methodology is effective to detect previous unseen network based attacks, but it raises false alarms. In other words, it classifies normal traffic as attacks. While stateful protocol methodology can detect some portion of the new attacks, it is resource intensive, complex, and cannot detect smart attacks. The details of each methodology are given below.

### A. SIGNATURE-BASED MODEL

A signature is a pattern that corresponds to a known attack. Signature-based detection is the process of comparing signatures with observed events to detect potential attacks [67]. In case of a match in the comparison process, the system will give a warning or additional report. Some examples of signatures are: An attempt to attack with the username "root" threatening the security of the network, an email with the subject of "Free programs" that is characteristic of known and common malware or an operating system indicating that host control is disabled in system log.

Signature-based detection is the simplest detection method because observed events are checked against a list of signatures using a comparison process. If there is a previously defined attack condition in the list, a warning is generated. Signature-based IDSs are very effective at detecting known threats, but are largely ineffective at detecting previously unknown threats or variants of known threats. For example, if an attacker replaced the malicious file "prog.exe" with the name "prog2.exe", a signature looking for "prog.exe" would not match.

#### 1) RELATED WORK

The summary of signature-based IDS methods can be seen in Table 7. in Table 7, where the main idea of the proposed method and success of each study are examined.

In the study of Kumar and Sangwan [68], signature-based attack detection was performed using Snort. While performing Intrusion Detection via Snort, DARPA Dataset was transferred over the network and focused on analysis of abnormal links detected during transmission. Snort is a popular NIDS for inspecting network packets and comparing them to a database of known attack signatures. In addition, the Snorts attack signature database may be updated from time to time. This IDS System has demonstrated that it can detect and analyze intrusions in real-time network traffic. According to the authors, this study will help new users to understand the concept of Snort-based IDS. This study can also be improved by applying and analyzing different intrusion detection tools.

One of the major challenges for signature-based IDSs is how to handle large volumes of inbound traffic when every packet has to be compared with every signature in the database. When an intrusion detection system can't keep up with the flood of traffic, it drops packets so potential attacks can be missed. Uddin *et al.* [69] proposed

**TABLE 7.** Summary of signature-based intrusion detection methods.

| Paper | Proposed Method | Goal/Success | Year |
|---|---|---|---|
| Kumar and Sangwan [68] | Signature-based attack detection was performed using Snort. | This IDS System can detect and analyze intrusions in real-time network traffic.<br>This study will help new users to understand the concept of Snort-based IDS. | 2012 |
| Uddin *et al.* [69] | A proposed a new Signature-Based Multi-Layer IDS model using mobile agents. | The proposed model is able to detect threats with a high success rate.<br>It also provides a mechanism to periodically update these small signature databases. | 2013 |
| Hubbali and Suryanarayanan [70] | Possible techniques for minimizing false alarm rate in signature-based Network Intrusion Detection System (NIDS) are examined. | Despite all known techniques, there are still problems that need to be addressed.<br>This study can help security researchers to implement a new post processing technique for IDS alerts. | 2014 |
| Rai *et al.* [71] | A decision tree algorithm based on the C4.5 decision tree approach. | The proposed Decision Tree Splitting (DTS) algorithm is an effective method for signature-based attack detection. | 2016 |
| Aldwairi *et al.* [72] | A vector algorithm is parallelized on a multi-core CPU under the MapReduce framework. | Phoenix++ and MAPCG MapReduce applications showed 1.3 and 1.7 times improvement over MPI, respectively. | 2017 |
| Baykara and Das[73] | A honeypot based approach for intrusion detection/prevention systems is proposed. The developed application is combined with IDSs to analyze data in real-time and to operate effectively. | The developed system is able to show the network traffic on servers visually in real-time animation.<br>It can detect zero-day attacks. This system also helps in reducing the false positive level in IDSs. | 2018 |
| Baykara and Das[74] | A centralized honeypot-based approach with a software-defined switching is proposed. | The proposed system has been run in GNS3 simulation software and good results have been obtained by reducing false alarm level, network traffic, and cybersecurity cost. | 2019 |
| Gunduz and Das[75] | The objectives, requirements, threats and potential solutions of the IoT-based smart grid are analyzed. | The paper presents specific solutions to threats on IoT-based smart grid applications and highlights possible research opportunities for researchers to provide future research directions. | 2020 |
| Malek *et al.*[76] | A new system to detect intrusions using a set of rules as a pattern recognized engine. | The combination of experimental results, SBID and PBID approaches provides a comprehensive system for intrusion detection. | 2020 |
| Otoum and Nayak[77] | An intrusion detection model called AS-IDS. | An attack detection rate of 96.9% was achieved on the NSL-KDD dataset. | 2021 |

a new Signature-Based Multi-Layer IDS model using mobile agents. The proposed model is able to detect threats with high success rate by creating and using small and multiple databases dynamically and automatically. It also provides a mechanism to periodically update these small signature databases using mobile agents. The proposed model can be developed as an automated system that can distribute, add and remove signatures between databases of multiple IDS systems.

Hubbali and Suryanarayanan [70] possible techniques for minimizing false alarm rate in signature-based Network Intrusion Detection System (NIDS) are examined. A classification, advantages and disadvantages of false alarm minimization techniques are given in signature-based IDS. In addition, several of the leading Security Information and Event Management tools that implement these techniques along with their performance are reviewed. According to the authors, despite all known techniques, there are still problems that need to be addressed. This study can help security researchers implement new post processing techniques for IDS alerts. Future research should also address different research issues that will increase the usability of the proposed techniques.

In the study of Rai *et al.* [71], a decision tree algorithm based on the C4.5 decision tree approach was developed. Feature selection and split value are important considerations for building a decision tree. In this study, the developed algorithm is designed to address these two issues. What is important is the information gain when choosing the features, and the value that will make the classifier unbiased against the most frequent values when choosing the split value. The proposed method was applied on the NSL-KDD dataset and the experiment was adhered to according to the number of features. The time taken to build the model and the accuracy obtained were used as metrics. According to the authors, the proposed Decision Tree Splitting (DTS) algorithm is an effective method for signature-based attack detection. This study can be developed by improving the split value and reducing the number of features used.

In this study, Aldwairi *et al.* [72] aim to reduce the matching load of the signature-based model and speed up the algorithm by parallelizing the signature matching algorithm on a multi-core CPU. In this paper, Myers algorithm, a vector algorithm, is parallelized on a multi-core CPU under the MapReduce framework. Approximately four times acceleration is achieved with the multi-core application compared to the serial version. They also used two implementations of MapReduce to parallelize the Myers algorithm. The implementation of the proposed method is compared with a previous message passing interface (MPI) based implementation of the algorithm. According to the results obtained, Phoenix++ and MAPCG MapReduce applications showed 1.3 and 1.7 times improvement over MPI, respectively.

Gunduz and Das [75] proposed a new system to detect intrusions using a set of rules as a pattern recognized engine. They used a PBID (Pattern Based Intrusion Detection) model, which confirmed previously applied SBID (Statistical Based Intrusion Detection) model. The proposed model was tested on the dataset produced within the scope of the study. 75% accuracy rate has been achieved. According to the authors, the combination of experimental results, SBID and PBID approaches provides a comprehensive system for intrusion detection. However, an effective result cannot be obtained with only signature-based attack detection. Therefore, this work can be further developed by integrating anomaly-based intrusion detection.

Malek *et al.* [76] propose an intrusion detection model called AS-IDS, which integrates these two approaches to detect known and unknown attacks in IoT networks. The proposed model consists of three phases: traffic filtering, preprocessing and hybrid IDS. In the first stage, network traffic is filtered at the IoT gateway by matching packet characteristics, then a Target Encoder, Z-score and Discrete Hessian Eigenmap (DHE) are applied respectively in the preprocessing stage. In the final stage, the signature base and the anomaly based model are combined. In the signature-based system part, the Generalized Suffix Tree (GST) algorithm is used and signatures are matched to classify attacks as intruder, normal or unknown. In the anomaly-based system part, it applies Deep Q-learning to identify unknown attacks and uses Signal to Noise Ratio (SNR) and bandwidth to classify attacks. The proposed AS-IDS model has been applied and tested in real-time traffic with the NSL-KDD dataset. An attack detection rate of 96.9% was achieved. Extensive experimental results can be obtained by applying this study on different datasets.

#### 2) EVALUATION OF SIGNATURE-BASED MODEL

Signature-based detection is the simplest detection method and it is easy to understand. The system compares activities such as packets or log entries against a list of registered signatures. Thus, users can control the signature database and the system administrator can easily understand which attack types will cause alarm. Signature-based IDSs are very effective at detecting known attacks, but are not effectively successful in detecting previously unknown threats, lurking threats, and any variant of known threats. In order to have a high success rate, a separate signature must be defined for all attacks that an attacker can make, and the signature database must be kept up-to-date.

### B. ANOMALY-BASED MODEL

Anomaly-based detection is the process of comparing observed activities with definitions considered normal to identify abnormal events [77]. An IDS using an anomaly-based detection system has rules that represent the normal behavior of users, hosts, network connections, or applications. These rules were developed by following the characteristics of ordinary activity over a period of time. For example, the rule for a network is the average usage time of web activity during business day hours. IDS then uses statistical methods to detect significantly higher-than-expected usage of web activity and to generate alerts while comparing characteristics of current activity with rules. Rules can be developed for many behavioral attributes, such as the number of emails sent by a user, the number of failed login attempts, and the number of packets transferred in a given time period. The most important advantage of anomaly-based detection methods is that they are effective in detecting previously unknown attack types. For instance, suppose a computer is infected with a new type of malware. The malware can consume the computer's processing resources, send a large number of emails, initiate many network connections, and engage in other behaviors that may differ significantly from the profiles created for the computer.

Rules created for anomaly-based detection are of two types: static and dynamic. Once created, the static rule list does not change unless the IDS is directed to create a new rule. A dynamic list is constantly updated as additional events are observed. As systems and networks change over time, the corresponding measures of normal behavior also change. A static list eventually expires, so it needs to be refreshed periodically. Dynamic profiles don't have this problem, but they are susceptible to hijacking attempts by attackers. For example, an attacker may perform a small amount of malicious activity, then slowly increase the frequency and amount of activity. If the rate of change is slow enough, IDS may consider malicious activity to be normal behavior and include it in its profile. Inadvertent inclusion of malicious activities as part of the rule is a common problem with anomaly-based IDS products.

Another problem with anomaly-based IDSs is that in some cases it can be difficult to get the rules right. For instance, if an event that performs large file transfers occurs only once a month, this behavior is not consistently observed, so it can be considered an abnormal situation and an alert may be triggered. Anomaly-based IDS products often produce many false positives due to benign activity that deviates significantly from the rules, especially in different or dynamic environments. Another major problem with the use of anomalous-based detection techniques is that due to the number and complexity of events, it is difficult to determine the cause of the alert or to verify that it is not a false positive.

#### 1) RELATED WORK

The literature review about anomaly-based detection methods is summarized in Table 8. The main idea of each paper and the pros and cons of each study have been summarized.

Samrin and Vasumathi [78] offered an anomaly-based intrusion detection system to reduce the number of false alarms and increase efficiency. Fuzzy rule-based modeling

**TABLE 8.** Summary of anomaly-based intrusion detection methods.

| Paper | Proposed Method | Goal/Success | Year |
|---|---|---|---|
| Geramiraz *et al.* [79] | An anomaly-based intrusion detection system. | Test results significantly improved the performance of the system by about 20% using adaptive IDS.<br>The proposed anomaly-based intrusion detection improved the accuracy of the system by around 15%. | 2012 |
| Yassin *et al.* [80] | Integrated machine learning algorithm based on K-Means clustering and the Naive Bayes Classifier (NBC) named KMC+NBC. | Performance evaluations were made on the ISCX-2012 dataset.<br>KMC+NBC increased the accuracy and detection rate up to 99% and 98.8%, respectively, while reducing the false alarm to 2.2%. | 2013 |
| Narsingyani and Kale [81] | Genetic algorithm (GA) based anomaly detection technique. | KDD99cup dataset was used and according to the results False Positive alarm rate can be reduced and detection speed can be increased. | 2015 |
| Harish and Kumar [82] | An anomaly-based method based on fuzzy clustering. | EDA dataset, which is a variant of the KDD dataset, was used.<br>86.3% accuracy and 17.04% false alarm rate were obtained. | 2017 |
| Aljawarneh *et al.* [83] | A new hybrid model. | An accuracy rate of 99.81% and 98.56% was obtained for the dual-class and multi-class NSL-KDD datasets, respectively. | 2018 |
| Tama *et al.* [84] | A method for selection of relevant features and an intrusion detection system based on two-level ensembles of classifiers. | An accuracy rate of 85.8% in the NSL-KDD dataset and 91.3% in the UNSW-NB15 dataset was achieved. | 2019 |
| Viegas *et al.* [85] | An IDS approach capable of processing evolving network traffic while being scalable to large packet rates is called BigFlow. | Experiments were made over a network traffic dataset spanning a full year. BigFlow can maintain high accuracy over time.<br>It requires as little as 4% of storage and between 0.05% and 4% of training time, compared with other approaches. | 2019 |
| Dwivedi *et al.*[86] | A new technique by combining Ensemble of Feature Selection and Adaptive Grasshopper Optimization Algorithm methods, called as EFSAGOA. | EFSAGOA has been evaluated on intrusion data as ISCX 2012. It has provided a high detection rate of 99.23%, accuracy of 99.13% and a low false alarm rate of 0.067. | 2020 |
| Eskandari *et al.*[87] | An anomaly-based intelligent intrusion detection system named Passban. | Passban can detect attacks with low false positive and high accuracy rates. | 2020 |
| Kumar *et al.*[88] | An anomaly-based intrusion detection system by decentralizing the existing cloud based security architecture to local fog nodes. | Proposed model is tested using an actual IoT-based dataset. The evaluation of the underlying approach outperforms in high detection accuracy and low false alarm rate using Random Forest algorithm. | 2021 |

and fuzzy controller were used to update the model during the creation and testing phase, respectively. In addition, the results of the system estimations are presented to the system user. Then the system user validates the decisions and the fuzzy controller adjusts the detection model using the system user's feedback. The NCL dataset was used in the evaluation of the system. The dataset is a subset of the KDD '99 dataset. According to the authors, their test results significantly improved the performance of the system by about 20% using adaptive IDS. In addition, the proposed anomaly-based intrusion detection improved the accuracy of the system by around 15%. This proposed intrusion detection system can also be tested on different datasets.

Geramiraz *et al.* [79] propose an integrated machine learning algorithm based on K-Means clustering and the Naive Bayes Classifier (NBC) named KMC+NBC to maximize detection and accuracy while minimizing false alarm. K-Means clustering has been applied to the labeling process. All the data are collected in their corresponding clusters as normal and aggressive according to their behavior with K-Means, while the Naive Bayes Classifier is used to re-categorize the misclassified data into the correct categories. Performance evaluations of KMC+NBC and NBC were made on the ISCX-2012 dataset. According to the results obtained, KMC+NBC increased the accuracy and detection rate up to 99% and 98.8%, respectively, while reducing the false alarm to 2.2%. This study can be extended to include feature selection methods.

Yassin *et al.* [80] applied Genetic algorithm (GA) based anomaly detection technique, which is one of the most effective evolutionary techniques in machine learning, for network attack detection. Since the decrease in the false positive rate will also increase the accuracy and performance, this study especially focused on the optimization of the false positive rate. The limitation of other accuracy techniques for their false positive rate is discussed in this paper. For the experiments, the KDD99cup dataset was used. According to the obtained results, False Positive alarm rate can be reduced and detection speed can be increased by using appropriate feature selection. This work can be improved by using dynamic feature selection techniques for selection of more important features.

Narsingyani and Kale [81] presented an anomaly-based method to detect anomalies in the network, based on fuzzy clustering. The proposed method consists of three stages: Preprocessing, Feature Selection and Clustering. In the preprocessing stage, duplicate data were eliminated from the dataset. Then, principal component analysis was applied to select the distinguishing features. In the clustering step, the network samples were clustered using the Robust Spatial Kernel Fuzzy C-Means (RSKFCM) algorithm. RSKFCM is a variation of the standard Fuzzy C-Means that takes into account neighborhood information and uses the kernel distance metric. In order to evaluate the proposed method, the EDA dataset, which is a variant of the KDD dataset, was used and compared with the standard techniques in the literature. Accuracy, false positive rate and cluster validity indices were used as performance measures. 86.3% accuracy and 17.04% false alarm rate were obtained. According to the authors, the proposed method gave better results than other methods. However, this study can be improved using different methods such as the Evolutionary algorithm.

Harish and Kumar [82] developed a new hybrid model based on optimal characteristics of network transaction data to estimate the intrusion coverage threshold. In the evaluation of the proposed model, 20% test dataset and NSL-KDD dataset, which is a binary and multi-class problem, were used. According to the results obtained, the hybrid approach has a significant effect on minimizing the computation and time cost while determining the feature association effect scale. An accuracy rate of 99.81% and 98.56% was obtained for the dual-class and multi-class NSL-KDD datasets, respectively. Besides, there are problems with high false and low false negative rates. In addressing these problems, a hybrid approach consisting of two main parts has been proposed. First, the Information Gain and Vote algorithm, which combines probability distributions, is used for the selection of important features that will increase the accuracy of the proposed model. Then, AdaBoostM1, REPTree, J48, Random Tree, Naïve Bayes, Meta Pagging and Decision Stump classification algorithms were used in the hybrid algorithm. As a result, improved accuracy, high false negative rate and low false positive status were observed. This study can be further developed by applying the proposed method on different datasets with different optimization techniques.

In the study of Aljawarneh *et al.* [83], a method for selection of relevant features and an intrusion detection system based on two-level ensembles of classifiers are proposed. In order to reduce the feature size of the training datasets, 3 different methods were used: particle swarm optimization, ant colony algorithm and genetic algorithm. Features are selected based on classification performance using a reduced error pruning tree (REPT). Then, rotation forest and bagging methods, which is a two-level group of classifiers, are applied. NSL-KDD and UNSW-NB15 datasets were used to evaluate the proposed system. An accuracy rate of 85.8% in the NSL-KDD dataset and 91.3% in the UNSW-NB15 dataset was achieved, significantly outperforming other recently proposed classification techniques according to the authors. This work can be further developed with new approaches that can achieve higher accuracy using fewer features.

Dwivedi *et al.* [86] propose an anomaly-based intelligent intrusion detection system (IDS), called Passban, that can protect IoT devices directly connected to it. The feature of the proposed system is that it can be deployed directly to cost-effective IoT gateways. Thanks to this feature, it can take full advantage of the edge computing paradigm to detect cyber threats as close to data sources as possible. Two different scenarios were applied during the Passban evaluation phase. In the first scenario, Passban was used as an IDS running directly on the gateway receiving data from IoT devices and the Internet. In the second scenario, ‘‘security in the box’’, a special device that receives traffic from the Internet and the local gateway, is used as the infrastructure element. According to the evaluation results, Passban can detect attacks with low false positive and high accuracy rates, including attacks such as HTTP and SSH Brute Force, Port Scanning and SYN Flood.

#### 2) EVALUATION OF ANOMALY-BASED MODEL

Anomaly-based detection is based on the principle of comparing traffic with what is considered normal to identify different situations. Anomaly-based intrusion detection systems are considered a better option than signature-based systems, as they do not require prior knowledge of the attack signature to detect an attack. But at the same time, the alarms generated by this system are more difficult to manage than signature-based intrusion detection systems. This may be because signature-based IDS generates information along with reported alarms, while anomaly-based IDS identifies traffic flow that is detected as malicious.

The anomaly-based IDS generates an alarm whenever it detects an activity that deviates from the basic pattern of normal behavior, but the cause of the anomaly is unknown to the intrusion detection system. This becomes a major challenge in managing alarms and distinguishing false positives from true alarms. Therefore, while anomaly-based detection systems can detect unknown attacks, it is also important to determine the classes of a detected attack.

### C. STATEFUL PROTOCOL ANALYSIS

Stateful protocol analysis is the comparison of predetermined profiles of generally accepted normal protocol activities for each protocol state with observed events to identify deviations. Unlike anomaly-based detection, which uses host-based or profiles belonging to a network, stateful protocol is based on universal profiles that specify how protocols should or should not be used.

The ‘‘stateful’’ mentioned in stateful protocol analysis means that an IDS has the ability to understand and monitor the state of the network, transport and application protocols. For example, when a user starts an FTP (File Transfer Protocol) session, the session is initially unauthenticated. In this case, unauthenticated users can only perform a few commands such as displaying help information or providing their username/password. After the user is successfully authenticated, the session is authenticated and users are

**TABLE 9. Summary of stateful protocol analysis.**

| Paper | Proposed Method | Goal/Success | Year |
|---|---|---|---|
| Mudzingwa and Agrawal [89] | A detailed review of main techniques used in intrusion detection and prevention systems. | Anomaly-based technique is superior to other techniques, but most of the IDPS use a combination of the main methodologies.<br>A combination of methods, there may be some confusion when trying to choose the appropriate methodology and system. | 2012 |
| Seo *et al.* [90] | A stateful SIP inspection mechanism called SIPAD. | The proposed approach significantly reduces the operating cost.<br>It can be used even in resource-constrained environments such as smartphones. | 2013 |
| Yang *et al.* [91] | A stateful Intrusion Detection System that uses the Deep Packet Inspection method. | A proposed approach specifically designed for the IEC 60870-5-104 protocol.<br>The new intrusion detection approach has been tested and validated. | 2014 |
| Kang *et al.* [92] | A framework for detecting smart grid attacks. | The attacks that can create dangerous situations can be detected effectively. | 2016 |
| Boite *et al.* [93] | The stateful paradigm is named StateSec. | StateSec detects and mitigates various attacks such as DDoS and port scans with high accuracy. | 2017 |
| Lewis *et al.* [94] | A filtering approach named as P4ID. | This system was evaluated by combining the CICS2017 dataset and the Emerging Threats rule set.<br>a significant reduction in traffic handled by IDS can be achieved. | 2019 |
| Sharma *et al.*[95] | A lightweight behavior rule specification-based misbehavior detection for the IoT-embedded cyber-physical systems (BRIoT). | The proposed approach is verified by an embedded system in an unmanned aerial vehicle. The feasibility of the proposed model is demonstrated with high reliability, low operational cost, low false-positives, low false-negatives, and high true positives in comparison with existing rule-based solutions. | 2019 |
| Rashid *et al.* [96] | A comprehensive and comparative analysis of the NSL-KDD and CIDDS-001 datasets. | k-NN, SVM, NN and DNN classifiers have approximately 100% accuracy in the NSL-KDD dataset, and approximately 99% accuracy in the k-NN and Naïve Bayes classifiers CIDDS-001 dataset. | 2020 |
| Sbai and Elboukhari [97] | An IDS using deep neural network technology to detect the subclass of the big class DDoS: Data flooding attack. | The proposed model evaluated on the dataset CICDDoS2019. The obtained results show that the proposed architecture model achieves interesting performance (Accuracy, Precision, Recall and F1-score). | 2020 |
| Choudhary and Kesswani [98] | A hybrid classification approach to detect multi-class attacks in the IoT network. | The 81.02% detection rate, 2.22% false alarm rate and 92.85% detection rate, 2.99% false alarm rate were obtained respectively on UNSW-NB15 and NSL-KDD dataset. | 2021 |

expected to execute any of the commands. That is, commands are considered suspicious to be executed in an unverified state, while performing them in an authenticated state is considered normal.

"Protocol analysis" in situational protocol analysis usually includes checks such as minimum and maximum lengths for individual commands. If a command typically has a password argument and that argument is a maximum of 10 characters long, a 20-character argument is suspect. Stateful protocol analysis methods often use protocol models based on software vendors' and organizations' standards. For specific protocols, information is often not available and clear. In this case, it is difficult for intrusion detection systems to perform comprehensive and accurate analysis.

### 1) RELATED WORK

The summary of stateful protocol analysis is given in Table 9. The related studies are examined based on the main idea, proposed method, and pros and cons of each detection method.

Kumar *et al.* [88] presented a detailed review of four main techniques used in intrusion detection and prevention systems. These are anomaly-based, signature-based, stateful protocol analysis, and hybrid-based techniques. A detailed description of each methodology and a comparison of these methodologies are given. According to the results of these comparisons, the anomaly-based methodology is superior to other techniques in detecting new threats without any updates or inputs for users, but most of the IDPS available use a combination of the main methodologies. In systems that use a combination of methods, there may be some confusion when trying to choose the appropriate methodology and system.

In summary, the authors presented their methodologies and comparisons used by IDPS products on the market. In addition, tests of the systems used in the market can be given in order to develop this study.

Many mobile VoIP applications have been released recently and have become attractive targets for attackers. They are especially important attacks as they target calling procedures and system resources and cause service interruption. In the study of Mudzingwa and Agrawal [89], a stateful SIP inspection mechanism called SIPAD, which is optimized to detect malformed SIP messages and SIP flood attacks, is proposed. Depending on the current situation and message type, the relevant branches are determined from the SIPAD stateful rule tree and the structure of a SIP message is compared with the branches. The optimized rule tree provides higher detection accuracy and faster detection than existing approaches. Traditional SIP schemes are costly due to the complexity of rule matching mechanisms. According to the authors, the proposed optimized approach significantly reduces the operating cost and can be used even in resource-constrained environments such as smartphones. This study can be developed not only for 2 attack types, but also for the detection of different attack types.

Cyber threats in Supervisory Control and Data Acquisition (SCADA) systems have the potential to physically damage and disrupt power system operation. How to develop a robust intrusion detection system that will meet SCADA requirements is one of the emerging problems. Seo *et al.* [90] proposed a stateful Intrusion Detection System that uses the Deep Packet Inspection method to improve the security of SCADA systems using the IEC 60870-5-104 protocol. A proposed stateful protocol analysis approach specifically designed for the IEC 608705-104 protocol is presented. Finally, the new intrusion detection approach has been tested and validated.

Smart grid systems use a set of application protocols for their devices and services. The information in these application protocols is useful for intrusion detection systems, which are important security solutions. In situational analysis-based intrusion detection systems, network and system behaviors are monitored and case records are kept to make detection decisions.

Monitoring these behaviors in smart grid systems requires expert knowledge and adaptation for specific application protocols. In this work, Yang *et al.* [91] present a framework for detecting smart grid attacks that can be run on Suricata, allowing it to define stateful rules. A stateful rule defines the state of smart grid devices and examines incoming network traffic to find any matches. In addition, an application has been developed to demonstrate the implementation of the proposed framework and for IEC 61850 stateful analysis. In the experiment with the presented test environment, it is claimed that the attacks that can create dangerous situations can be detected by the proposed framework. This work can be improved by adding functions for other application protocols for smart grid systems.

The study of Kang *et al.* [92] is based on the stateful paradigm to protect edge hosts from DDoS attacks. A new approach, named StateSec, based on in-key processing capabilities, is proposed to detect and mitigate DDoS attacks. StateSec monitors matched packets without consulting the controller, thanks to its configurable traffic features. By feeding an entropy-based algorithm with monitoring features in this way, StateSec detects and mitigates various attacks such as DDoS and port scans with high accuracy. The implementation of StateSec has been compared to sFlow, a new technology approach to monitoring traffic on SDN, and StateSec is more efficient, according to the authors. It also has a very sensitive detection level. This study can be developed for the detection of different types of attacks.

Boite *et al.* [93] have introduced P4ID as a solution to the challenges of deployment and scalability of Intrusion Detection Systems. P4ID is a filtering approach to intrusion detection that uses a P4 stateful data plane to dynamically determine whether traffic should be forwarded to an intrusion detection system. In the P4ID system, the P4 rule parser is used and stateless/stateless packet processing is combined. This system was evaluated by combining the CICS2017 dataset and the Emerging Threats rule set. According to the authors, a significant reduction in traffic handled by IDS can be achieved with the proposed technique. This work can be further enhanced by adding elements to limit traffic to the IDS.

Sharma *et al.* [95] performed a comprehensive and comparative analysis of the NSL-KDD and CIDDS-001 datasets. In order to obtain ideal results, Neural Networks used hybrid feature selection and ranking methods (combination of signature-based, anomaly-based and stateful protocol analysis) before applying classification approaches such as Naïve Bayes, SVM k-NN, DAE and DNN. The performance of the IDS was evaluated using metrics such as recall, precision, accuracy and f1-score. Experimental results have shown that the k-NN, SVM, NN and DNN classifiers have approximately 100% accuracy in the NSL-KDD dataset, and approximately 99% accuracy in the k-NN and Naïve Bayes classifiers CIDDS-001 dataset. According to the results of the study, hybrid methods should be preferred for better performance of the state-of-the-art classifiers. This study can be developed by applying different deep learning approaches on current datasets such as CIC-IDS 2017, CIC-IDS 2018.

### 2) EVALUATION OF STATEFUL PROTOCOL ANALYSIS

Stateful protocol analysis methods generally use standard protocol models and take into account changes in the implementation of each protocol. Thus, it can detect an unexpected sequence of commands and follow the aforementioned profiles in both the network layer and the application layers. However, many standards have not fully explained the details of the protocols. In addition, many vendors violate the standards or add special features, some of which can replace the standards. For such special protocols, clear information about the protocols is often not available. In this case, it is not possible to make comprehensive and accurate analyzes.

A disadvantage of stateful protocol analysis methods is that the analysis process is complex and resource usage is high due to the simultaneous tracking of many sessions. Another problem is the inability to detect types of attacks that do not violate the characteristics of protocol behavior. For example, performing many normal actions in a short time to cause a denial of service. Another problem is that the way the protocol model used by an IDS is implemented on certain versions of operating systems or on different clients/servers may differ.

## V. INTRUSION DETECTION APPROACHES

There are several approaches to detect intrusions in the computer networks including statistical, rule, heuristic, pattern, cloud, machine learning, and deep learning based. The name varies based upon the techniques and platforms that are used during the detection process.

### A. STATISTICAL-BASED

Statistical based (statistic-based) intrusion detection system monitors the normal transactions to build a legitimate profile [10]. When the observed events are different from the normal (legitimate) profile, it is the indicator of the attacks. For each transaction, a score is assigned to deviate intrusions from the normal traffic. When the measured score is bigger than a threshold value, the alarm is triggered. The threshold value is set based upon the number of events that occur in the specified time period. Statistical metrics such as mean, mode, median, variance, and standard deviation are used when building the normal profile [98]. The Statistic based techniques vary. Mainly it can be classified into three categories: univariate, multivariate, and time series mode. These statistical based techniques can also be divided into subcategories such as time series model, operational model, markov process model, parametric and nonparametric models, threshold metric, statistical moments, and multivariate model.

The statistic based IDS includes some advantages which can be listed as follows:

1. It can recognize zero-day attacks because attack signature is not needed
2. It is easy to maintain because no need to update
3. It can detect DoS, and DDoS attacks

On the other hand, drawbacks of statistic based approach can be listed as the following:

1. Building a normal profile takes time
2. Normal profile can change overtime
3. Statistical distributions used need to be effective and accurate

### B. RULE-BASED

Rule based IDS applies rules when detecting potential intrusions in the network traffic [99]. The rule based IDS uses technology more than manual work. The rules can be defined as a pattern of patterns. When rules are extracted, artificial intelligence is being used to derive rules from the attack patterns. By single rule, many attacks can be recognized. To recognize the same types and number of attacks, rule-based approach requires only a few rules while signature-based approach needs thousands of signatures. It is easy to maintain rule based IDS. After the first rules are defined, additional rules can easily be added to the system by the service provider. Rule based detection systems can detect new attacks because simple changes in attack could not change intrusion patterns completely. However, it requires several rules to detect all possible attacks in the network.

### C. HEURISTIC-BASED

Heuristic based IDS searches for intrusions based on malicious behaviors. The heuristic based IDS builds a detection model which specifies acceptable behaviors and revoke for any other behaviors. To correctly identify the attack behaviors from normal behaviors, the heuristic approach requires knowledge and experiences. In the heuristic approach, collected execution traces are analyzed for any suspicious behaviors. If any suspicious behavior pattern is found, the alert will be raised. Timely, malicious behaviors list can be extended when new types of behaviors are detected in network attacks. With heuristic based approaches, well-known as well as zero-day attacks can be detected. However, some attack types can use hiding techniques to escape from the heuristic detection engine.

### D. PATTERN-BASED

Pattern based approach identifies the characters, strings, and forms to extract the meaningful patterns in the collected data, and find attacks based on those patterns [100]. The pattern based approach specifies the malicious instruction sequences in the attacks. The recognized patterns in the intrusion detection system known as signatures. Pattern based approach can detect known attacks fast and efficiently, but it cannot detect most of the zero-day attacks because their signatures (patterns) are not known yet. It is easy to implement a pattern based detection system. Pattern matching algorithms can be divided into two types including single and multiple pattern matching. The multiple matching algorithm is more efficient for current IDS because it avoids scanning the packet several times [100]. However, it requires more memory and processing time.

### E. CLOUD-BASED

Cloud computing is one of the most promising technologies which support several services online efficiently [101]–[104]. Cloud computing provides many advantages including pay-as-you-go, 24/7 data access, unlimited services with lower cost, and more computational power. Cloud environments provide three types of services: SaaS (software as a service), PaaS (platform as a service), and IaaS (infrastructure as a service) [104]. SaaS provides great benefits for users to run complex activities over the internet. PaaS presents a huge software platform for users to implement various algorithms and techniques easily. IaaS delivers users as well as companies' infrastructure including network, fundamental computing devices, and storage resources. Building IDS on top of the cloud environment brings several advantages. Cloud environments enable the identification of malicious network activities from different perspectives and overcome the classical intrusion detection deficiency [105]. In addition, using public and private clouds gives the opportunity to detect various types of network attacks in parallel with high performance.

Cloud based IDS comprises three different components including user data collector, cloud service, and cloud intrusion detection [105]. The user data collector is an independent server which collects network packets, filter, standardize, and sends to the cloud service. The cloud service analyzes and validates the received data from the user data collector, and translates the data into a common format for cloud intrusion detection component. Cloud intrusion detection component is the main part of detecting intrusion at the cloud. It takes the data from the cloud service component, analyzes the packets, and specifies the intrusions. Cloud intrusion detection component consists of a signature database, service console, and analysis engine. The cloud-based IDS approach is still at the early stage. This approach should be used more in intrusion detection systems in the future to increase model performances.

### F. MACHINE LEARNING-BASED

The purpose of machine learning is to automate the analysis process without human intervention. In other words, it is an algorithmic way to describe the data. Machine learning approach can use different learning techniques including supervised (labeled/tagged data), unsupervised (untagged data), and semi-supervised (few tagged, several untagged data). It is a new approach that has been used for intrusion detection. There are a broad category of techniques and algorithms which are used in IDSs including Bayesian algorithms, k-nearest neighbor, support vector machines, decision trees, neural networks, genetic algorithms, and fuzzy logic. Recently, several papers have been published which apply the machine learning approach for IDSs [106]–[108].

The primary advantages of ML approach are adaptability, high performance, flexibility, and detect new types of attacks. On the other hand, there are some drawbacks of ML-based IDS which can be listed as the following:

1. ML algorithms make assumption about the data
2. ML algorithms prone to bias (history, knowledge, vs.)
3. Cannot handle outliers all the time
4. Detection and prevention of unknown attacks are challenging
5. The complexity of the attacks is increasing
6. Large data size (millions of network connections)
7. High dimensionality (hundreds of dimensions are possible)
8. Data preprocessing is difficult (Converting data from monitoring system into appropriate format for analysis)
9. Need contextual features not just IP addresses
10. Algorithms do not take domain knowledge into account
11. The process is (feature engineering, parameter choices, etc.) more important than the algorithms

### G. DEEP LEARNING-BASED

Deep learning is a new way of learning which has become popular recently. It is a subfield of machine learning which learns from examples, and most of the time it eliminates the feature engineering step. Deep learning uses multiple consecutive layers during the learning process. Layers are connected and the previous layer's output is the input of the posterior layer. Features are extracted in a hierarchical manner. In each layer, the features are identified more clearly. Deep learning has been used in many different areas including image processing, human recognition, face recognition, driving safety, and malware detection [109]. Recently, it has been used in intrusion detection systems as well. Deep learning based IDS can detect anomalies by taking advantage of classification and dimension reduction. Besides, deep learning can recognize large-scale and multi-dimensional data [110]. Deep learning approach can handle dynamic data which changes in a timely manner. The deep learning model classifies the new traffic as normal or attack. It can also classify further by specifying the types of attacks [111], [112].

The advantages of deep learning approach in detecting intrusions can be listed as follows:

1. Auto feature extraction
2. Handle very large datasets
3. It is powerful, effective and reduces feature space drastically

The disadvantages of DL-based IDSs approach can be listed as follows:

1. Lots of data available, but a single record does not indicate good/bad
2. Not enough information within flows – need context knowledge
3. Bad understanding of the data to engineer meaningful features
4. No well trained domain experts and data scientists to check the implementation
5. Not enough labelled data
6. Building a hidden layer takes time and adding extra hidden layers rarely increases the model performance
7. it is not resistant to evasion attacks (Crafted inputs lead to deceive DL)

### H. EVALUATION OF INTRUSION DETECTION APPROACHES

In this section IDS approaches (statistical-, rule-, heuristic-, pattern-, cloud-, machine learning-, and deep learning based) are summarized based on the techniques that are used, main idea of each detection approach, advantages, and disadvantages of each detection approach. The evaluation of each approach can be summarized as follows:

- ➢ The **statistic based IDS** approach can detect known intrusions as well as zero-day attacks. It does not need updating every time. It can detect a wide range of attack types. However, building a normal profile takes time and can change overtime.
- ➢ The **rule based IDS** requires only a few rules to detect thousands of attacks. It recognizes new attack variants because simple changes in attack codes cannot change

intrusion patterns exactly. But, it requires multiple rules to identify all possible attacks in the network.

- The **heuristic based IDS** can extend malicious behaviors list when detecting new attacks' behaviors. With a heuristic based approach both known and unknown attacks are detected. However, some attack variants use evading techniques to remain undetected in the communication networks and hosts.
- The **pattern based approach** determines known attacks fast and efficiently, but it cannot detect most of the unknown attacks because their signatures (patterns) are not available yet. The implementation of a pattern based approach is fast and easy.
- The **cloud based approach** brings several advantages including unlimited services with lower cost, 24/7 data access, more computational power, etc. Different IDS techniques can be applied on top of the cloud. The cloud based IDS consists of a signature database, service console, and analysis engine. It increases the IDS performance as well as decreasing analysis time.
  The cloud-based IDS approach is still at the early stage, and it should be used more in IDSs in the near future.
- The **machine learning based approach** automates the analysis process to detect intrusions. It uses various learning techniques including supervised, unsupervised, and semi-supervised as well as different algorithms including decision trees, k-nearest neighbor, support vector machines, neural networks, genetic algorithms, and fuzzy logic. The main advantages of the machine learning approach is to support high performance, adaptability, flexibility, and detect zero-day attacks. However, there are several disadvantages of ML-based IDS including prone to high bias, cannot handle outliers, difficult to handle large data, data preprocessing is difficult, and algorithms do not take domain knowledge into account.
- The **deep learning based approach** uses multiple consecutive layers during the learning process. It can detect anomalies in the data, can recognize large-scale and multi-dimensional data, and can handle dynamic data which changes overtime. On the other hand, there are several cons of DL based approach including not enough information is available within flows, bad understanding of the data to engineer the meaningful features, no well trained domain experts to check the implementation, not enough labelled data, and not resistant to evasion attacks.

To sum up, each detection approach has its own advantages and disadvantages, and performs better on different datasets. Various features can be used to evaluate the performances of IDS approaches including the size of the data, dimensionality of the data, number of available features, the distribution of the data, etc. We concluded that statistical-, heuristic-, and pattern based approaches are used sufficiently in the current IDS. Thus, researchers need to focus more on cloud-, machine learning-, and deep learning based approaches. Besides, when building an intrusion detection system, researchers and developers need to be aware of various evasion techniques including: address spoofing, avoiding defaults, pattern change evasion, coordinated, low-bandwidth attacks, and fragmentation.

## VI. DATASETS USED IN IDSs

To evaluate the effectiveness of intrusion detection systems, the proposed IDS needs to be tested on well-known datasets. To build a reliable IDS dataset, initially, network flows are collected by using packet analyzing tools. Then, collected network flows are analyzed manually, or automatically in order to gather significant network features. Network flows consist of source and destination IP addresses, source and destination ports, packet length, type of network services, and failed login attempts. [113]. After the dataset is being created, intrusion detection systems extract attack patterns to detect and classify the network attacks. This work investigated several publicly available datasets including KDD '99, CAIDA, NSL-KDD, ADFA-LD and ADFA-WD, AWID, UNSW-NB15, and CICIDS for intrusion detection systems. These datasets are well-known for network intrusion detection and used in many scientific and business studies. Since the network attack types are changing overtime, IDSs datasets and features must be updated [114] to fulfil the current needs.

### *A. KDD'99 DATASET*

First dataset which includes network traffic flows was DARPA (Defense Advanced Research Project Agency) [115]. The DARPA dataset was created in 1998 in MIT Lincoln LAB and consists of raw data TCP packets dump [10]. Since the DARPA dataset consisted of raw data, performing machine learning classification algorithms was not possible. Before performing machine learning classifiers, features were needed to be extracted. In 1999, Knowledge Discovery and Data Mining (KDD '99) dataset, which is a feature extracted version of the DARPA dataset, was proposed. KDD '99 dataset has a labeled class which is categorized into five groups: DoS (denial of service) attack, remote to local (R2L) attack, user to remote (U2R) attack, probe attack, and normal. For training, the dataset contains 24 different attacks, and for testing, the dataset contains 14 unknown attacks. In KDD '99 dataset, attack types and normal class labels are not equally distributed. Besides, it is quite a big dataset, contains many redundant features, and does not contain recent network attacks.

### *B. CAIDA DATASET*

CAIDA dataset consists of one hour of anonymized network traffic traces in 2007 [116]. This dataset contains DDoS attacks which try to block legitimate users from accessing the targeted servers. The drawback of CAIDA dataset is that it does not contain a diversity of network attacks, as well as not whole network data is included. In addition, the CAIDA dataset is not labeled and consists of 20 features.

### C. NSL-KDD DATASET

NSL-KDD is an improved version of the KDD99 dataset. In the KDD dataset, several features are repeated. Due to the huge percentage of duplicate records, the performance of machine learning is decreasing overtime. Besides, the size of the KDD dataset is very big which makes ML analysis challenging. On the other hand, Hick *et al.* [117] created an NSL-KDD dataset from KDD which removed the duplicate records in the KDD dataset, and decreased the size of the dataset as well. The NSL-KDD training dataset comprises 125,973, and the test dataset consists of 22, 544 instances [10]. In addition, the NSL-KDD has 22 intrusion attack labels for training with 41 features. NSL-KDD dataset is appropriate to test the current IDSs. In order to increase the ML performances further, feature engineering can be applied on NSL-KDD dataset before performing ML algorithms.

### D. ADFA-LD AND ADFA-WD DATASET

In 2014, researchers who work in the Australian Defence Force Academy (ADFA) created two modern intrusion detection datasets, namely: ADFA-LD (Linux dataset) and ADFA-WD (Windows dataset) [118]. The datasets have samples from both Windows as well as Linux operating systems. The datasets contain host-based intrusion detection system (HIDS) system-call traces. The data consist of three different datasets including: training, validation, and attack [10]. The attack dataset contains usual as well as zero-day attacks.

### E. AWID DATASET

Aegean Wi-Fi Intrusion Dataset (AWID) contains labeled data which includes real 802. 11 Wi-Fi network traffic data can be categorized as intrusion and normal class [119]. The dataset has two distinct dataset training and testing data.

The AWID dataset comprises two labeled datasets, namely: high level and finer grained. The high level dataset contains 4 labels including Injection, Flooding, Impersonating, and normal [119]. The training dataset comprises 10 classes including Arp, Authentication request, Amok, Cafe latte, Beacon, Evil twin, Probe response, Deauthentication, fragmentation, and normal. The test dataset contains 17 classes. The large dataset of AWID includes 162,375,247 records for training, and 48,524,866 records for testing [120]. The reduced dataset has 1,795,575 records for training, and 575,643 for testing. AWID dataset is important for current IDSs tests because performing feature engineering and applying machine learning algorithms are easy on this dataset.

### F. UNSW-NB15 DATASET

UNSW-NB15 was generated in 2015 which consists of comprehensive network traffic. The raw network traffic is generated with the IXIA tool [121]. The generated traffic is captured with tcpdump. The dataset contains 49 features, 9 unique attack types, and 2, 540, 044 records [122]. The dataset consists of 4 csv files including UNSW-NB15_1.csv, UNSW-NB15_2.csv, UNSW-NB15_3.csv, and UNSW-NB15_4.csv. The training set contains 175,341 records, and the test set contains 82,332 records from 9 attack categories and normal. Since the UNSW-NB15 dataset contains new attack types as well as comprehensive features, it can be pretty favorable for testing modern IDSs.

### G. CICIDS 2017 DATASET

CICIDS dataset created in 2017 and contains both attacks and normal network traffic. The data collected from modem, switches, routers, firewall, and several operating systems such as different versions of Windows, Linux and macOS [10]. In the dataset, there are different attack profiles including brute force (FTP, SSH), Infiltration, DoS, DDoS, Heartbleed, Botnet, and Web attack [123]. The CICIDS dataset consists of 80 features.

### H. CIC-DDoS2019 DATASET

The CIC-DDoS2019 dataset contains benign as well as various types of DDoS attacks. Dataset has 88 features with hundreds of thousands records. There are several separated.csv files which shows the different DDoS types based on the protocol that are used including NTP, DNS, LDAP, MSSQL, NetBIOS, SNMP, SSDP, UDP, UDP-Lag, WebDDoS, SYN, and TFTP. It is a novel dataset which consists of unknown attacks. This dataset can be used to measure the effectiveness of the current IDSs.

### I. BoT-IoT DATASET

The BoT-IoT dataset contains network traffic from legitimate and emulated IoT networks. The dataset's source files are in different formats including pcap, generated argus and csv files. The dataset is publicly available and comprises 49 Features. There are 72.000.000 records in the dataset. The dataset has different types of attacks such as OS and service scan, DoS, DDoS, keylogging and data exfiltration. The BoT-IoT dataset consists of realistic network traffic as well as a diverse attack scenario which might be effective to test current IDSs on it.

### J. EVALUATION OF DATASET USED IN IDSs

In order to evaluate the effectiveness of the current IDSs, the well-known IDS datasets are examined. Each dataset has its own pros and cons, and works better for different situations. The comparative table of current intrusion detection datasets can be seen in Table 10. The KDD '99 dataset is the biggest and most used dataset for IDSs, but this dataset has many redundant features which makes the ML classification process challenging. The NSL-KDD dataset is a modification of KDD. It is quite effective to test modern IDSs on NSL-KDD dataset, but this dataset lacks modern network attacks. Other datasets including CAIDA, ADFA-LD

**TABLE 10. Comparison of well-known IDS dataset.**

| Number | Dataset name | Year | Features | Attack types | Labeled/ Unlabeled | Number of instances |
|---|---|---|---|---|---|---|
| 1 | DARPA | 1998 | No-features | Dos, U2R, R2L, Probe | Unlabeled | - |
| 2 | KDD '99 | 1999 | 41 | Dos, U2R, R2L, Probe, Normal | Labelled | 4,900,000 |
| 3 | CAIDA | 2007 | 20 | DDoS, Normal | Unlabeled | - |
| 4 | NSL-KDD | 2009 | 41 | Dos, U2R, R2L, Probe, Normal | Labeled | 125,973 training<br>22, 544 testing |
| 5 | ADFA-LD and ADFA-WD | 2014 | 26 | Stealth, Webshell, Zero-day, Dydra, Meterpreter, Adduser | Labeled | 13,675 traces |
| 6 | AWID | 2015 | 155 | Injection, Flooding, Impersonating, Normal | Labeled | 162,375,247 training<br>48,524,866 testing |
| 7 | UNSW-NB15 | 2015 | 49 | Analysis, Backdoors, DoS, Exploits, Generic, Fuzzers, Reconnaissance, Shell code, Worms | Labeled | 2,540,044 |
| 8 | CICIDS | 2017 | 80 | Brute force, Infiltration, DoS, DDoS, Heartbleed, Botnet, ,Web attack | Labeled | - |
| 9 | CIC-DDoS2019 | 2019 | 88 | Benign, Various DDoS attacks | Labeled | Hundreds of thousands records |
| 10 | BoT-IoT | | 49 | OS and service scan, DoS, DDoS, keylogging, data exfiltration attacks | Labeled | 72.000.000 |

and ADFA-WD, AWID, UNSW-NB15, and CICIDS have different deficiencies.

The CIC-DDoS2019 and BoT-IoT datasets contain recent intrusions in the network traffic which can be used to measure the effectiveness of current IDSs. These datasets are popular for network intrusion detection systems and used in several scientific studies. Since the network attacks are evolving overtime, IDS datasets and features must be updated from time to time in order to evaluate the future network attacks accuracy.

## VII. IDS TOOLS

There are a number of unique IDS tools to detect intrusions in the communication networks. There are mainly two types of IDS tools, namely: open source and commercial. Open source IDS tools have several advantages including simple license management, lower hardware and software costs, a lot of support, and no vendor restrictions. On the other hand, commercial IDS tools have clear usage policy, high-quality software, more funds for development and maintenance, timely updates, and help for problems. General working principle of IDS tools can be seen in Figure 2.

### *A. SNORT*

Snort is open source HIDS and IPS which supports various operating systems such as Unix, Linux, FreeBSD, MacOs, and Windows. The first version of snort was released in 1998 by Martin Roesch. Snort captures the network traffic and analyzes the captured traffic to detect attacks in real-time [124]. Snort detects network intrusion such as Dos, DDos, port scans, nmap scans, SBM probes, CGI attacks, and NetBIOS queries. [125]. Snort consists of different components including packet capture module, packet decoder, preprocessors, detection engine, logging and alerting system, and output module.

In general, the snort IDS tool works as follows. Snort first uses a packet capture library to collect network packets from various network interfaces. Then, use a packet decoder to analyze the packet headers. For further analysis, data is decrypted. In the preprocessors phase, IP defragmentation takes place. In this process, TCP stream reassembly, HTTP, HTTPS, FTP, SMTP, SSH, etc [126]. After that, inspected packets results are compared with snort rules to decide whether each packet has intrusion or not. Based on the output, snort generates alerts. Snort has several advantages in open source IDS tools. Advantages can be listed as follows:

1. Snort can be used as IDS as well as IPS
2. It can be installed on different types of networks as well as operating Systems
3. It has enormous signature database for known attacks
4. It is scalable
5. Live and real-time monitoring possible
6. Any user can write its own rules
7. Weaknesses in the snort can easily find and publish

There are some disadvantages of snort which can be listed as follows:

1. It is challenges to detect fragmented packets
2. In the large networks, it is difficult and expensive to monitor the packets
3. Rule database is extremely large
4. It does not stop attacks in progress

### *B. SURICATA*

Suricata is an open source IDS and IPS tool. It supports several operating systems including Unix, Linux, MacOS, Windows, and FreeBSD. The Suricata IDS tool is working similar to snort tool. It supports multithreading and consists of three modules: capture, decode, and detect [124]. In the capture module, data is collected from the network interfaces.

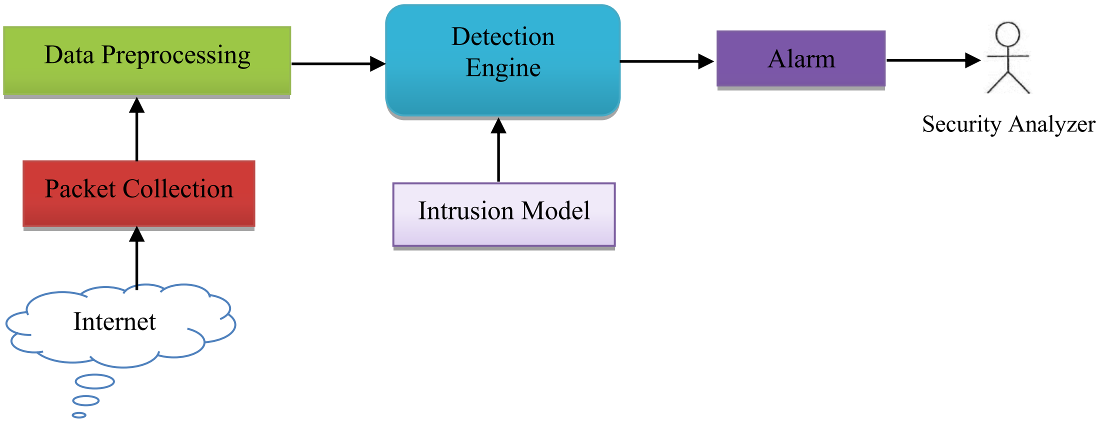

**FIGURE 2.** General working principle of intrusion detection tools.

In the decode module, packets are decoded to support the Suricata structure. In the detection module, packets are decided to be attack or normal based on predefined rules, signatures, and anomalies.

The pros of Suricata can be listed as follows:

1. It supports multithreading
2. It supports various operating system
3. It produces high performance
4. It detects protocols automatically
5. It filters the events and alarms
6. It collects data at the application layer
7. It filters the events and alarms
8. It Works as intrusion detector, network security monitoring tool, and inline intrusion prevention tool in realtime
9. It supports higher-level protocols including SMB, HTTP, FTP, as well as lower-level protocols such as TLS, UDP, TCP, ICMP [127]
10. It supports higher-level protocols including SMB, HTTP, FTP, as well as lower-level protocols such as TLS, UDP, TCP, ICMP [128]
11. It ensures some file extraction capability to network administrators in order to analyze suspicious files manually
12. It supports signature and anomaly-based IDS methods
13. It supports third-party tools including built-in scripting module, Anaval, Snorby, and Squil [127]

There are some cons of Suricata tool which needs to be addressed:

1. The usage of CPU is high
2. Update information is not always available
3. The support community is smaller than snort community
4. The installation process is complex

### C. BRO (ZEEK)

Bro is an IDS tool which identifies anomalies in the network traffic. It also supports signature analysis. The bro consists of three components: libpcap library, event engine, and policy script interpreter [127]. Libpcap library captures packets from network interfaces. Event engine keeps track of events including TCP connection or HTTP requests. It also runs integrity checks to ensure the whether packets are well formed or not. Policy script interpreter uses its own scripts to specify the rules which detect the anomalies in the network. Bro can be installed only on some operating systems including Unix, Linux, and Mac OS.

There are some advantages of using Bro IDS, these advantages can be addressed as follows:

1. It can perform application level inspection
2. It can perform tunnel analysis
3. It supports forensic analysis
4. It can capture packets on various protocols including SNMP, FTP, DNS, and HTTP.

On the other hand, there are a few drawbacks of Bro as the following:

1. It does not support all operating systems
2. It is difficult to use for who does not have domain knowledge in the area
3. There are some challenges on usability, GUIs, and installation.

### D. OSSEC

OSSEC (Open Source Security) IDS is a HIDS (host-based intrusion detection system) which supports cross-platform. It is an open source IDS tool which provides a powerful analysis engine, rootkit detection, Windows registry checking, and real time alert and response [125]. In addition, the OSSEC also provides checklists which frequently validate the significant files from time to time. This leads to alerting the administrator when suspicious events occur. The OSSEC IDS tools has several advantages including free and open source, support cross-platform, can detect changes on files and registries, can detect rootkit, and real time alerting with active response. On the other hand, there are some problems that can be addressed as follows: problems on pre-sharing keys, supports only server-agent mode in Windows, and installation and managing requires technical knowledge on the field.

### E. OpenWIPS-NG

OpenWIPS-NG (open source Wireless IPS) is a IDS and IPS tool designed specifically for wireless networks [128]. It is an open-source tool which consists of three components: sensor, server, and interface. Each installation comprises 1 sensor which acts like a packet sniffer to capture wireless traffic. The captured wireless traffic is sent to the server for data analysis. On the server side, there is a detection engine which detects the intrusion patterns. The dashboard (interface) shows events and alerts for network administrators. After the packets are analyzed on the server, the necessary answers are given to the attacks. It detects Dos attacks, supports plugins, and assembles the packets. Furthermore, the OpenWIPS-NG tool runs as an intrusion detector as well as Wi-Fi packet sniffer. However, OpenWIPS-NG has some limitations as a NIDS and only requires 1 sensor for each installation.

### F. SOLARWINDS SECURITY EVENT MANAGER

SolarWinds Security Event Manager (SEM) is a commercial network security tool which can log messages generated by Windows, Mac-OS, Unix and Linux operating systems. It can be categorized as HIDS, but it manages data collected by Snort which is regarded as NIDS as well [129]. It not only recognizes suspicious activities, but also performs responses and recovery actions. In SolarWinds, for event correlation, the system configuration is done with a lot of rules [129].

There are some significant features of SolarWinds Event Manager which can be list as follows:

1. The licensing is simple and low-cost
2. The log collection is centralized with real-time monitoring
3. It supports behavior profiling
4. It supports data, application, and user monitoring
5. It provides file integrity monitoring
6. It supports log management with reporting
7. The threat detection and response is automated
8. The data collected by Snort can be managed
9. It supports intuitive dashboard with friendly user interface

The drawbacks of SolarWinds Security Event Manager can be addressed as follows:

1. The first setup takes plenty of time
2. The configuration of alerting can be confusing
3. The version updates takes time

### G. SECURITY ONION

Security Onion is an IDS tool as well as log management, and enterprise security monitoring for Linux distribution [130]. It integrates different components from front-end analysis tools such as ELSA, Kibana, NetworkMiner, Sguil, Snorby, and Xplico. Although Security Onion can categorize NIDS, it has some HIDS functionality, too.

The advantages of Security Onion IDS include supporting log management, integrating components from various front-end tools supporting NIDS and HIDS in the same time, and generating effective charts. On the other hand, the drawbacks of Security Onion are as follows: It uses complex methods for network monitoring, and learning usage of the tool is challenging.

### H. SAMHAIN

Samhain is a HIDS (host-based intrusion detection system) that supports open source. It logs the file, and port monitoring, provides file integrity checking, and detects rootkit [131]. Samhain is available as a standalone application on a single host as well as multiple hosts with various operating systems. Samhain provides central logging, storage with central updates, and web-based management console. To protect its integrity, Samhain uses several features including stealth mode, configuration files, steganography and PGP-signed database. Tamper resistance, centralized monitoring, complete integrity checking, port monitoring, and rootkit detection can be shown pros of Samhain HIDS. However, due to expensive integrity checkers and data analyzer, the Samhain uses too much processor power [132].

### I. FAIL2BAN

Fail2ban is an open source IDS/IPS software framework which mainly prevents or stops brute force attacks [133]. It monitors log files, prevents traffic from malicious IPs, limits the number of requests per seconds, and stops probing attempts. Fail2Ban runs multiple actions when a suspicious IP address is detected. It has filters for several services including Apache, asterisk, lighttpd, mysql, nginx, qmail, ssh, sshd, proftpd, and postfix. In Fail2ban, the filters are identified by regular expression in Python scripting language. A filter and action combination is called jail. The jail is used to distinguish malicious hosts, and block those hosts from accessing the designated network services.

### J. SAGAN

Sagan is a real time HIDS log analyzer which supports open source, and multi-thread architecture. It is written in C language and runs under the Unix operating system. Sagan rule structure is similar to Snort. The Sagan tool supports several output formats for analysis, reporting, event detection, and log normalization. It is compatible with data collected from Snort, Squil, Anaval, and Base. it can be installed on several OS including Unix, Linux, and Mac-OS. The cons of the Sagan is the difficult installation process and not a true IDS.

### K. AIDE

AIDE (Advanced Intrusion Detection Environment) is a HIDS, which supports cross platform including Unix, Linux, and MacOS, that checks the integrity of the files. The AIDE first creates a baseline database by taking the snapshot of the file contents, register hashes, permissions, modification

**TABLE 11. Comparison of well-known IDS tools.**

| Tool Name | IDS Type | Supported Platform | Main Features/Success |
|---|---|---|---|
| Snort | NIDS | Unix, Linux, MacOs, Windows, FreeBSD | It analyzes the captured traffic to detect attacks in real-time. It can detect various attacks including Dos, DDos, port scans, nmap scans, SBM probes, CGI attacks, NetBIOS queries. |
| Suricata | NIDS | Unix, Linux, MacOs, Windows, FreeBSD | It supports multithreading, higher-level protocols including SMB, HTTP, FTP, as well as lower-level protocols such as TLS, UDP, TCP, ICMP. |
| Bro | NIDS | Unix, Linux, MacOs | It uses signature analysis and identifies anomalies.It performs application level inspection, tunnel and forensic analysis, and supports various protocols: SNMP, FTP, DNS, and HTTP. |
| OSSEC | HIDS | Unix, Linux, MacOs, Windows | It provides a powerful analysis engine, rootkit detection, Windows registry checking, and real time alerting and response. |
| OpenWIPS-NG | NIDS | Linux | OpenWIPS-NG tool runs as an intrusion detector as well as Wi-Fi packet sniffer. It detects Dos attacks, supports plugins, and assembles the packets. |
| SolarWinds Security Manager | HIDS | Linux and Windows | It is a commercial network security tool which automated the threat detection and responses. It supports behavior profiling, application, and user monitoring, log management with reporting, and provides file integrity monitoring. |
| Security Onion | NIDS and HIDS | Linux distribution, MacOs | It integrates different components from front-end analysis tools such as ELSA, Kibana, NetworkMiner, Sguil, Snorby, and Xplico. The advantages of Security Onion IDS includes supporting log management, integrating components from various front-end tools supporting NIDS and HIDS at the same time. |
| Samhain | HIDS | Linux, all POSIX/UNIX Systems | It logs the file, and port monitoring, provides file integrity checking, and detects rootkit. Samhain provides central logging, storage with central updates, and web-based management console. |
| Fail2Ban | HIDS | Unix-like Systems | It monitors log files, prevents traffic from malicious IPs, limits the number of requests per seconds, and stops probing attempts. The jail is used to distinguish malicious hosts, and block those hosts from accessing the designated network services. |
| Sagan | HIDS log analyzer | Unix, Linux, MacOs | It is a real time log analyzer which supports open source, and multi-thread architecture. The Sagan tool supports several output formats for analysis, reporting, event detection, and log normalization. The drawback of the Sagan is the difficult installation process as well as not being a true IDS. |
| AIDE | HIDS | Unix, Linux, MacOS | File integrity checker which can detect changes on file contents, register hashes, permissions, modification times, etc. Although the AIDE can capture several changes on the system, it cannot capture the rootkits effectively which run in the kernel mode. |

times, etc. Then, the real status of the system is compared with the previously built database. If there is a change, the change can be detected. Although the AIDE can capture many changes on the system, it cannot capture the rootkits effectively which run in the kernel mode. AIDE is not user friendly and lack of many features which is found in other IDSs, can be addressed as a cons of AIDE.

## L. EVALUATION OF IDS TOOLS

In this subsection, IDS tools are investigated based on working principle, detection intrusions strategy, supported platforms, advantages, and disadvantages. The comparison of current IDS tools is summarized in Table 11. As it can be seen from Table 11, the IDS types can be HIDS and NIDS. Some of the IDS tools support multiple-platforms while others support single platform. The Snort IDS is one of the first tools, which supports multiple platforms, and has many rules to detect intrusions based on signatures and anomalies. However, Snort cannot effectively detect fragmented packets, and it is difficult to monitor the packets in the large networks. Suricata IDS tool supports multiple operating system environments as well as supports higher-level and lower-level protocols. On the other hand, for Suricata the usage of CPU is high, the installation process is complex, and the support community is smaller than the snort community.

Bro uses signature analysis as well as detects anomalies in the network. It performs application level inspection, forensic analysis, and supports various protocols including SNMP, FTP, DNS, and HTTP. However, there are some challenges on usability, GUIs, and installation on the Bro IDS tool (Table 11). OSSEC supports cross-platforms, provides a powerful analysis engine, rootkit detection, Windows registry checking, and real time alerting and response. However, OSSEC supports only server-agent mode in Windows, and installation and managing of OSSEC IDS tools requires domain knowledge on the field. OpenWIPS-NG is a IDS and IPS tool designed specifically for wireless networks. It Works on Linux operating system and performs as an intrusion detector as well as Wi-Fi packet sniffer. It detects DoS attacks, supports plugins, and assembles the packets. On the other hand, OpenWIPS-NG requires 1 sensor for each installation, and has some limitations as a NIDS. SolarWinds Security Event Manager is a commercial network security tool that can log messages generated by various operating systems. It can be categorized as HIDS, but it manages data collected by Snort which is regarded as NIDS as well. It supports behavior profiling, application, and user monitoring, log management with reporting, and provides file integrity monitoring.

Security Onion is an IDS tool as well as log management, and enterprise security monitoring for Linux distribution. It integrates different components from front-end analysis. Although Security Onion can categorize NIDS, it has several HIDS functionality. On the other hand, the drawbacks of Security Onion is that it uses complex method for network monitoring, and the learning curve of the tool is challenging. Samhain logs the file, monitors the ports, provides file integrity checking, and detects rootkit. Samhain provides central logging, storage with central updates, and web-based management console. To protect its integrity, Samhain applies several features including stealth mode, configuration files, and steganography (Table 9). However, due to expensive integrity checkers, the Samhain uses too much processor power. Fail2ban is a IDS/IPS software framework which monitors log files, prevents traffic from malicious IPs, limits the number of requests per seconds, and stops probing attempts. A filter and action combination is called jail in the Fail2ban IDS tool. The jail is used to detect malicious hosts, and block those hosts from accessing the specified network services.

Overall, it can be said that for different situations and platforms one IDS tool can perform better than another. This is because companies' needs vary and change overtime. The bandwidth of the networks, the scalability of IDS tools, the performance of IDS, the size of the company, and the complexity of the victim system are other concerns which need to be taken into account when the best suitable IDS is chosen for the target system.

## VIII. GENERAL EVALUATION OF INTRUSION DETECTION SYSTEMS

This extensive review paper is different from the previous survey papers in many aspects. Previous works are mostly focused on only one or two subjects such as intrusion detection methodologies or datasets which have been used. On the contrary, in this study, the various aspects of the current IDSs are examined. In addition, several suggestions are being made for each subject. The paper also makes contributions not only for researchers but also private companies which want to utilize IDSs more effectively.

In the paper, all technologies, methodologies and approaches of IDSs were examined in detail. Each method used has superior aspects to each other in many respects such as data collection method, speed and accuracy of detecting attacks, and alarming method. The advantages and disadvantages of the intrusion detection systems are given in Table 12. Despite many IDSs developed and under development, existing systems still appear to be prone to false alarms or false positives.

These IDSs need to be properly configured to distinguish normal traffic on their network from potentially malicious activity. However, despite the inefficiencies they cause, false positives generally do not cause serious damage to the network. The much more serious IDS error is a false negative. This is because IDS fails to detect a threat and mixes it with normal traffic. In a false-negative scenario, there is no indication that the attack has occurred, no alarm is generated, and attacks are often defined after the network has been affected in some way. In summary, it is better for IDS to be hypersensitive to abnormal behavior and produce false positives than to be insensitive by producing false negatives.

As attacks evolve and become more complex, false negatives in IDSs are becoming a bigger problem. Signature-based IDSs can detect known attacks quickly and with high accuracy. However, when it comes to unknown attacks, even anomaly-based detection systems, which include approaches that can detect these attacks, are still insufficient. As a result, there is a growing need for IDSs to detect new behaviors, proactively identify new threats and avoidance techniques as soon as possible. In this context, some recommendations are listed below:

- Next generation attacks use some techniques to hide themselves. In order to detect these attacks both with high accuracy and quickly, a hybrid system that includes a combination of signature-based and anomaly-based approaches can be developed.
- Observing the network in real time and detecting attacks is a difficult process. Most of the studies done so far have been detection studies using ready datasets and are not suitable for real-time monitoring. Developing a system that can detect real-time attacks will make a great contribution to this field.
- Most intrusion detection systems are prone to FPs and FNs. In addition to obtaining a high accuracy rate in new studies, studies should be carried out to reduce FP and FNs.
- There is still a lack of a well-known, high-volume, and most importantly up-to-date dataset that can be used to evaluate the performance of intrusion detection systems. The development of a new dataset containing novel attack types that will fill the gap in this area will also be an important step in this area.

## IX. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

Cyber-related attacks are increasing exponentially, and there is no well-known method to stop all these attacks. IDS is one of the most important approaches to decrease or stop the cyber-attacks. Besides, attackers are using the most recent tools and technologies to evade the IDSs, firewalls, and antivirus systems. It can be said that a well-organized zero-day attack will remain undetectable in the computer-based system. To increase the detection of new and complicated cyber-attacks, the weak points of the current IDSs need

**TABLE 12.** Advantages and disadvantages of intrusion detection systems.

| | | | Pros | Cons |
|---|---|---|---|---|
| INTRUSION DETECTION SYSTEMS | IDS Technologies | **Network Based** | - Extensive detection<br>- It is not easily detected by attackers | - False positive and false negative<br>- Require a significant amount of tuning and customization<br>-It cannot detect attacks on encrypted network traffic |
| | | **Host Based** | - Detects local events.<br>- Can view encrypted traffic<br>- It can detect inconsistencies in applications and programs by investigating logs | - Difficult to manage<br>- Vulnerable to attacks against host operating system<br>- Potential targets for a DoS attack<br>- It can use a lot of disk space |
| | | **Wireless IDS** | - It can detect WLAN protocol level attacks, misconfigurations<br>- It has high intrusion detection accuracy | - Susceptible to denial-of-service attacks and physical attacks<br>- It cannot detect attack types that are passive and involve offline processing of traffic |
| | | **Network Behavioral Analysis** | - Giving high accuracy in detecting attacks that generate large amounts of traffic<br>- It has the potential of discovering unknown threats | - It is not very effective at detecting small-scale attacks<br>- Detects attacks in a long time<br>- It is prone to false positives |
| | IDS Methodologies | **Signatured-Based** | - It is easy to understand and quick.<br>- It is very effective at detecting known attacks<br>- Low false positive rate | - It is not successful in detecting unknown attacks<br>- Signatures must be defined for all attacks<br>- Signature database must be kept up-to-date |
| | | **Anomaly-Based** | - No prior knowledge required<br>- Detects unknown attack<br>- Does not require constantly following attack techniques | - System-generated alarms are difficult to manage<br>- It is prone to high false alarm rates<br>- The defined rules and profiles must be kept up to date |
| | | **Stateful Protocol Analysis** | - It can detect an unexpected sequence of commands<br>- It can follow the profiles in both the network layer and the application layers | - Analysis process is complex<br>- Resource usage is high<br>- Inability to detect types of attacks that do not violate the characteristics of protocol behavior |
| | IDS Approaches | **Statistic Based** | - Detects unknown attack<br>- It does not need updating every time<br>- It can detect a wide range of attack types | - Building a normal profile takes time<br>- It can generate many false positives |
| | | **Rule Based** | - Ease of interpretation (as long as there aren't too many rules)<br>- It is successful for data with categorical features | - Not the best performers in terms of prediction quality<br>- Discretization of data or setting thresholds for rules is difficult. |
| | | **Heuristic Based** | - It can detect unknown attacks<br>- It can use both static and dynamic features | - Vulnerable to metamorphic attacks<br>- The training phase is difficult |
| | | **Pattern Based** | - Efficiently detects known attacks<br>- It is quick and easy to apply | - It cannot detect most of the unknown attacks. |
| | | **Cloud Based** | - Lower costs<br>- More computing power<br>- Short analysis time | - Overhead between client and server<br>- lack of real-time monitoring |
| | | **Machine Learning Based** | - High performance<br>- Easily adaptable and supports flexibility<br>- It can detect unknown attacks | - Prone to high bias.<br>- Cannot handle outliers.<br>- Data preprocessing is difficult |
| | | **Deep Learning Based** | - It can recognize large-scale and multidimensional data<br>- It can handle dynamic data that changes over time | - Not resistant to evasion attacks<br>- Additional time for model building |

to be eliminated and current IDSs must be integrated with new technologies such as cloud, machine learning, and deep learning. In this paper, first intrusion detection systems are summarized based on the techniques that are used, different detection methods, and the main idea of each detection approach. Then, available datasets, advantages, and disadvantages of each detection technology, and current state-of-the-art studies are analyzed. Finally, comparison of the detection techniques and the future of the research directions, and our thoughts about IDSs are given.

Network-based IDSs are effective to detect intrusions on the computer network. The integration of various detection methods increases the detection rate as well as attack types. NIDSs face difficulties to detect attacks on encrypted network traffic. On the other hand, host-based IDSs detect attacks on the host. Generally, host-based IDSs use more than one detection technique to increase the detection rate. Wireless IDSs provide high detection capabilities. However, they cannot detect passive monitoring and offline processing attacks in the wireless traffic. Signature-based detection techniques are fast and effective to detect known attacks, but it fails to detect unknown attacks. Anomaly-based IDS generates an alarm when it catches an activity which deviates from the normal attack patterns. The anomaly-based IDS can detect unknown attacks as well as types, but it raises false alarms. We concluded that each detection approach has its own advantages and disadvantages, and performs better on different datasets. Various features can be used to evaluate the performances of IDS approaches including the size of the data, dimensionality of the data, number of available features, the distribution of the data. It can be said that statistical-, heuristic-, and pattern based approaches are used sufficiently in the current IDS. Thus, researchers need to focus more on cloud-, machine learning-, and deep learning- based approaches. When building an IDS, researchers and developers need to be aware of various evasion techniques such as address spoofing, avoiding defaults, pattern change evasion, coordinated, low-bandwidth attacks, and fragmentation.

In addition, the well-known IDS datasets are analyzed. Each dataset has its own pros and cons, and works better for different situations. The KDD '99 dataset is the biggest and most used dataset for IDSs, but this dataset has many redundant features which makes the ML classification process challenging. The NSL-KDD dataset is a modification of KDD. It is quite effective to test modern IDSs on NSL-KDD dataset, but this dataset lacks modern network attacks. Other datasets such as CAIDA, ADFA-LD and ADFA-WD, AWID, UNSW-NB15, and CICIDS have different deficiencies. These datasets are popular for network intrusion detection systems and used in several scientific studies. Since the network attacks are evolving overtime, IDS datasets and features must be updated from time to time to evaluate the future network intrusions accuracy. Paper also discussed the current IDS tools. For different cases and platforms one IDS tool can perform better than another. This is because companies' needs vary and change overtime. The bandwidth of the networks, the performance of IDS, the scalability of IDS tools, the size of the company, and the complexity of the victim system are other concerns which need to be taken into account when the best suitable IDS is chosen for the target system.

## REFERENCES

[1] T. Shon and J. Moon, "A hybrid machine learning approach to network anomaly detection," *Inf. Sci.*, vol. 177, no. 18, pp. 3799–3821, Sep. 2007.

[2] K. Ethala, R. Seshadri, N. G. Renganathan, and M. S. Saravanan, "A role of intrusion detection system for wireless LAN using various schemes and related issues," *Amer. J. Appl. Sci.*, vol. 10, no. 9, p. 979, 2013.

[3] E. Biermann, E. Cloete, and L. M. Venter, "A comparison of intrusion detection systems," *Comput. Secur.*, vol. 20, no. 8, pp. 676–683, Dec. 2001.

[4] Y. Bai and H. Kobayashi, "Intrusion detection systems: Technology and development," in *Proc. 17th Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, Mar. 2003, pp. 710–715.

[5] A. Lazarevic, V. Kumar, and J. Srivastava, "Intrusion detection: A survey," in *Managing Cyber Threats*. Boston, MA, USA: Springer, 2005, pp. 19–78.

[6] M. Garuba, C. Liu, and D. Fraites, "Intrusion techniques: Comparative study of network intrusion detection systems," in *Proc. 5th Int. Conf. Inf. Technol., New Generat. (ITNG)*, Apr. 2008, pp. 592–598.

[7] V. V. R. P. V. Jyothsna, R. Prasad, and K. M. Prasad, "A review of anomaly based intrusion detection systems," *Int. J. Comput. Appl.*," vol. 28, no. 7, pp. 26–35, 2011.

[8] N. A. Alrajeh, S. Khan, and B. Shams, "Intrusion detection systems in wireless sensor networks: A review," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 5, May 2013, Art. no. 167575.

[9] A. Milenkoski, M. Vieira, S. Kounev, A. Avritzer, and B. D. Payne, "Evaluating computer intrusion detection systems: A survey of common practices," *ACM Comput. Surv.*, vol. 48, no. 1, pp. 1–41, 2015.

[10] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1–22, Dec. 2019.

[11] F. Sabahi and A. Movaghar, "Intrusion detection: A survey," in *Proc. 3rd Int. Conf. Syst. Netw. Commun.*, Oct. 2008, pp. 23–26.

[12] K. Scarfone and P. Mell, *Guide to Intrusion Detection and Prevention Systems (IDPS)*, Standard NIST SP 800-90, 2007.

[13] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Comput. Secur.*, vol. 28, nos. 1–2, pp. 18–28, Feb. 2009.

[14] C. Kruegel, F. Valeur, and G. Vigna, *Intrusion Detection and Correlation: Challenges and Solutions*, vol. 14. Springer, 2004.

[15] S. Han, M. Xie, H. H. Chen, and Y. Ling, "Intrusion detection in cyber-physical systems: Techniques and challenges," *IEEE Syst. J.*, vol. 8, no. 4, pp. 1052–1062, Dec. 2014.

[16] R. J. Santos, J. Bernardino, and M. Vieira, "Approaches and challenges in database intrusion detection," *ACM SIGMOD Rec.*, vol. 43, no. 3, pp. 36–47, Dec. 2014.

[17] A. M. Aleesa, B. B. Zaidan, A. A. Zaidan, and N. M. Sahar, "Review of intrusion detection systems based on deep learning techniques: Coherent taxonomy, challenges, motivations, recommendations, substantial analysis and future directions," *Neural Comput. Appl.*, vol. 32, no. 14, pp. 9827–9858, Jul. 2020.

[18] M. Ozkan-Okay and R. Samet, "Hybrid intrusion detection approach for wireless local area network," in *Proc. 7th Int. Conf. Control Optim. Ind. Appl. (COIA)*, 2020, pp. 311–313.

[19] P. Hadem, D. K. Saikia, and S. Moulik, "An SDN-based intrusion detection system using SVM with selective logging for IP traceback," *Comput. Netw.*, vol. 191, May 2021, Art. no. 108015.

[20] G. Vigna and R. A. Kemmerer, "NetSTAT: A network-based intrusion detection system," *J. Comput. Secur.*, vol. 7, no. 1, pp. 37–71, Jan. 1999.

[21] S. More, M. L. Mathews, A. Joshi, and T. Finin, "A semantic approach to situational awareness for intrusion detection," in *Proc. Nat. Symp. Moving Target Res. (MTR)*. Baltimore, MD, USA: UMBC, 2012.

[22] N. Wattanapongsakorn, S. Srakaew, E. Wonghirunsombat, C. Sribavonmongkol, T. Junhom, P. Jongsubsook, and C. Charnsripinyo, "A practical network-based intrusion detection and prevention system," in *Proc. IEEE 11th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Jun. 2012, pp. 209–214.

[23] J. P. Amaral, L. M. Oliveira, J. J. P. C. Rodrigues, G. Han, and L. Shu, "Policy and network-based intrusion detection system for IPv6-enabled wireless sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 1796–1801.

[24] S. Kumar, A. Viinikainen, and T. Hamalainen, "Machine learning classification model for network based intrusion detection system," in *Proc. 11th Int. Conf. Internet Technol. Secured Trans. (ICITST)*, Dec. 2016, pp. 242–249.

[25] Q. Qassim, A. M. Zin, and M. J. A. Aziz, "Anomalies classification approach for network-based intrusion detection system," *Int. J. Netw. Secur.*, vol. 18, no. 6, pp. 1159–1172, 2016.

[26] G. Karatas and O. K. Sahingoz, "Neural network based intrusion detection systems with different training functions," in *Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS)*, Mar. 2018, pp. 1–6.

[27] A.-U.-H. Qureshi, H. Larijani, J. Ahmad, and N. Mtetwa, "A novel random neural network based approach for intrusion detection systems," in *Proc. 10th Comput. Sci. Electron. Eng. (CEEC)*, Sep. 2018, pp. 50–55.

[28] M. Mazini, B. Shirazi, and I. Mahdavi, "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms," *J. King Saud Univ., Comput. Inf. Sci.*, vol. 31, no. 4, pp. 541–553, Oct. 2019.

[29] S. Meftah, T. Rachidi, and N. Assem, "Network based intrusion detection using the UNSW-NB15 dataset," *Int. J. Comput. Digit. Syst.*, vol. 8, no. 5, pp. 478–487, 2019.

[30] P. Devan and N. Khare, "An efficient XGBoost–DNN-based classification model for network intrusion detection system," *Neural Comput. Appl.*, vol. 32, pp. 12499–12514, Jan. 2020.

[31] P. Bedi, N. Gupta, and V. Jindal, "I-SiamIDS: An improved siam-IDS for handling class imbalance in network-based intrusion detection systems," *Int. J. Speech Technol.*, vol. 51, no. 2, pp. 1133–1151, Feb. 2021.

[32] S. Gupta and R. Mamtora, "Intrusion detection system using wireshark," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 2, no. 11, pp. 358–363, 2012.

[33] P. Deshpande, S. C. Sharma, S. K. Peddoju, and S. Junaid, "HIDS: A host based intrusion detection system for cloud computing environment," *Int. J. Syst. Assurance Eng. Manage.*, vol. 9, no. 3, pp. 567–576, Jun. 2018.

[34] Y.-J. Ou, Y. Lin, Y. Zhang, and Y.-J. Ou, "The design and implementation of host-based intrusion detection system," in *Proc. 3rd Int. Symp. Intell. Inf. Technol. Secur. Informat.*, Apr. 2010, pp. 595–598.

[35] G. Creech and J. Hu, "A semantic approach to host-based intrusion detection systems using contiguousand discontiguous system call patterns," *IEEE Trans. Comput.*, vol. 63, no. 4, pp. 807–819, Apr. 2014.

[36] F. Lydia Catherine, R. Pathak, and V. Vaidehi, "Efficient host based intrusion detection system using partial decision tree and correlation feature selection algorithm," in *Proc. Int. Conf. Recent Trends Inf. Technol.*, Apr. 2014, pp. 1–6.

[37] B. Subba, S. Biswas, and S. Karmakar, "Host based intrusion detection system using frequency analysis of n-gram terms," in *Proc. IEEE Region 10 Conf. (TENCON)*, Nov. 2017, pp. 2006–2011.

[38] A. Chawla, B. Lee, S. Fallon, and P. Jacob, "Host based intrusion detection system with combined CNN/RNN model," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Cham, Switzerland: Springer, Sep. 2018, pp. 149–158.

[39] J. Byrnes, T. Hoang, N. N. Mehta, and Y. Cheng, "A modern implementation of system call sequence based host-based intrusion detection systems," in *Proc. 2nd IEEE Int. Conf. Trust, Privacy Secur. Intell. Syst. Appl. (TPS-ISA)*, Oct. 2020, pp. 218–225.

[40] R. Gassais, N. Ezzati-Jivan, J. M. Fernandez, D. Aloise, and M. R. Dagenais, "Multi-level host-based intrusion detection system for Internet of Things," *J. Cloud Comput.*, vol. 9, no. 1, pp. 1–16, Dec. 2020.

[41] M. Liu, Z. Xue, X. He, and J. Chen, "SCADS: A scalable approach using spark in cloud for host-based intrusion detection system with system calls," 2021, *arXiv:2109.11821*.

[42] D. Park, S. Kim, H. Kwon, D. Shin, and D. Shin, "Host-based intrusion detection model using Siamese network," *IEEE Access*, vol. 9, pp. 76614–76623, 2021.

[43] Y.-X. Lim, T. S. Yer, J. Levine, and H. L. Owen, "Wireless intrusion detection and response," in *Proc. IEEE Syst., Man Cybern. Soc. Inf. Assurance Workshop*, Jun. 2003, pp. 68–75.

[44] S. Boob and P. Jadhav, "Wireless intrusion detection system," *Int. J. Comput. Appl.*, vol. 5, no. 8, pp. 9–13, 2010.

[45] Y. Meng and W. Li, "Evaluation of detecting malicious nodes using Bayesian model in wireless intrusion detection," in *Proc. Int. Conf. Netw. Syst. Secur.* Berlin, Germany: Springer, Jun. 2013, pp. 40–53.

[46] Z. Afzal, J. Rossebø, B. Talha, and M. Chowdhury, "A wireless intrusion detection system for 802.11 networks," in *Proc. Int. Conf. Wireless Commun., Signal Process. Netw. (WiSPNET)*, Mar. 2016, pp. 828–834.

[47] C. Kolias, V. Kolias, and G. Kambourakis, "TermID: A distributed swarm intelligence-based approach for wireless intrusion detection," *Int. J. Inf. Secur.*, vol. 16, no. 4, pp. 401–416, 2017.

[48] R. Abdulhammed, M. Faezipour, A. Abuzneid, and A. Alessa, "Effective features selection and machine learning classifiers for improved wireless intrusion detection," in *Proc. Int. Symp. Netw., Comput. Commun. (ISNCC)*, Jun. 2018, pp. 1–6.

[49] D. S. Vijayakumar and S. Ganapathy, "Machine learning approach to combat false alarms in wireless intrusion detection system," *Comput. Inf. Sci.*, vol. 11, no. 3, pp. 67–81, 2018.

[50] S. M. Kasongo and Y. Sun, "A deep learning method with filter based feature engineering for wireless intrusion detection system," *IEEE Access*, vol. 7, pp. 38597–38607, 2019.

[51] S. M. Kasongo and Y. Sun, "A deep learning method with wrapper based feature extraction for wireless intrusion detection system," *Comput. Secur.*, vol. 92, May 2020, Art. no. 101752.

[52] P. Satam and S. Hariri, "WIDS: An anomaly based intrusion detection system for Wi-Fi (IEEE 802.11) protocol," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 1, pp. 1077–1091, Mar. 2021.

[53] B. Riyaz and S. Ganapathy, "A deep learning approach for effective intrusion detection in wireless networks using CNN," *Soft Comput.*, vol. 24, no. 22, pp. 17265–17278, Nov. 2020.

[54] N. B. Singh, M. M. Singh, A. Sarkar, and J. K. Mandal, "A novel wide & deep transfer learning stacked GRU framework for network intrusion detection," *J. Inf. Secur. Appl.*, vol. 61, Sep. 2021, Art. no. 102899.

[55] D. Zhao, I. Traore, B. Sayed, W. Lu, S. Saad, A. Ghorbani, and D. Garant, "BotNet detection based on traffic behavior analysis and flow intervals," *Comput. Secur.*, vol. 39, pp. 2–16, Nov. 2013.

[56] Y. H. Yu, F. U. Yu, and X. P. Wu, "Unknown attack detection model based on network behavior analysis," *Chin. J. Netword Inf. Secur.*, vol. 2, no. 6, p. 54, 2016.

[57] A. Youssef and A. Emam, "Network intrusion detection using data mining and network behaviour analysis," *Int. J. Comput. Sci. Inf. Technol.*, vol. 3, no. 6, pp. 87–98, Dec. 2011.

[58] T. Nitin, S. R. Singh, and P. G. Singh, "Intrusion detection and prevention system (IDPS) technology-network behavior analysis system (NBAS)," *ISCA J. Eng. Sci*, vol. 1, no. 1, pp. 51–56, 2012.

[59] A. Srivastav, P. Kumar, and R. Goel, "Evaluation of network intrusion detection system using PCA and NBA," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 2, no. 11, pp. 1–9, 2013.

[60] D. Moon, H. Im, I. Kim, and J. H. Park, "DTB-IDS: An intrusion detection system based on decision tree using behavior analysis for preventing APT attacks," *J. Supercomput.*, vol. 73, no. 7, pp. 2881–2895, 2017.

[61] K. K. Ghanshala, P. Mishra, R. C. Joshi, and S. Sharma, "BNID: A behavior-based network intrusion detection at network-layer in cloud environment," in *Proc. 1st Int. Conf. Secure Cyber Comput. Commun. (ICSCCC)*, Dec. 2018, pp. 100–105.

[62] J. Pacheco, V. Benitez, and L. Félix, "Anomaly behavior analysis for IoT network nodes," in *Proc. 3rd Int. Conf. Future Netw. Distrib. Syst.*, Jul. 2019, pp. 1–6.

[63] S. Ahn, H. Yi, Y. Lee, W. R. Ha, G. Kim, and Y. Paek, "Hawkware: Network intrusion detection based on behavior analysis with ANNs on an IoT device," in *Proc. 57th ACM/IEEE Design Autom. Conf. (DAC)*, Jul. 2020, pp. 1–6.

[64] T. Fladby, H. Haugerud, S. Nichele, K. Begnum, and A. Yazidi, "Evading a machine learning-based intrusion detection system through adversarial perturbations," in *Proc. Int. Conf. Res. Adapt. Convergent Syst.*, Oct. 2020, pp. 161–166.

[65] M. A. Khan, "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system," *Processes*, vol. 9, no. 5, p. 834, May 2021.

[66] S. Yang, "Research on network malicious behavior analysis based on deep learning," in *Proc. IEEE 5th Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Mar. 2021, pp. 2609–2612.

[67] J. Farshchi. (2003). *Wireless Intrusion Detection Systems*. [Online]. Available: http://www.securityfocus.com/infocus/1742

[68] V. Kumar and O. P. Sangwan, "Signature based intrusion detection system using SNORT," *Int. J. Comput. Appl. Inf. Technol.*, vol. 1, no. 3, pp. 35–41, Nov. 2012.

[69] M. Uddin, A. A. Rahman, N. Uddin, J. Memon, R. A. Alsaqour, and S. Kazi, "Signature-based multi-layer distributed intrusion detection system using mobile agents," *Int. J. Netw. Secur.*, vol. 15, no. 2, pp. 97–105, 2013.

[70] N. Hubballi and V. Suryanarayanan, "False alarm minimization techniques in signature-based intrusion detection systems: A survey," *Comput. Commun.*, vol. 49, pp. 1–17, Aug. 2014.

[71] K. Rai, M. S. Devi, and A. Guleria, "Decision tree based algorithm for intrusion detection," *Int. J. Adv. Netw. Appl.*, vol. 7, no. 4, p. 2828, 2016.

[72] M. Aldwairi, A. M. Abu-Dalo, and M. Jarrah, "Pattern matching of signature-based IDS using Myers algorithm under MapReduce framework," *EURASIP J. Inf. Secur.*, vol. 2017, no. 1, pp. 1–11, 2017.

[73] M. Baykara and R. Das, "A novel honeypot based security approach for real-time intrusion detection and prevention systems," *J. Inf. Secur. Appl.*, vol. 41, pp. 103–116, Aug. 2018.

[74] M. Baykara and R. Daş, "SoftSwitch: A centralized honeypot-based security approach usingsoftware-defined switching for secure management of VLAN networks," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 27, no. 5, pp. 3309–3325, Sep. 2019.

[75] M. Z. Gunduz and R. Das, "Cyber-security on smart grid: Threats and potential solutions," *Comput. Netw.*, vol. 169, Mar. 2020, Art. no. 107094.

[76] Z. S. Malek, B. Trivedi, and A. Shah, "User behavior pattern -Signature based intrusion detection," in *Proc. 4th World Conf. Smart Trends Syst., Secur. Sustainability (WorldS4)*, Jul. 2020, pp. 549–552.

[77] Y. Otoum and A. Nayak, "AS-IDS: Anomaly and signature based IDS for the Internet of Things," *J. Netw. Syst. Manage.*, vol. 29, no. 3, pp. 1–26, Jul. 2021.

[78] R. Samrin and D. Vasumathi, "Review on anomaly based network intrusion detection system," in *Proc. Int. Conf. Electr., Electron., Commun., Comput., Optim. Techn. (ICEECCOT)*, Dec. 2017, pp. 141–147.

[79] F. Geramiraz, A. S. Memaripour, and M. Abbaspour, "Adaptive anomaly-based intrusion detection system using fuzzy controller," *Int. J. Netw. Secur.*, vol. 14, no. 6, pp. 352–361, 2012.

[80] W. Yassin, N. I. Udzir, Z. Muda, and M. N. Sulaiman, "Anomaly-based intrusion detection through $k$-means clustering and Naives Bayes classification," in *Proc. 4th Int. Conf. Comput. Informat. (ICOCI)*, vol. 49, Aug. 2013, pp. 298–303.

[81] D. Narsingyani and O. Kale, "Optimizing false positive in anomaly based intrusion detection using genetic algorithm," in *Proc. IEEE 3rd Int. Conf. MOOCs, Innov. Technol. Educ. (MITE)*, Oct. 2015, pp. 72–77.

[82] B. S. Harish and S. A. Kumar, "Anomaly based intrusion detection using modified fuzzy clustering," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 4, no. 6, pp. 54–59, 2017.

[83] S. Aljawarneh, M. Aldwairi, and M. B. Yassein, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *J. Comput. Sci.*, vol. 25, pp. 152–160, Mar. 2018.

[84] B. A. Tama, M. Comuzzi, and K. Rhee, "TSE-IDS: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system," *IEEE Access*, vol. 7, pp. 94497–94507, 2019.

[85] E. Viegas, A. Santin, A. Bessani, and N. Neves, "BigFlow: Real-time and reliable anomaly-based intrusion detection for high-speed networks," *Future Gener. Comput. Syst.*, vol. 93, pp. 473–485, Apr. 2019.

[86] S. Dwivedi, M. Vardhan, S. Tripathi, and A. K. Shukla, "Implementation of adaptive scheme in evolutionary technique for anomaly-based intrusion detection," *Evol. Intell.*, vol. 13, no. 1, pp. 103–117, Mar. 2020.

[87] M. Eskandari, Z. H. Janjua, M. Vecchio, and F. Antonelli, "Passban IDS: An intelligent anomaly-based intrusion detection system for IoT edge devices," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6882–6897, Aug. 2020.

[88] P. Kumar, G. P. Gupta, and R. Tripathi, "Design of anomaly-based intrusion detection system using fog computing for IoT network," *Autom. Control Comput. Sci.*, vol. 55, no. 2, pp. 137–147, Mar. 2021.

[89] D. Mudzingwa and R. Agrawal, "A study of methodologies used in intrusion detection and prevention systems (IDPS)," in *Proc. IEEE Southeastcon*, Mar. 2012, pp. 1–6.

[90] D. Seo, H. Lee, and E. Nuwere, "SIPAD: SIP-VoIP anomaly detection using a stateful rule tree," *Comput. Commun.*, vol. 36, no. 3, pp. 562–574, 2013.

[91] Y. Yang, K. McLaughlin, S. Sezer, Y. B. Yuan, and W. Huang, "Stateful intrusion detection for IEC 60870-5-104 SCADA security," in *Proc. IEEE PES Gen. Meeting Conf. Expo.*, Jul. 2014, pp. 1–5.

[92] B. Kang, K. McLaughlin, and S. Sezer, "Towards a stateful analysis framework for smart grid network intrusion detection," in *Proc. 4th Int. Symp. ICS SCADA Cyber Secur. Res.*, Aug. 2016, pp. 124–131.

[93] J. Boite, P.-A. Nardin, F. Rebecchi, M. Bouet, and V. Conan, "StateSec: Stateful monitoring for DDoS protection in software defined networks," in *Proc. IEEE Conf. Netw. Softw. (NetSoft)*, Jul. 2017, pp. 1–9.

[94] B. Lewis, M. Broadbent, and N. Race, "P4ID: P4 enhanced intrusion detection," in *Proc. IEEE Conf. Netw. Function Virtualization Softw. Defined Netw. (NFV-SDN)*, Nov. 2019, pp. 1–4.

[95] V. Sharma, I. You, K. Yim, R. Chen, and J. H. Cho, "BRIoT: Behavior rule specification-based misbehavior detection for IoT-embedded cyber-physical systems," *IEEE Access*, vol. 7, pp. 118556–118580, 2019.

[96] A. Rashid, M. J. Siddique, and S. M. Ahmed, "Machine and deep learning based comparative analysis using hybrid approaches for intrusion detection system," in *Proc. 3rd Int. Conf. Adv. Comput. Sci. (ICACS)*, Feb. 2020, pp. 1–9.

[97] O. Sbai and M. El Boukhari, "Data flooding intrusion detection system for MANETs using deep learning approach," in *Proc. 13th Int. Conf. Intell. Syst., Theories Appl.*, Sep. 2020, pp. 1–5.

[98] S. Choudhary and N. Kesswani, "A hybrid classification approach for intrusion detection in IoT network," *J. Sci. Ind. Res.*, vol. 80, no. 9, pp. 809–816, 2021.

[99] N. Ye, S. M. Emran, Q. Chen, and S. Vilbert, "Multivariate statistical analysis of audit trails for host-based intrusion detection," *IEEE Trans. Comput.*, vol. 51, no. 7, pp. 810–820, Jul. 2002.

[100] S.-J. Han and S.-B. Cho, "Detecting intrusion with rule-based integration of multiple models," *Comput. Secur.*, vol. 22, no. 7, pp. 613–623, Oct. 2003.

[101] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, "Intrusion detection system: A comprehensive review," *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 16–24, 2013.

[102] M. Aldwairi, A. M. Abu-Dalo, and M. Jarrah, "Pattern matching of signature-based IDS using Myers algorithm under mapreduce framework," *EURASIP J. Inf. Secur.*, vol. 2017, no. 1, pp. 1–11, Dec. 2017.

[103] O. Aslan and R. Samet, "A comprehensive review on malware detection approaches," *IEEE Access*, vol. 8, pp. 6249–6271, 2020.

[104] O. Aslan, M. Ozkan-Okay, and D. Gupta, "Intelligent behavior-based malware detection system on cloud computing environment," *IEEE Access*, vol. 9, pp. 83252–83271, 2021.

[105] D. Gupta, S. Bhatt, M. Gupta, O. Kayode, and A. S. Tosun, "Access control model for Google cloud IoT," in *Proc. IEEE 6th Int. Conf. Big Data Secur. on Cloud (BigDataSecurity), IEEE Int. Conf. High Perform. Smart Comput. (HPSC), IEEE Int. Conf. Intell. Data Secur. (IDS)*, May 2020, pp. 198–208.

[106] W. Yassin, N. I. Udzir, Z. Muda, A. Abdullah, and M. T. Abdullah, "A cloud-based intrusion detection service framework," in *Proc. Int. Conf. Cyber Secur., Cyber Warfare Digit. Forensic (CyberSec)*, Jun. 2012, pp. 213–218.

[107] C.-F. Tsai, Y.-F. Hsu, C.-Y. Lin, and W.-Y. Lin, "Intrusion detection by machine learning: A review," *Expert Syst. Appl.*, vol. 36, no. 10, pp. 11994–12000, 2009.

[108] D. P. Vinchurkar and A. Reshamwala, "A review of intrusion detection system using neural network and machine learning," *Int. J. Eng. Sci. Innov. Technol.*, vol. 1, no. 2, pp. 54–63, 2012.

[109] U. S. Musa, M. Chhabra, A. Ali, and M. Kaur, "Intrusion detection system using machine learning techniques: A review," in *Proc. Int. Conf. Smart Electron. Commun. (ICOSEC)*, Sep. 2020, pp. 149–155.

[110] O. Aslan and A. A. Yılmaz, "A new malware classification framework based on deep learning algorithms," *IEEE Access*, vol. 9, pp. 87936–87951, 2021.

[111] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," *Knowl.-Based Syst.*, vol. 189, Feb. 2020, Art. no. 105124.

[112] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," 2018, *arXiv:1802.09089*.

[113] G. Bovenzi, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescape, "A hierarchical hybrid intrusion detection approach in IoT scenarios," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2020, pp. 1–7.

[114] J. Verma, A. Bhandari, and G. Singh, "Review of existing data sets for network intrusion detection system," *Adv. Math., Sci. J.*, vol. 9, no. 6, pp. 3849–3854, Jul. 2020.

[115] A. Thakkar and R. Lohiya, "A review of the advancement in intrusion detection datasets," *Proc. Comput. Sci.*, vol. 167, pp. 636–645, Jan. 2020.

[116] A. Özgür and H. Erdem, "A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015," *PeerJ Preprints*, vol. 4, Apr. 2016, Art. no. e1954v1.

[117] P. Hick, E. Aben, K. Claffy, and J. Polterock. *The CAIDA 'DDoS Attack 2007' Dataset*. Accessed: Jun. 9, 2012. [Online]. Available: http://www.caida.org/data/passive/ddos-20070804dataset.xml

[118] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1–6.

[119] G. Creech, "Developing a high-accuracy cross platform host-based intrusion detection system capable of reliably detecting zero-day attacks," Ph.D. dissertation, School Eng. Inf. Technol., Univ. New South Wales, Canberra, ACT, Australia, 2014.

[120] U. S. K. P. M. Thanthrige, J. Samarabandu, and X. Wang, "Machine learning techniques for intrusion detection on public dataset," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2016, pp. 1–4.

[121] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, "Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 184–208, 1st Quart., 2016.

[122] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2015, pp. 1–6.

[123] A. R. Sonule, M. Kalla, A. Jain, and D. Chouhan, "UNSW-NB15 dataset and machine learning based intrusion detection systems," *Int. J. Eng. Adv. Technol.*, vol. 9, pp. 2638–2648, 2020.

[124] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. ICISSP*, Jan. 2018, pp. 108–116.

[125] D. A. Bhosale and V. M. Mane, "Comparative study and analysis of network intrusion detection tools," in *Proc. Int. Conf. Appl. Theor. Comput. Commun. Technol. (iCATccT)*, Oct. 2015, pp. 312–315.

[126] K. R. Karthikeyan and A. Indra, "Intrusion detection tools and techniques—A survey," *Int. J. Comput. Theory Eng.*, vol. 2, no. 6, p. 901, 2010.

[127] J. Timofte, "Intrusion detection using open source tools," *Inform. Econ. J.*, vol. 2, no. 46, pp. 75–79, 2008.

[128] *Top 10 Best Intrusion Detection Systems (IDS) [2021 Rankings]*. Accessed: Jun. 29, 2021. [Online]. Available: https://www.softwaretestinghelp.com/intrusion-detection-systems/

[129] A. M. Resmi and R. Manicka, "Intrusion detection system techniques and tools: A survey," *Scholars J. Eng. Technol.*, vol. 5, no. 3, pp. 122–130, 2017.

[130] Administrator Guide. *Security Event Manager, Version 2021.2*. Accessed: Jun. 30, 2021. [Online]. Available: https://documentation.solarwinds.com/en/success_center/sem/content/sem_administrator_guide.htm

[131] *Security Onion Documentation*. Accessed: Jun. 30, 2021. [Online]. Available: https://docs.securityonion.net/en/2.3/

[132] *The SAMHAIN File Integrity/Host-Based Intrusion Detection System*. Accessed: Jul. 1, 2021. [Online]. Available: https://www.la-samhna.de/samhain/index.html

[133] X. Wang, A. Kordas, L. Hu, M. Gaedke, and D. Smith, "Administrative evaluation of intrusion detection system," in *Proc. 2nd Annu. Conf. Res. Inf. Technol.*, Oct. 2013, pp. 47–52.

[134] M. Ford, C. Mallery, F. Palmasani, M. Rabb, R. Turner, L. Soles, and D. Snider, "A process to transfer Fail2ban data to an adaptive enterprise intrusion detection and prevention system," in *Proc. SoutheastCon*, Mar. 2016, pp. 1–4.

**MERVE OZKAN-OKAY** received the B.S. and M.S. degrees in computer engineering from Ankara University, in 2014 and 2016, respectively, where she is currently pursuing the Ph.D. degree with the Department of Computer Engineering. She is a Research Assistant with the Department of Computer Engineering, Ankara University. She has published several papers on international journals and conferences. Her current research interests include cyber security, cloud-based systems, machine learning, and image processing.

**REFIK SAMET** received the Ph.D. degree from the Department of Fault-Tolerant Multicomputer and Multiprocessor Systems, Institute of Control Sciences, Russian Academy of Sciences. He is currently a Professor and the Head of the Computer Engineering Department, Ankara University. He has published several papers on international journals and conferences. His research interests include parallel systems, cyber security, machine learning networks, and mobile applications.

**ÖMER ASLAN** received the B.Sc. degree from the Computer Engineering Department, University of Trakya, Turkey, in 2009, the M.Sc. degree in information security from The University of Texas at San Antonio, USA, in 2014, and the Ph.D. degree in cyber security from the University of Ankara, Turkey, in 2020. He is currently a Dr. Researcher with the Department of Computer Technologies, University of Bandırma 17 Eylül, Turkey. He is working on computer systems, information security, cyber security, malware analysis, cloud computing, and the IoT device security. He has published several papers on international journals and conferences. He has been also serving as a reviewer in some prestigious journals.

**DEEPTI GUPTA** received the M.S. degree in computer science from The University of Texas at San Antonio, where she is currently pursuing the Ph.D. degree in computer science. She has worked as an Adjunct Faculty with the Department of Computer Science, St. Edward University, Austin. She is the Faculty of the Department of Computer Science, Huston Tillotson University, Austin. Her primary area of research interests include security and privacy in cloud computing and the Internet of Things, security models, and deep learning. She has also served as a reviewer and a committee member in conferences.