Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 6 (2023)

Intrusions Detecting in Network Systems using Data Mining Techniques

Amin Heidari
Research scholar, Department of Computer Engineering, Tehran Branch, Islamic Azad University Science and Research Branch, Fars, Iran.

Abstract:- The intrusion detection system is one of the most important security parameters of modern computer networks, which will be able to detect intrusions in the network by examining the series of different events. The system is used independently in the network and can detect the occurrence of an attack based on two abnormal behavior detection techniques. Currently, most research has been conducted on infiltration detection systems in the field of unusual behavior-based detection using various techniques (statistical, artificial intelligence, data mining, machine learning). The study used an improved ant colony algorithm to detect Trojan malware. The study used a KDD dataset with an accuracy rate of 96.33.

Keywords*: intrusion detection, ant colony, data mining.*

1. Introduction

creating computer systems without any vulnerabilities and security breaches is practically impossible. The detection of intrusions in computer systems is of particular importance. Intrusion detection systems are hardware or software systems that monitor computer networks for malicious activities, policy violations, or security flaws and provide reports to the network management department [1-2]. Intrusion detection systems are responsible for identifying and detecting any unauthorized use of the system, misuse, or exploitation by both internal and external users [3]. The goal of these systems is not to prevent attacks but rather to detect and potentially identify attacks and security flaws in the system or computer network and report them to the system administrator. Intrusion detection systems are often used in conjunction with firewalls as a complementary security measure [4-5]. Traditional intrusion detection systems cannot adapt to new attacks, which is why data mining-based intrusion detection systems have become popular. Identifying patterns in large volumes of data greatly assists in this task. Data mining methods, by labeling data as normal or abnormal and identifying features and characteristics with classification algorithms, can detect abnormal data. This has increased the accuracy and effectiveness of intrusion detection systems, ultimately enhancing network security [6-7]. Intrusions can be motivated by various reasons such as political, financial, military, or simply to demonstrate skill by exploiting weaknesses in application programs, software bugs, and flaws in protocol and operating system design. During the identification phase, vulnerabilities are discovered, and an intrusion scenario is developed by the attacker. In order to counter intrusions into computer systems and networks, various methods have been developed, collectively known as intrusion detection methods. The goal of intrusion detection is to identify unauthorized use, misuse, and damage to computer systems and networks by both internal users and external attackers. Intrusion detection systems are considered one of the key components of security infrastructure in many organizations. These systems consist of hardware and software models and patterns that automate processes involved in monitoring events occurring in the network or computer systems. Intrusion detection systems analyze these events to solve security problems in computer systems and networks [8-9].

Penetration detection systems attempt to identify user activities in both normal and anomalous ways by comparing web connection transactions based on known penetration patterns designed by professionals and experts. Traditional methods cannot efficiently explore unknown patterns of penetration because manpower encounters networks of computer systems with high speed and complexity while performing penetration detection analysis. Therefore, intelligent decision-making techniques and technologies based on data mining are used in this regard. To identify patterns or patterns that are effective and efficient in detecting penetration[10-11]. In this section, we will examine the work done. most topics include classification methods with supervision, unsupervised and

association rules. Comparisons between monitoring methods in implementation as well as identifying abnormal data using clustering methods and using Association rules to detect fraud are the main axes under consideration. In [12] a penetration detection model is provided using a combination of feature selection Chi-Square and multi-class SVM.And many intrusion detection systems use only one classification algorithm to categorize network traffic as normal and abnormal. Due to the large amount of data, this model of classifications manages to achieve high attack detection rates and decrease false alarm rates. However, by reducing the dimensions of the data they can achieve an optimal set of features without losing information and then classify the identification of different network attacks using multi-class modeling methods.

In reference [15] feature analysis, evaluation and comparison of classification algorithms based on infiltration dataset has noise. Real-world network data traffic is associated with a large amount of noise-infected information, and IDS often works in such an environment. One of the most challenging issues in IDS is dealing with a noisy data environment to detect attacks from network activities. The study evaluated and compared different data mining and machine learning algorithms with NSL-KDD and KDD99 datasets with 10% and 20% noise. Experimental results of this study show that the NN(SOM) algorithm is much better in terms of noise environment resistance compared to other algorithms studied. However JRip and J48 from the tree algorithm family performed better than other algorithms. In reference [16] an intrusion detection system is put forward to effectively detect attacks. To this end, a new feature selection algorithm called the information ratio-based optimal feature selection algorithm is provided. This feature selection algorithm selects only a number of desirable and important features from the KDD Cup dataset. In addition, the backup vector machine classification and rule-based classification have been used to influence data classification and achieve greater accuracy. In reference [17] one of the important challenges in penetration detection research is the design of an accurate penetration detection system in terms of high detection rates, high accuracy and low false alarm rates. In this paper, a general structure of a hybrid learning approach is presented. The proposed method is then implemented using K-means Clustering and multiple classifications. The data is segmented by a method based on the K-means clustering algorithm. Then each part is divided, using a distinct classification. The gasoline grid, backup vector machine, and Uber classification algorithms have been used as classifications. The combined method presented has better results than single-classified in terms of detection speed, accuracy and false alarm rate. The combined method diagnosis rate offered is 50/99 percent. In reference [18] a new method called G-LDA is presented, which combined the integration of the diricle latent allocation and the genetic algorithm with the aim of identifying anomalies in network traffic. In addition, feature selection plays an important role in identifying the optimal subset of features to determine abnormal packages.Dirickle's hidden allocation identifies the optimal set of wiggies for classification, and the genetic algorithm has been used to calculate the score of data items and generate a population of candidate subgroups.And using the evaluation function, the competence of the elements of the current population is determined and finally after filtering the elements are better selected for the population of the next generation. This method was performed on the KDD-Cup99 dataset and experimental results show that the combined method obtained better accuracy to detect known and unknown attacks. A lower false positive rate has also been reported. In reference [19] the purpose in IDs intrusion detection systems is to detect a wide variety of malicious network traffic that is not detected by a normal, conventional firewall. Many penetration detection systems have been developed based on machine learning techniques. This article presents a new feature display method called cluster centralized method and nearest neighbor. The feature display method is an important classification pattern that makes it easy to classify correctly, however, few studies have focused on how to extract more important features for normal connections and effectively detect attacks. In the cluster center and nearest neighbor (CANN) method, two distances are measured. The first distance is based on the distance between both the data sample and the center of its cluster, and the second distance between the data and its closest neighbor in the same cluster. The results show that the accuracy of Cann classification is greater than that of KNN and SVM in the KDD-Cup99 dataset. It also provides high computational efficiency in time, training and classification testing. In reference [20] for maximizing the effectiveness of each feature extraction algorithm and creating an efficient penetration detection system, a set of feature extraction algorithms has been implemented linear differential pattern analysis (LDA) and PCA. This method has led to good results and has shown greater accuracy compared to a specific feature extraction method. The aim of the survey is to enable the use of the set of features generated by the flexible feature extraction

duroche to detect penetration. This article achieves high accuracy in the penetration detection system. The low number of features means that the stratification requires less information for training. Experimental results show that dimension reduction can increase detection rate, as well as the sum of feature extraction methods, directly demonstrate better performance in detecting penetration rate. The following article will first discuss the proposed method and the implementation methods in this study, and then examine the results of simulated algorithms, and finally draw conclusions and compare different algorithms.

2. proposed method

The term malware refers to viruses, worms, trojans, and any other program created with the intention of acts of sabotage. Countless numbers and large variations in existing malevolent codes make it difficult to accurately classify them. Malware falls into different categories: viruses, Trojan horses, worms, logic bombs, spyware, etc. But what has raised concerns about this is that today, malware developers are producing hybrid malware that is much more dangerous than simple malware. These malware have the properties of two or more types of malware together, and the damage they cause to the system is more common than simple malware. It is commonly used to copy a Trojan onto a target system, aze a virus or computer worm. The above process is called dropping. A Trojan, or Trojan horse, is a program that appears to be useful or safe, but contains hidden codes that are used to misuse or harm the system on which it is run. Trojan horses are usually sent to users through emails that indicate the purpose and function of the application as something other than its Truth [1].Such programs are also called Trojan codes. The Trojan horse exerts a sabotage operation on the system when executed. The main purpose of a Trojan horse is to spoil the work of the user or the typical operation of the system. For example, the Trojan may open a back door in the system so that the hacker can steal information or change the configuration of the system. There are two other equivalent terms that mean the same Trojan, namely RAT and Rootkit. So it's essential that there are ways to deal with this kind of software. There are generally two types of methods for dealing with malware: signature-based methods and behavior-based methods [1].

Malware is one of the most important threats of the modern era, according to research by Symantec over the decade since November 2007, about 60 percent of the software downloaded was malware. [18]

Almost all existing dynamic analysis methods, to check whether a software is healthy or bad-minded, run it in a virtual environment or an isolated environment so that the software's performance does not harm the main system. In general, after the execution of the software was completed, it was determined whether the malware was malware by checking the performance of the software. In fact, the software review process does not coincide with its implementation. Rather, after the implementation of the software in question has ended, its malware is determined.

One of the problems with anti-virus software today is the need for a database of malware signatures and updating this database. The problem increases when the updating process needs to be done in large numbers at short intervals to protect the system. Another problem is that timely updating of the antivirus software database does not guarantee that computer systems will be immune from damage. This is because the time between the release of the malware and its discovery is significant. For this reason, it is necessary to be able to perform the malware detection process in such a way that the need for repeated updates to the malware signature database is eliminated. Behavioral modeling methods are used for this purpose. Trying to model the behavioral patterns of malware. Using these methods, new malware can be detected without the need for prior knowledge about them.

In signature-based methods, we try to identify malware by using its static properties. These methods, which are used in all today's protective software, are very weak in detecting malware. Considering that these methods use static properties., A simple blurring algorithm can make them unable to detect malware.

Behavior-based methods, instead of using static properties, try to identify malware by investigating software behaviors. These methods are resistant to code blurring techniques and due to the use of software behaviors, they also identify malware in which M-blurring techniques are used. In general, these methods use modeling and data mining solutions to detect malware. After collecting a large number of malware and observing their behavior, a model of bad-minded behavior is developed and this model is used to detect malware.

Almost all existing methods of checking whether a software is healthy or bad-minded run it in a virtual environment or an isolated environment so that the software's performance does not harm the main system. In general, after the implementation of the software is completed, it is determined whether the software was malware by checking the performance (behavior) of the software. In fact, the software review process does not coincide with its implementation, but is determined to be malware after the implementation of the ramrod software is completed. This prevents the software review process from being carried out in the real environment, meaning that in order to prevent damage to the operating system, the software review process is carried out in an isolated or virtual environment and if the software is intact, it is allowed to run on the real system.

This research is the use of data mining methods to detect malware. Because attacks are constantly occurring, and traditional penetration detection systems cannot detect these attacks. When infiltration occurs the most important task is identification. The occurrence of penetration at any time is related to a pattern of events that have occurred in the past. These historical data are a very important source of traits that need to be effectively identified as signs and symptoms of intrusion into the data set. Data mining helps the process of building these models by discovering the right patterns from previous data. In this method, a set of rules of categories of network data is obtained. These laws have the ability to determine normal behavior from abnormal.

One of the fundamental problems for understanding malicious behavior correctly and the new trend in malware development is that malware, like computers and software, has changed drastically. And they use more sophisticated ways to escape detection. Expression compression is one of the best known ways to ambiguate malware and avoid detecting it. In addition, attackers often use unknown vulnerabilities, over-the-counter techniques of productive algorithms that greatly increase the impact of malware and its number.

The problems with today's anti-virus software are the need for a malware signature database and updating this database. The problem is increased when the updating process needs to be done in large numbers and at short intervals to protect the system. Another problem is that timely updating of the antivirus software database does not guarantee that computer systems will be immune from damage. This body issue is the reason why the time interval from the release of malware to its discovery is significant. For this reason, it is necessary to be able to perform the malware detection process in such a way that the need for repeated updates to the malware signature database is eliminated. The objectives of the research are:

Introducing a method to detect Trojan malware with high accuracy and prevent it from running on various platforms.

It's a new way of detecting malware that covers existing problems and solves them properly.

The proposed model is based on the feature selection method, the combination of data mining algorithms and artificial intelligence. Because the KddCup99 data has many duplicate rows (these duplicate rows disrupt the training of existing class clauses) and also has a large number of features that require filtering as well as ranking them. The model presented is divided into two phases: pre-processing and penetration detection, which is provided to the model in the pre-processing phase by removing duplicate rows and removing low-impact records. Efficiency assessment will be carried out by two sets of training and testing data, aimed at increasing the percentage of safe, contaminated and suspicious communication detection.

the steps of the proposed method are described case by case in this section and are discussed in detail in other sections of this chapter.
1-Receiving the data set
2- Delete additional records
3- Selection of effective features
4-Creating a model using an ant colony
5- Evaluation of the model
2-1-Receiving the data set

This dataset consists of 41 predictor fields and one target field. The collected data was obtained from about half a million connections, which was reduced to 494,024 by removing duplicate data by the feature selection algorithm. The following table shows the list of features of this dataset along with its description:

Table 1- List of basic features in the dataset

Description	**The name of the feature**
Duration of communication between origin and destination systems	duration
The communication protocol includes tcp, udp and...	Protocol type
The type of service offered in communication includes telnet, ftp and...	Service
Number of bytes sent from Origin to destination	Src_Byte
Number of bytes sent from destination to Origin	Des_Byte
Determining whether the communication is normal or wrong	Flag
1 if the connection to the host is current and 0 otherwise	Land
Number of missing or damaged data packets in connection	Wrong fragment
Percentage of correct data packets received at the destination	Urgent

Table 2- List of content features in each communication

Description	**The name of the feature**
The number of influencers in each relationship	Hot
The number of failed attempts to login to the current host	Num_failed_login
1 if the authorized user is logged in, otherwise 0	Logged_in
Number of remaining login opportunities	Num_compromised
1 if the root shell has been reached, 0 otherwise	Root_shell
1 if su_root has been called and executed, 0 otherwise	su_attempted
The number of root accesses	Num_root
The number of files created during the connection	Num_file_creations
The number of shells in the current connection	Num_shells
Number of attempts to access files	Num_access_files
Number of commands executed in ftp_session	Num_outbound_cmds
1 if the current host is entered, otherwise 0	Is_hot_login
1 if the guest host is logged in, otherwise 0	Is_guest_login

Table 3- List of related traffic characteristics

Description	**The name of the feature**
The number of connections from the origin to the current host during the last 2 seconds	Count
Percentage of connections that have a Syn error	Serror_rate
Percentage of connections that have a rej error	Rerror_rate
Percentage of connections to the current host	Same_srv_rate
Percentage of connections to different hosts	diff_srv_rate
The number of connections from the source to the current service during the last 2 seconds	Srv_count
Percentage of connections on the same service that have a Syn error	Srv_serror_rate
Percentage of connections on the same service that have a rej error	Srv_rerror_rate
Percentage of connections to different hosts	Srv_diff_host_rate

All the introduced features are used as predictors in the proposed model, the target field in this dataset includes five general classes of attacks (communication status) as follows:

Denial of Service Attack: In this type of attack, the attacker tries to influence the process of computing the host computer by relying on heavy processing operations or filling the memory of the host computer. In these attacks, users' access to the host system is usually denied.

User to Root Attack (U2R): In this type of attack, the attacker aims to damage the host system by showing his activities and logging in as a normal user (this login will be done by entering a password).

Remote to Local Attack (R2L): This type of attack occurs when a user with the permission to send packets over the network platform logs into the host system as a normal user (while there is no active user account in that computer) and It is trying to create havoc in the host system by exploiting this issue.

Probing Attack: In this type of attack, the attacker aims to obtain information about the network and its connection, to achieve a specific goal in controlling the security of the system, which is sabotage.
Normal: the classic mode in the connection between origin and destination.
The following table shows how many of each of the above attacks have occurred in the dataset (these numbers appeared after performing the RoughSet algorithm on the said dataset).

Table 4- Distribution of attack classes in the dataset

R2L	**U2R**	**Probe**	**Dos**	**normal**	
1126	52	4107	391458	97277	**number of attendance**

Each of the above classes includes several types of attacks, which are listed in the table below. Also, in this table, the emergence of each type of attack depends on what characteristics. (based on the features determined by the feature selection algorithm)

Table 5- Member attacks of classes and dependent features of each

number influential characteristics	attack class	Attack name
5,6	DOS	Back
7	DOS	Land
3.4,5,23,26.29,30,31,32,34,36,37,38,39	DOS	Neptune
8	DOS	Pod
2,3,5,6,12,25,29,30,32,36,37,39	DOS	Smorf
8	DOS	Tear drop
27	PROBE	Satan
36	PROBE	Ipsweep
5	PROBE	Nmap
28	PROBE	Portsweep
3,6,12,23,25,26,29,30,33,34,35,36,37,38,39	normal	Normal
11,6,3,4	R2L	Guess_passwd
9,23	R2L	ftp_write
3,39	R2L	Imap
6,10,14,5	R2L	Phf
23	R2L	Multihop
6,1	R2L	Warezmaster
3,24,26	R2L	Warezclient
39,1	R2L	Spy
3,24,14,6	U2R	Buffer_overflow
36,24,3	U2R	Loadmodule
14,16,18,5	U2R	Perl
24,23,3	U2R	rootkit

In this research, the aim is to identify three normal, suspicious and infected classes, where the R2L class type is considered as a suspicious relationship, due to the following reasons.
In these attacks, the attackers try to identify themselves as users of a system by guessing the password and such operations, which is a little milder than the other three types of attacks that directly cause sabotage in the host system.
From another point of view, the probability of failure or victory in these attacks is equal, the result of which, in case of victory, is to control the system temporarily or locally.

2-2- Selecting the feature
The KddCup99 dataset has one million data entries in which duplicate lines are abundantly found. Failure to delete or edit these lines will cause bad training of the classes that use these data to generate their prediction model. For this purpose, the dependence rate of the output classes with each of the available fields should be checked first

and determine which features have a direct effect on the emergence of which classes. This helps to remove unnecessary rows from the dataset in addition to having a correct understanding of our training dataset. In this research, the algorithmic feature selection method has been used.
This method is based on two standard coefficients, whose general form is S = U, A, V, f. If we assume that U is a finite set of N objects, A is the defining attributes of these objects, V is the attribute values, B is a subset of the attributes of A, then the inseparable set IND(B) can be defined as follows:

$$IND(B) = \{(x, y) \in U \times U \mid Attr(x) = Attr(y), \forall Attr \in B\}$$

In this regard, all x values of the member of the set B must also be a member of its inseparable set.

$$[x]B = \{y \in U \mid (x, y) \in IND(B)\}$$

For a better understanding of inseparable sets, consider the following table from the kddcup dataset:
Table 6. Different values of dependent features in determining the type of communication class

type of connection	number of bytes received	number of bytes sent	type of service	Type of protocol	Row
Normal	1460	130	http	tcp	1
Normal	1580	240	http	tcp	2
Smurf	16200	4500	ecmp	tcp	3
Satan	0	1032	Private	Udp	4
Normal	64500	1590	http	tcp	5
Normal	18000	2500	Private	tcp	6

In this table, A is equal to 5 attributes of protocol type, service type, number of bytes sent, number of bytes received and connection type (as the target field). The inseparable set IB will consist of {(1, 2, 5)(3)(6)(4)}. (Equivalence values are placed in a group) This inseparable set represents well the repeatability in the dataset on the feature set B.
2-3-Determining the degree of dependence between the target field and other fields
As mentioned earlier, the selection of algorithmic features is based on two standard coefficients, which are called high B and low B. If we consider X as a subset of U with attributes B, then upper B defines all values in IND(B) that have at least one member in common with X, and lower B denotes all values whose members are all subsets of X. The following formula is the mathematical scheme of the above definitions:

$$\underline{B} = \{x \in X \mid [x] \underline{\subset} X\}$$
$$\overline{B} = \{x \in X \mid [x] \cap X \neq 0\}$$

If in the above set, we consider all connections of Normal type equal to set X, then:

B-(x)={1,2,5,6}, B-(x)={6}

will be. Now, assuming that sets C and D are considered as subsets of attributes A and have nothing in common with each other, the degree of dependence between two sets of attributes C and D is defined as follows:

$$k = \gamma(C, D) = \sum_{\underline{X \epsilon U}}^{D} \frac{|\underline{C}X|}{|U|}$$

Roughset sets can determine the degree of dependence of features with the target field and also remove duplicate records (Olosola et al., 2010). This work will greatly help the classifiers to increase the productivity by not training their model several times and prevent the creation of Overfitting space. (Nidhi Sharma et al., 2008).
2-4- Residency in memory
At this stage, all the predictive fields along with the target field should be provided to the class. In this case, if a large amount of dataset entries are to be read from the files at each identification time, the efficiency of the model in online detection will be significantly reduced, for this purpose, by making the dataset file reside in the memory, the time to access the file streams and read from it we will remove that this work will help to reduce the detection time.

2-5- Creating a model based on the proposed algorithm
One combination of ant colony algorithm and fuzzy approach is the Fuzzy Ant Colony Clustering (FACC) algorithm. FACC combines the ant colony optimization (ACO) algorithm with fuzzy logic principles to improve the clustering process.

Here are the relevant details and formulas for Fuzzy Ant Colony Clustering:

1. Problem Definition:
- Given a dataset with n data points, the goal is to partition the data points into k clusters such that the intra-cluster similarity is maximized and the inter-cluster similarity is minimized.

2. Fuzzy Ant Colony Clustering Steps:
a. Initialization:
- Randomly initialize k centroids as the initial positions for the ants.
- Calculate the initial pheromone values for each data point and centroid pair.

b. Ant Movement:
- Each ant selects a data point based on the fuzzy membership values of the centroids.
- The fuzzy membership values are calculated using a fuzzy membership function, Gaussian membership functions.
- The ant moves towards the selected data point based on the pheromone values and a heuristic information value, such as the distance between the data point and centroid.

c. Pheromone Update:
- After all ants have moved, update the pheromone values based on the quality of the solutions.
- The pheromone update rule can be defined as:
pheromone(i, j) = (1 - ρ) * pheromone(i, j) + Δpheromone(i, j)
where ρ is the evaporation rate, and Δpheromone(i, j) is the amount of pheromone deposited on the edge between data point i and centroid j.

d. Cluster Formation:
- Assign each data point to the centroid that has the highest fuzzy membership value for that data point.
- Update the centroids based on the newly formed clusters.

e. Termination:
- Repeat steps b to d until a termination criterion is met, such as a maximum number of iterations or convergence of the solutions.

3. Evaluation:
- Calculate the objective function value to evaluate the quality of the clustering solution.
- The objective function can be defined as the sum of the intra-cluster similarity and the inverse of the inter-cluster similarity.

4. Parameters:
- The FACC algorithm requires several parameters to be defined, including the number of ants, the number of clusters, the evaporation rate, the fuzzy membership function, and the heuristic information value.

3-8 K-Fold training
The training dataset has a normal class, which of course, for testing the data, the model should be trained by all samples. In order to train the model better and also due to the limitations of using large datasets in the main memory, the K-Fold method has been used.

3. Results

Intrusion detection systems are responsible for identifying and detecting any unauthorized use of the system, abuse or damage by both internal and external users. Detecting and preventing intrusion is considered as one of the main mechanisms in meeting the security of networks and computer systems, and it is generally used alongside firewalls

and as a security supplement for them. In this section, the proposed method will be reviewed and its results have been reviewed.
In this section, we will compare the proposed model with other presented models. The comparison criterion here is the percentage of classification accuracy in detecting the type of connection class. In order to determine the accuracy of the predictions made, we have used the confusion matrix along with the following formula.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100$$

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \times 100$$
$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \times 100$$
$$\text{F\_Measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \times 100$$

Here, TP equals the expected correctly classified examples in the current class, TN equals other correctly classified examples from other classes in the matrix, FP equals the number of failed predictions of the current class in other classes, and FN equals the number of predicted failed examples. It is in the current expected class. The results of algorithms used in data mining software and MATLAB simulation environment are displayed separately. These algorithms are used in a basic and combined form and their results are displayed in the form of graphs and tables for different criteria.
In Table 6-4, the results of the accuracy criterion of the proposed method on the data set are shown with a number of other classification algorithms such as Random forest, J48. The accuracy rate of the algorithms that used the combined process of etiquette is equal to 85% on average.
To create a suitable model in the process of data classification, the use of clustering can increase the ability to recognize samples in creating a model. Therefore, in the stage of creating the model, the created rules will create more distinct clusters, so the amount of attack or being a normal person will increase, in which the accuracy rate is equal to 96%. Figure 3 shows the results of different algorithms of this research.

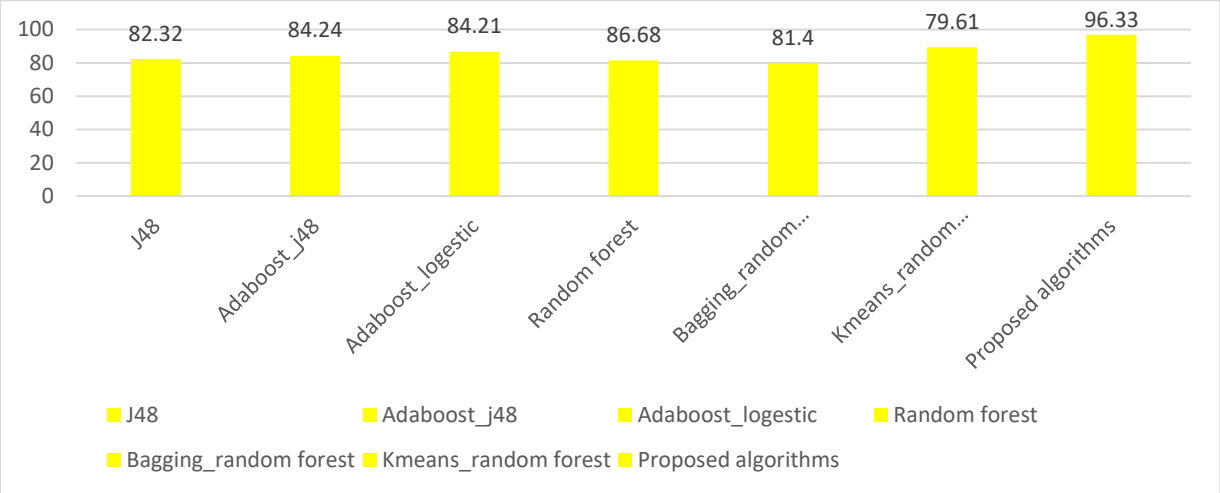

Figure 3- Results of the accuracy criteria of the proposed process and data mining algorithms

Table 7- The results of the accuracy criteria of the proposed process and data mining algorithms

Type of meta hurestic approach used for rule mining	Accuracy(%)
J48	82.32
Adaboost_j48	84.24
Adaboost_logestic	86.68
Random forest	81.40
Bagging_random forest	79.61
Kmeans_random forest	89.87
Proposed algorithms	96.33

In data mining researches, it is not enough to consider accuracy criteria alone and other criteria should be used for evaluation. Table 8 compares the results of the Precision criterion of the proposed process and other classification algorithms. The Precision measure will show the percentage of detection of positive samples, in fact, it represents the operation in which the attack did not occur. The amount of this criterion for the proposed process is equal to 97%. As the results of other algorithms have shown, the proposed algorithm has more suitable results than other algorithms.

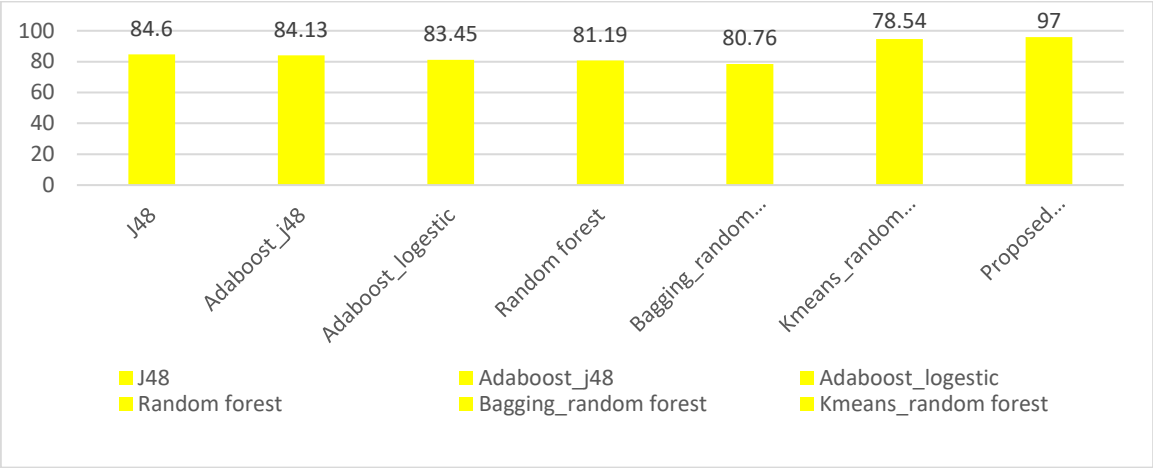

Figure 4- Precision benchmark results of the proposed process and data mining algorithms

Table 8- The results of the Precision criterion of the proposed process and other data mining algorithms

Type of meta hurestic approach used for rule mining	Precision (%)
J48	84.60
Adaboost_j48	84.13
Adaboost_logestic	81.19
Random forest	80.76
Bagging_random forest	78.54
Kmeans_random forest	94.60
Proposed algorithms	97

Table 9 compares the results of the Recall criterion of the proposed process and other data mining algorithms, this criterion for the proposed process is equal to 92%. This criterion shows that this algorithm has been able to have a high percentage of correct answers on negative samples, that is, the existence of an attack and higher results than other algorithms. In hybrid algorithms such as the used algorithm approach, the created models are not specific to the training data, and therefore, appropriate results will be made in the prediction. The reason that these models are not specific is due to the different sampling done in this data, with different sampling and creating a model with them, all aspects of the data set will be taken into account, so the amount of criteria obtained will be higher.

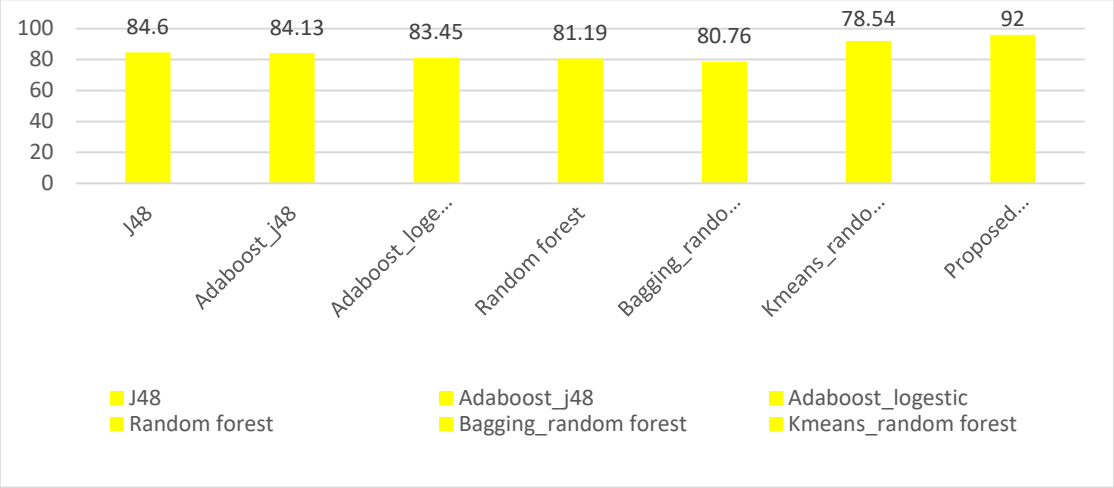

Figure 5- Recall criteria results of the proposed process and data mining algorithms

Table 9- The results of the Recall criterion of the proposed process and other data mining algorithms

Type of meta hurestic approach used for rule mining	Recall (%)
J48	80.76
Adaboost_j48	85.86
Adaboost_logestic	83.07
Random forest	82.88
Bagging_random forest	80.18
Kmeans_random forest	80.11
Proposed algorithms	92

Table 10 compares the results of the F_Measure criterion of the proposed process and other data mining algorithms, this criterion for the proposed process is equal to 92%. This criterion shows that this algorithm has been able to have a high percentage of correct answers on negative samples, that is, the existence of an attack and higher results than other algorithms. In hybrid algorithms such as the used algorithm approach, the created models are not specific to the training data, and therefore, appropriate results will be made in the prediction. The reason that these models are not specific is due to the different sampling done in this data, with different sampling and creating a model with them, all aspects of the data set will be taken into account, so the amount of criteria obtained will be higher. .

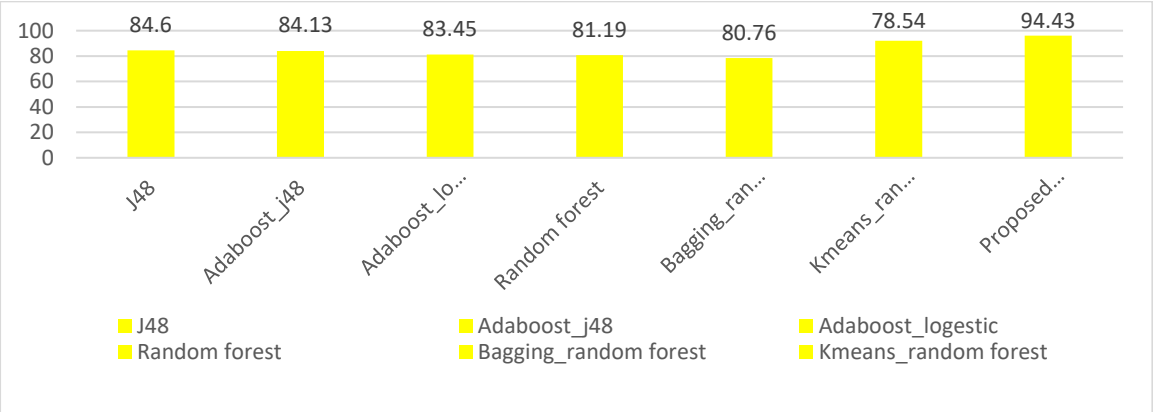

Figure 6 - F_Measure results of the proposed process and data mining algorithms

Table 10- The results of F_Measure of the proposed process and other data mining algorithms

Type of meta hurestic approach used for rule mining	F_Measure (%)
J48	80.76
Adaboost_j48	85.86

Adaboost_logestic	83.07
Random forest	82.88
Bagging_random forest	80.18
Kmeans_random forest	80.11
Proposed algorithms	94.43

4. Discussion

The proposed method provides better performance in detecting attacks compared to methods such as random forests and hybrid algorithms. The detection percentage of infected and normal classes by the proposed model is far higher than all the mentioned methods. This superiority can be due to the ranking of features and the creation of a suitable model, which will be compared using the fuzzy criterion. The purpose of intrusion detection systems is to detect the possibility of intrusion and to detect the general weaknesses of the system and report it to the system manager. Intrusion detection systems face a lot of data, one of the most important tasks in which is to maintain the quality of features by removing inappropriate features. In this regard, this thesis has been carried out with the aim of increasing the accuracy of the intrusion detection system by using the combination of data mining algorithms. The data of KDDCup99, which is related to the types of attacks, has been used. In the first stage, the feature selection operation has been carried out and suitable features have been selected. In the second stage, these features have been ranked using the gain criterion, and using the colony Ants and fuzzy criteria will be used to detect attacks, and these results have been compared with other algorithms and their results have been compared in the form of tables in the previous chapter. The proposed model has high accuracy in detecting normal and contaminated classes. As a solution in the future, it is possible to investigate more classes except normal and infected and try to implement stronger algorithms to detect the characteristics that influence the appearance of these classes, because due to the nature of suspicious connections and also their uncertainty, sometimes they are considered deterministic attacks. and contaminated communications are considered, which reduces the percentage of efficiency in multi-class intrusion detection systems.

Refrences

1. Mittal, M., Kumar, K., & Behal, S. (2023). Deep learning approaches for detecting DDoS attacks: A systematic review. Soft computing, 27(18), 13039-13075.
2. Sahidullah, M., Delgado, H., Todisco, M., Nautsch, A., Wang, X., Kinnunen, T., ... & Lee, K. A. (2023). Introduction to voice presentation attack detection and recent advances. Handbook of Biometric Anti-Spoofing: Presentation Attack Detection and Vulnerability Assessment, 339-385.
3. Ju, Z., Zhang, H., Li, X., Chen, X., Han, J., & Yang, M. (2022). A survey on attack detection and resilience for connected and automated vehicles: From vehicle dynamics and control perspective. IEEE Transactions on Intelligent Vehicles.
4. Basit, A., Zafar, M., Liu, X., Javed, A. R., Jalil, Z., & Kifayat, K. (2021). A comprehensive survey of AI-enabled phishing attacks detection techniques. Telecommunication Systems, 76, 139-154.
5. Khraisat, A., & Alazab, A. (2021). A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges. Cybersecurity, 4, 1-27.
6. Zhang, J., Pan, L., Han, Q. L., Chen, C., Wen, S., & Xiang, Y. (2021). Deep learning based attack detection for cyber-physical system cybersecurity: A survey. IEEE/CAA Journal of Automatica Sinica, 9(3), 377-391.
7. Lin, W. C., Ke, S. W., & Tsai, C. F. (2015). CANN: An intrusion detection system based on combining cluster centers and nearest neighbors. *Knowledge-based systems*, *78*, 13-21.
8. Duque, S., & bin Omar, M. N. (2015). Using data mining algorithms for developing a model for intrusion detection system (IDS). *Procedia Computer Science*, *61*, 46-51.
9. Tian, Z., Luo, C., Qiu, J., Du, X., & Guizani, M. (2019). A distributed deep learning system for web attack detection on edge devices. IEEE Transactions on Industrial Informatics, 16(3), 1963-1971.
10. Soe, Y. N., Feng, Y., Santosa, P. I., Hartanto, R., & Sakurai, K. (2020). Machine learning-based IoT-botnet attack detection with sequential architecture. Sensors, 20(16), 4372.
11. Zhang, D., Wang, Q. G., Feng, G., Shi, Y., & Vasilakos, A. V. (2021). A survey on attack detection, estimation and control of industrial cyber–physical systems. ISA transactions, 116, 1-16.
12. Thaseen, I. S., & Kumar, C. A. (2017). Intrusion detection model using fusion of chi-square feature selection and multi class SVM. *Journal of King Saud University-Computer and Information Sciences*, *29*(4), 462-472.
13. Hussain, J., & Lalmuanawma, S. (2016). Feature analysis, evaluation and comparisons of classification algorithms based on noisy intrusion dataset. *Procedia Computer Science*, *92*, 188-198.

14. S. Balakrishnan, K. Venkatalakshmi, and A. Kannan, "Intrusion detection system using Feature selection and Classification technique," International Journal of Computer Science and Application, 2014.
15. Farrahi, S. V., & Ahmadzadeh, M. (2015). KCMC: a hybrid learning approach for network intrusion detection using K-means clustering and multiple classifiers. *International Journal of Computer Applications*, *124*(9).
16. Kasliwal, B., Bhatia, S., Saini, S., Thaseen, I. S., & Kumar, C. A. (2014, February). A hybrid anomaly detection model using G-LDA. In *Advance Computing Conference (IACC), 2014 IEEE International* (pp. 288-293). IEEE.
17. Aburomman, A. A., & Reaz, M. B. I. (2016, October). Ensemble of binary SVM classifiers based on PCA and LDA feature extraction for intrusion detection. In *Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2016 IEEE* (pp. 636-640). IEEE.
18. Tuan, T. A., Long, H. V., Son, L. H., Kumar, R., Priyadarshini, I., & Son, N. T. K. (2020). Performance evaluation of Botnet DDoS attack detection using machine learning. Evolutionary Intelligence, 13, 283-294.
19. Sriram, S., Vinayakumar, R., Alazab, M., & Soman, K. P. (2020, July). Network flow based IoT botnet attack detection using deep learning. In IEEE INFOCOM 2020-IEEE conference on computer communications workshops (INFOCOM WKSHPS) (pp. 189-194). IEEE.
20. Mothukuri, V., Khare, P., Parizi, R. M., Pouriyeh, S., Dehghantanha, A., & Srivastava, G. (2021). Federated-learning-based anomaly detection for IoT security attacks. IEEE Internet of Things Journal, 9(4), 2545-2554.