Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.043173

ARTICLE

Utilizing Machine Learning with Unique Pentaplet Data Structure to Enhance Data Integrity

Abdulwahab Alazeb*

Department of Computer Science, College of Computer Science and Information System, Najran University, Najran, 55461, Saudi Arabia

*Corresponding Author: Abdulwahab Alazeb. Email: afalazeb@nu.edu.sa

Received: 24 June 2023 Accepted: 23 October 2023 Published: 26 December 2023

ABSTRACT

Data protection in databases is critical for any organization, as unauthorized access or manipulation can have severe negative consequences. Intrusion detection systems are essential for keeping databases secure. Advancements in technology will lead to significant changes in the medical field, improving healthcare services through real-time information sharing. However, reliability and consistency still need to be solved. Safeguards against cyber-attacks are necessary due to the risk of unauthorized access to sensitive information and potential data corruption. Disruptions to data items can propagate throughout the database, making it crucial to reverse fraudulent transactions without delay, especially in the healthcare industry, where real-time data access is vital. This research presents a role-based access control architecture for an anomaly detection technique. Additionally, the Structured Query Language (SQL) queries are stored in a new data structure called Pentaplet. These pentaplets allow us to maintain the correlation between SQL statements within the same transaction by employing the transaction-log entry information, thereby increasing detection accuracy, particularly for individuals within the company exhibiting unusual behavior. To identify anomalous queries, this system employs a supervised machine learning technique called Support Vector Machine (SVM). According to experimental findings, the proposed model performed well in terms of detection accuracy, achieving 99.92% through SVM with One Hot Encoding and Principal Component Analysis (PCA).

KEYWORDS

Database intrusion detection system; data integrity; machine learning; pentaplet data structure

1 Introduction

The rapid expansion of wearable technology, smart homes, connected vehicles, and microgrids proves that the Internet of Things (IoT) will play a crucial role in shaping the future of the Internet. IoT is a technology that connects and exchanges data with other devices embedded within physical objects, incorporating various software, sensors, and cutting-edge technologies through the Internet. As of the latest data, more than 5 billion IoT devices are connected to the Internet [1].

By 2030, it is projected that nearly 125 billion IoT devices will be online, generating massive amounts of data [2]. However, due to bandwidth limitations and the significant growth in data, current

information systems are ill-equipped to handle and transfer such vast quantities of data to the cloud. In many cases, the proliferation of IoT devices may prove unfeasible.

Today, several critical and real-time IoT services are already integrated into our daily lives, including connected car systems, video conferencing applications, and monitoring systems. These services depend on low latency and location-based information to provide users with reliable, high-quality experiences [3].

To address the challenges mentioned earlier, a proposed solution is needed. Cisco developed fog computing, a virtualized framework that offers essential services close to the ground, including the capacity to handle large volumes of data, storage, and networking services [4]. Fog computing is particularly well-suited to secure applications that require real-time collection and location-based services. It improves security and privacy by maintaining and evaluating data near authorized users at the edge nodes [5].

Fog computing offers several distinct properties that define it as a cloud extension and provide additional services over the cloud. Firstly, its presence at the edge networks offers high-quality services and minimal latencies, making it ideal for healthcare systems, online gaming, streaming, and group meetings involving location-based services and minimal latency. Another essential feature of fog computing is the large number of disparate locations of fog nodes, facilitating flexibility in several systems, such as moving objects.

Fog's presence at the cloud's edge and its regional dissemination increases bandwidth efficiency and protects critical data. Fog computing may be able to tackle various issues in the medical system [6]. Protecting user or patient data is vital for fog computing and health services. However, relatively economical IoT devices typically have limited computing capabilities, making it challenging to integrate cybersecurity primitives on a system level. These devices usually lack a powerful Central Processing Unit (CPU), resulting in low network threat resiliency [1].

In the past, cyberattacks have frequently targeted IoT systems. For example, in the September 2016 Mirai assault, 400,000 IoT devices were attacked and used to launch a significant Distributed Denial of Service (DDoS) attack via a botnet. Similarly, in April 2020, the Dark Nexus botnet, built on the Mirai code, corrupted more than 1300 Internet of Things devices [7].

The term "cybersecurity" encompasses the methods and tools employed to safeguard digital assets such as computers, network services, applications, and data. The cybersecurity infrastructure is formed by the security systems of the network and the host computer. These systems are equipped with multiple layers of protection, including Intrusion Detection Systems (IDS), antivirus software, and firewalls [8]. The primary goal of an IDS is to monitor and defend a network against threats like data theft, tampering, and destruction. Intruders can originate from outside (external intrusions) or within (internal intrusions), both of which are considered privacy violations.

IDSs can be broadly classified into three categories: misuse-based, anomaly-based, and hybrid. Misuse-based approaches identify malicious activity by analyzing attack pattern signatures. However, they are unable to detect new (zero-day) attacks. Anomaly-based algorithms, on the other hand, model typical system and network performance to detect abnormalities, which are defined as deviations from the norm. Their ability to identify zero-day attacks makes them appealing. Still, they may have high False Alarm Rates (FARs) due to the mislabeling of previously undetected (but legitimate) system activities as anomalies. Hybrid methods combine anomaly and misuse detection, increasing the True Positive (TP) ratio for known intrusions while reducing the False Positive (FP) rate for unknown attacks.

Privilege elevation attacks occur when an attacker gains unauthorized administrative access to a system. Users can then access and misuse sensitive information by making unauthorized copies. One of the most dangerous attacks is SQL injection, which involves the insertion of malicious code through a vulnerability in the front end, such as inadequate authorization or improper validation [9].

A Database Intrusion Detection System (DIDS) can identify and report attackers who submit malicious queries to a database. There are two types of intrusion detection systems: signature-based and anomaly-based. Signature-based methods detect only predefined patterns, while anomaly-based detection monitors all regular queries, classifying deviations from typical queries as attacks [10]. Our proposed system employs anomaly detection.

Database attacks generally fall into two categories: insider and external attacks. Insider attacks involve individuals within an organization abusing their privileges, while external attacks are executed by individuals outside the organization who have gained unauthorized privileges. This can occur when database privileges granted to users exceed what is necessary. The DIDS system effectively defends against both types of attacks [11,12].

Thus, the proposed system is an enriched IoT-enabled intrusion detection monitoring scheme using machine learning and Pentaplet architecture. The IoT enables proper monitoring, and ML enables intrusion detection. Our proposed system addresses these vulnerabilities and can identify any of these threats, alerting the database administrator and even canceling the transaction as needed. If no anomalies are detected, the system takes no action. The Pentaplet structure stores all database-related information transaction by transaction. With a transaction-based approach, a single Pentaplet can keep all queries within a transaction. The proposed DIDS employs a supervised machine learning method, the Support Vector Machine (SVM), to train and identify malicious queries.

2 Literature Review

The article [13] highlighted the deployment of various one-class classifiers to stop assaults on the Message Queuing Telemetry Transport (MQTT) protocol used by the IoT. They have been able to train the one-class algorithms by exploiting simple datasets, demonstrating outstanding performance in identifying attacks.

Applying Machine Learning (ML) algorithms to a dataset while comparing and evaluating their performance was the primary goal of the research [14]. They employed Correlation-Based and Chi-Squared-Based feature selection algorithms to minimize the datasets by removing unnecessary data. For this research, the NSL KDD dataset was utilized. The Artificial Neural Networks (ANN) model performs far better on this particular dataset than SVM.

The study [15] provided an approach for constructing a more effective Intrusion Detection System (IDS) by using anomaly detection and data-mining techniques. The data mining techniques will continuously simulate what a typical network should look like and lower the procedure's false positive and false negative alert rates. They used classification-tree techniques to make precise predictions regarding the sessions of possible attacks.

Research on ML and data mining approaches for cyber analytics was presented in the study [8]. The ML/DM algorithm's complexity was discussed, problems for applying ML/DM for cybersecurity were also described, and they recommend when to utilize a specific given method.

The processes for damage assessment utilizing various versions of data from the Database System were the primary purpose of the research [16]. It is achievable to reduce the consequences of fraudulent database transactions by delivering suitable variations of data items to exchanges during the damage

assessment phase if the suggested multi-version data method is utilized. This will allow for the elimination of the effect of malicious data transfers.

The method researchers provided in the study [17] can detect attacks at the database transaction and inter-transaction levels based on these two attacks they developed. For this reason, they suggest a detection mechanism at the transaction level that is based on characterizing the everyday activities within the database systems. This will allow us to determine whether or not an anomaly has occurred. Furthermore, at the stage of inter-transactions, they present a detection approach that is founded on the concept of anomaly identification and makes use of data mining to discover dependence and sequencing rules. This method has a distinct advantage over earlier database intrusion detection systems because it can identify suspicious attacks on both transactional and inter-transactional levels.

The paper [18] presented an entirely autonomous database intrusion detection system that detects internal and external threats and may prevent breaches not seen by networks or hosts based on IDS. The designed methodology is flexible and can be fine-tuned as databases become more sophisticated and dynamic. Anomaly detection and role-based access restriction are implemented in their architecture. An Octraplet-based data structure is used to store SQL queries. The Naive Bayes Classifier approach is used to identify abnormal requests in this system. The method that has been proposed has the potential to increase both the detection rates and the overall efficiency of the system.

As demonstrated in this work, data-dependence connections can be used to identify suspicious activity in a database management system [10]. The proposed approach compares the dataset with items received or created by legitimate user transaction data to find fraudulent transactions. Petri-Nets describe typical data update patterns at the user level, and they have developed techniques for identifying the interconnections among transactions that rely on data.

In this research [19], they present new data mining methods that will be used to create data dependencies, miners, for the database IDS. This approach will be called the ODADRM (Optimal Data Access Dependency Rule Mining). To make the k-optimal rule discovery algorithm more applicable to the database IDS, ODADRM was developed as an extension of this technique. ODADRM circumvents a significant number of the restrictions that were present in earlier data dependency mining algorithms.

The authors [20] proposed a novel technique for anomaly identification called Fog-Empowered anomaly detection. This methodology allows the use of the processing capacity offered by the fog platform and an effective hyper-ellipsoidal clustering model.

In the study [21], the authors proposed an effective technique for sequential pattern mining on network traffic data. The proposed solution provided highly accurate data mining results and preserved sites' privacy. Using the N-repository server model, which made numerous servers act as a single mining server, and the retention replacement methodology, which altered the result based on a certain probability, the system frequently detected recurring network traffic patterns while concealing site information. Additionally, the technique kept meta tables in each site to quickly ascertain whether candidate patterns had ever been there. This increased the efficiency of the overall mining process. Additionally, they conducted thorough testing on actual network traffic data to show the accuracy and effectiveness of the suggested approach.

The authors of the research [22] proposed a hybrid system for IDS called Convolutional Neural-Learning Classifier System (CN-LCS), which combines a Learning Classifier System (LCS) with a Convolutional Neural Network (CNN) to identify intrusions on databases, particularly against insider intrusion. The study of the CN-LCS classification results using the t-SNE technique showed that

the low-dimensional embedding of the query commands caused the low classification performance. Furthermore, the proposed CN-LCS outscored other machine learning classifiers in experiments, with a test accuracy of 94.64%.

The authors of this work [23] presented a unique method for detecting illegal user activity in databases. Their newly proposed outlier mining approach could detect vulnerabilities such as a compromised user account or unauthorized use by a user. They focused on detecting abnormalities in a user's behavior that could indicate a wide range of harmful behaviors. The suggested technique was based on two major components that analyzed the consistency of a user's behavior and compared it with activity patterns learned from previous access. The first component is used to test for self-consistency, which determines whether a user's actions are consistent with previous patterns. The second component analyzed global consistency to see whether a user's activities are compatible with the prior behavior of users with similar characteristics. The combined system achieved an F1-score of up to 0.88, which is a combination of both the first and the second component.

The researchers of this publication [24] created a new method for determining database intrusions by combining data resources and employing belief updates as part of their research. The model utilized information gathered from both present and previous user behavior to detect an intrusion. This approach comprised a rule-based element, a belief combining element, a security sensitivity history database element, and a Bayesian learning element. The modified Dempster's method connected various proofs from the rule-based element. This was done to compute a preliminary belief regarding each incoming transaction. The outcomes of the experimental assessment demonstrated that the suggested database intrusion detection system was capable of effectively detecting intrusive assaults despite producing an excessive amount of false alarms.

The study [25] developed a new concept for a smart government structure using fog computing technologies. Data control and management were the primary goals of this study. They came up with some novel algorithms and tested them out to see how well the model could protect the data authenticity of the system in the situation where it came under assault. Several techniques were implemented to protect systems from fraudulent transactions or fog node data manipulations. To examine and keep track of the goings-on of each transaction, the framework incorporates the functionality of a transaction-dependency graph.

This research [26] discussed the datasets commonly used for training and evaluating intrusion detection systems. Then, the paper presents a comparison of various machine learning techniques and their effectiveness in detecting different types of attacks. Overall, the article provides a comprehensive overview of the current research in the field of network intrusion detection using machine learning techniques.

Kernel techniques and Support Vector Machine (SVM) were suggested by the researchers of this work [27] to improve the accuracy of anomaly-based intrusion detection. A method to boost the identification rate and reduce false alarms was also developed by combining specification-based intrusion detection with anomalous intrusion detection. This study also created a framework for the automatic generation of software applications to identify both misuse and anomalies in intrusions. A Colored Petri Net (CPN) depicting an intrusion detection framework was quickly transformed from an SFT indicating an incursion. Accuracy was 93.5 percent for the Markov Chain kernel and a one-class SVM; detection performance was 91.75 percent, and false alarms were 5.5 percent, respectively.

For novel binary and multi-class classifications, this research [28] suggested a new approach for a system of detecting intrusions using Recurrent Neural Networks (RNNs) with deep learning. The dataset called "NSL KDD" was used to analyze the parameters of the standards to acquire an actual

detection performance, and shape-based gathering was used in the future to increase the model's efficiency. Using deep learning methods, they aimed to create an IDS that could be used to check modern systems using RNNs and other RNN-based architectures.

Researches [29] have shown how to quickly and efficiently retrieve all damaged database pieces of data following an assault using agent-based vulnerability assessment and recovery strategies. The technique for evaluating degradation made use of timestamps as well as numerous copies of each data element. Every agent handles a predetermined collection of data objects, which are unique to that agent among all others. Both aspects of the contest require the cooperation of all of the participants. The operations of assessing damages and recovering from them were performed concurrently by agents following the appropriate methods. This approach substantially reduced the recovery period compared to prior techniques because it accurately detected the damages instead of underestimating them. The authors of this study [30] suggested a Dynamic Sensitivity-Driven Rule Generation Algorithm to identify the invasive transaction and thereby protect vital data from alteration.

3 Proposed System

For the purpose of storing information that is relevant to Structured Query Language (SQL) transactions, our proposed system uses an innovative data structure known as a Pentaplet. By applying the ensembling method, several machine learning algorithms are utilized for training and classification. The benefit of this suggested technique is that it generates a new data structure, Pentaplet, for effective data storage, enhancing performance regardless of dynamic changes in database size and structure. This system also protects against database attacks such as privilege escalation, unauthorized privilege abuse, and SQL Injection attacks. When a user submits a database query, it is processed to generate a Pentaplet, which is a structure consisting of five arrays. The database administrator maintains the roles, which can be dynamically modified based on the requirements. Role profiles are generated using Pentaplet, and the new profiles are then be compared to the current profiles generated by the classifier from the log file. The checking module performs the comparison, and based on the result, the response handling algorithm either generates the appropriate response or notifies the administrator.

3.1 Proposed System's Structure

The proposed system consists of four main components: database log files, a response engine, a comparison mechanism, and a profile generator. The database log files are utilized to create standard profiles. In this process, machine learning algorithms are employed to train typical profiles from the log files. Ensembling is used to achieve our objective. The profiles are kept separate for comparison purposes. Whenever a user issues a transaction, a Pentaplet is generated. Based on the creation of pentaplets, a role profile is constructed. The resulting structure is then compared to the current structure. Fig. 1 illustrates the architecture. If there is a successful match, the query is executed. Otherwise, the response engine provides three different types of responses based on the matching percentage. If the match rate exceeds 8.0 on a scale of 10, the administrator receives an alarm warning. If the rate is between 6.0 and 8.0, the query is instantly blocked. Otherwise, the query proceeds. Fig. 1 depicts the suggested architecture, which collects analytical data from the database log files.

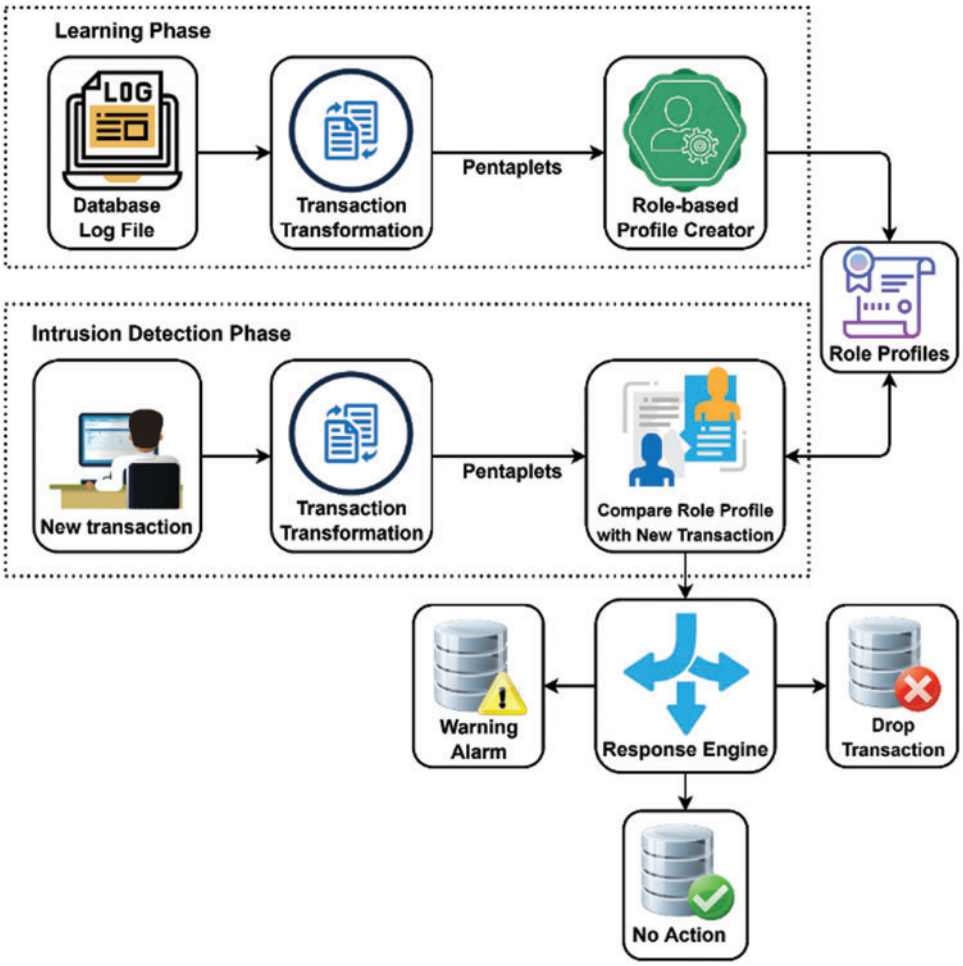


Figure 1: Diagram of the proposed intrusion detection system

The log file contains all data related to previous inquiries and transactions. The queries are then converted into binary values in a transaction processor before being inputted into a machine learning classifier [31] based on the roles that typically create normal profiles. Any deviation from the standard profiles is labelled as abnormal. The log queries are used to generate pentaplets. Similarly, pentaplets are formed from the user's requests and fed into an ensembling classifier to obtain probabilities.

3.2 Pentaplet Data Structure

To determine user behavior patterns, the suggested system queries database log files that contain information about users' actions. After undergoing the processing phase, the log entries are used to construct preliminary profiles that indicate acceptable activities. In order to create the appropriate profiles, each item (i.e., transaction) in the log file is treated as a single data unit. This section assumes that SQL queries linked to the same data transaction are grouped together in the log file. The system must first pre-process the contents of the log file and convert them into an understandable format for profile generation. As a result, each transaction is represented by a basic data block with five fields, hence the term "Pentaplet". Sets of these pentaplets are used to characterize user actions. Each transaction is denoted by a Pentaplet, which contains the following data: the first five array

components are used to represent the initial SQL command of a transaction, including the user-issued SQL command, the relationship sets queried, and the set of referenced attributes for each relation. If necessary, additional optional components may be added after the fifth element of the array to store additional data for the remaining SQL statements of the transaction.

The Pentaplet is a five-array relation-based data structure that can be expressed as P (F_C, R_P, A_P, R_S, A_S, ...,O_{RSQL}), where F_C represents to the first query command; R_P to the information of projection relation, which represents the attribute of projection array of each query; A_P to the information of projection attribute, which denotes the transaction projection attributes as a 2D array. R_S to denotes the information of selection relation, identifies the list of attributes used to filter results for each query, A_S to the information of selection attribute, which represents the transaction selection attribute as a 2D array, and lastly O_{RSQL} is optional, for any additional information about the remaining SQL queries in the transaction if there any.

If such a database structure is considered which consists of two relationships between R1 = [A1, B1, C1, D1] and R2 = [A2, B2, C2, D2], then the Pentaplet will be constructed as: {$SQL_{Command}$, $PROJECTION_{Relation}$[], $PROJECTION_{Attribute}$[][], $SELECTION_{Relation}$[], $SELECTION_{Attribute}$[][], $Optional_{RestSQL}$[]} [18,32].

3.3 Classifier

For the task of intrusion detection in RBAC databases, this work applies a collection of classifiers, including Decision Tree (DT), K-Nearest Neighbor (KNN), Logistic Regression (LR), Naive Bayes (NB), Support Vector Machine (SVM), and Random Forest (RF). The model is trained using the aforementioned conventional classifiers as well as an ensemble classifier. These classifiers are then combined with an ensemble voting approach to select the proposed solution with the highest accuracy. Once the classifier is successfully created, the classification results and corresponding experimental data are evaluated using a cross-validation technique. In classification problems, the classifier is given a group of examples to learn from and a new instance with predetermined attribute values. The task is to determine which category the new instance belongs to (corresponding to the observational set). Finally, the appropriate model provides a decision to anticipate the specific targeted value or class of this new instance, as shown in Fig. 2.

A total of six different classifiers, namely Logistic Regression (LR), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF), Naive Bayes (NB), and Decision Tree (DT), were used to construct our machine learning algorithm. The following sections discuss the classifier algorithms utilized in our proposed model.

3.3.1 Logistic Regression (LR)

Another supervised classification model is Logistic Regression (LR) which represents the input into the probability calculation for the dependent variable of interest. The dependent variable is binary; therefore, the values can either be "1" (success) or "0" (failure). It can be broken down into binomial, multinomial, and ordinal categories. The resulting LR equation is a straight-line equation, as follows:

$$y = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n \tag{1}$$

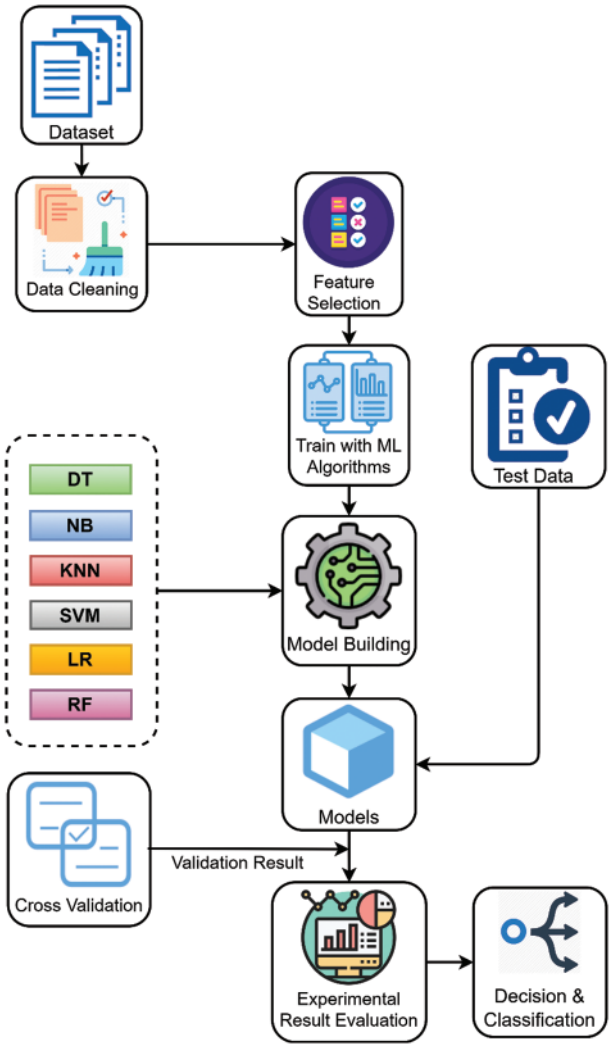


Figure 2: A block diagram of the classification pipeline

3.3.2 K-Nearest Neighbor (KNN)

Our proposed architecture used K-Nearest Neighbor (KNN), a supervised learning technique. The technique relies on the existing data being comparable to the new data. When classifying newly acquired information, it should be filed under a heading like those already in use. KNN's method of categorization is depicted in Fig. 3. To determine how far apart nodes are in this system, the Manhattan equation is employed for KNN. The distance formula of the Manhattan equation is shown in Eq. (2).

Manhattan:

$$\sum_{i=1}^{k} |x_i - y_i| \tag{2}$$

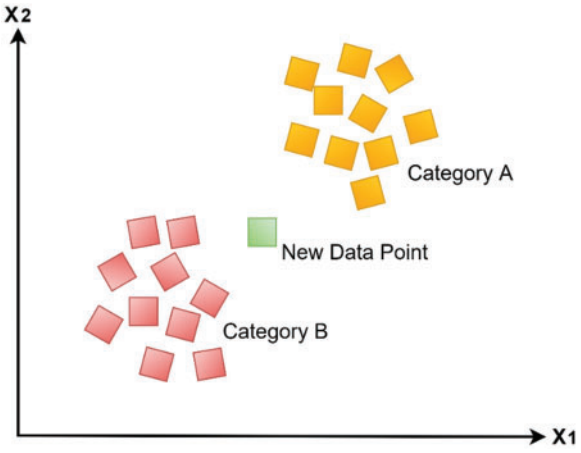


Figure 3: KNN classifier with the classification mechanism

3.3.3 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a classifier and regression technique that can classify any object based on the provided data. To make adding new information in the future simple, it generates the optimal decision boundary that partitions spaces with n dimensions into classes. Eq. (3) displays the corresponding kernel trick of the SVM equation, which was applied in our study to construct the classification. When dealing with data that cannot be separated linearly into two dimensions, the user of the dataset instead transforms it to a higher dimension, such as three, four, or even ten is called the kernel trick method.

$$\text{Kernel trick}: \; k\left(x_i, x_j\right) = x_i.x_j \tag{3}$$

3.3.4 Random Forest (RF)

The random forest is a classification and regression algorithm in machine learning. This technique is also known as an ensemble classifier because it employs numerous decision tree models. It is a classifier that uses the averaged results of applying numerous decision trees to diverse subsets of a dataset to improve the overall accuracy of the dataset's predictions.

3.3.5 Naive Bayes (NB)

Bayes' theory, often known as Bayes' rule or Bayes' law, is a formula that, given prior knowledge, may determine the probability of a hypothesis. The term "Naive Bayes" refers to the Bayes' theorem in its most common version. Bayes' theorem can be expressed mathematically, as seen in Eq. (4). Using the Bayes Theorem as its foundation, Naive Bayes is a probabilistic machine learning technique often applied to classification problems.

$$P(A|B) = (P(B|A)P(A))/P(B) \tag{4}$$

where, P(A | B) is the chance of occurrence. And P(B | A) is likely probability.

3.3.6 Decision Tree (DT)

As a form of supervised ML methods, a decision tree helps to make decisions more quickly where the data structure of a tree-based defines the algorithm. The internal nodes of a decision tree stand in

for characteristics of the dataset, while the branches stand in for the rules for making a call, and the leaves indicate the result. The decision tree is capable of representing any Boolean value, whether it be true or false.

$$S = -P\left(\frac{1}{true}\right) log_2 P\left(\frac{1}{true}\right) - P\left(\frac{0}{false}\right) log_2 P\left(\frac{0}{false}\right) \tag{5}$$

where,

- S = Total sample count
- P(1/true) = possibility of being YES
- P(0/false) = possibility of being NO

Based on its attributes ($a_1, \ldots, a_n$), the procedure given here is to assign the most likely class value to this new instance, $v_{class} \varepsilon V$, which is:

$$v_{class} = \arg\max_{v_j \varepsilon V} P\left(v_j | a_1, a_2, \ldots, a_n\right) \tag{6}$$

In this scenario, predicting $P(v_j)$ is easy because it only takes counting the frequency of v_j in the training data. $P(a_i \mid v_j)$ just requires a frequency count of the tuples in the training data that have a class value of v_j. Here, the suggested anomaly detection framework is directly applied ML algorithms through the following equation, which treats the set of roles in the system as classes and the log file Pentaplet as observations (7).

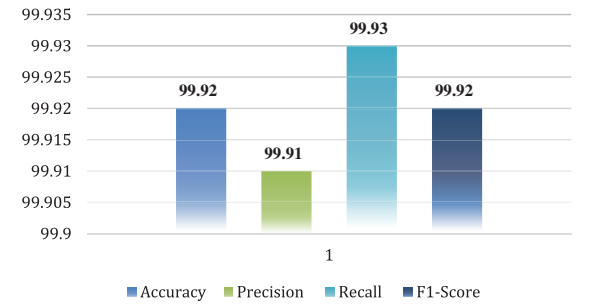
$$r_{\text{class}} = \arg\max_{r_j \varepsilon R} \prod_{c=1}^{N_C} p\left(r_j\right) p\left(c_i \mid r_j\right) \times \left\{\prod_{i=1}^{N} \left\{p\left(S_R[i].S_A[i] \mid r_j\right) p\left(S_R[i].S_A[i] \mid r_j\right)\right\}\right\} \tag{7}$$

For the equation above, N is the number of relations within the DBMS, whereas Nc is the number of SQL statements performed. The intrusion detection process is relatively simple with the above equation in place. The trained classifier is used to make predictions about the r_{class} of all future transactions. An anomaly is flagged if this r_{class} does not match the role originally linked to the transaction.

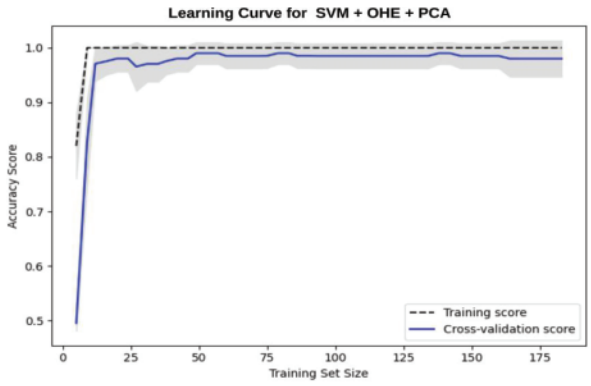
4 Experiments and Results

Our proposed work relied on the KDD Cup'99 dataset [33,34], developed by MIT Lincoln Lab and derived from the tcpdump subset of the 1998 DARPA IDS evaluation dataset. The dataset is also available in secondary data sources like Kaggle. The synthetic data was generated with a closed network and manually injected attacks to develop a wide variety of attacks without disrupting the normal flow of traffic. The competition aimed to create a network intrusion detector, a prediction model that can tell the difference between regular connections and malicious intrusions or attacks. Auditing is performed on a standard data collection from this database, which comprises numerous simulated intrusions onto a military network. A total of 24 different attack types were used for training in the datasets, including another 14 used only for testing. Furthermore, the KDD Cup'99 dataset's total number of the records is 494,021 and also has 41 features.

Our proposed system will use data preprocessing techniques to remove redundant information from the dataset. Next, the Feature Important (FI) methodology will be used to choose the most relevant features from the dataset. Algorithm 1 shows the feature selection method in the proposed method. The model will then be trained using both a traditional and an ensemble classifier. Fig. 4 shows our experimental data analysis's performance evaluation matrices and learning curve.



(a)



(b)

Figure 4: Data analysis with (a) performance evaluation matrices (b) learning curve

Algorithm 1: Pseudo-code of the feature selection method

***Input*:** *Dataset*
***Output*:** *Performance Analysis*
***Initialization*:**
1. *C ← Data Cleaning*
2. *P ← Data Preprocessing*
3. *FI ← Feature Importance Method*
4. *FS ← Feature Selection Method*

***Start*:**
1. ***if*** *(NaN present)*
2. *Perform C to clean or handle the NaN values in the dataset.*
3. ***End if***
4. *Prepare the dataset for performance analysis of stroke risk prediction.*
5. ***Start:*** *Perform FI and get the important features.*
6. *Perform EC*

(Continued)

Algorithm 1 (continued)

Analysis of the experimental Results
End FI
Start: *Perform FS and get the important features.*
Perform EC
Analysis of the experimental Results
End FS
End:

Our work's confusion matrix evaluation is depicted in Figs. 5–7. In addition, we have integrated 5-fold cross-validation into our proposed approach. Moreover, we implemented *OneHotEncoder* because it allows us to select categorical features as a single numeric array in the proposed architecture.

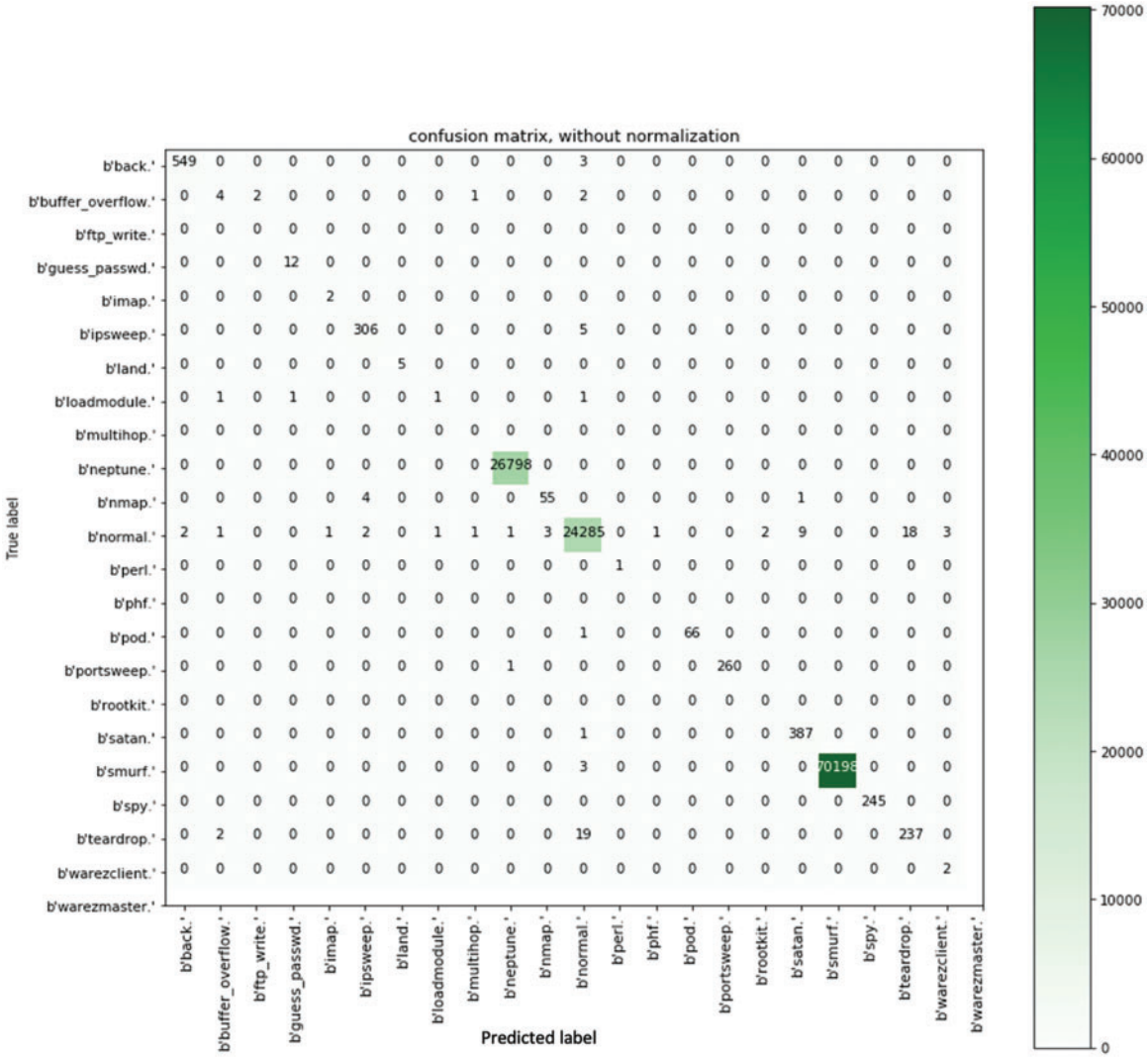


Figure 5: Confusion matrix of SVM algorithm

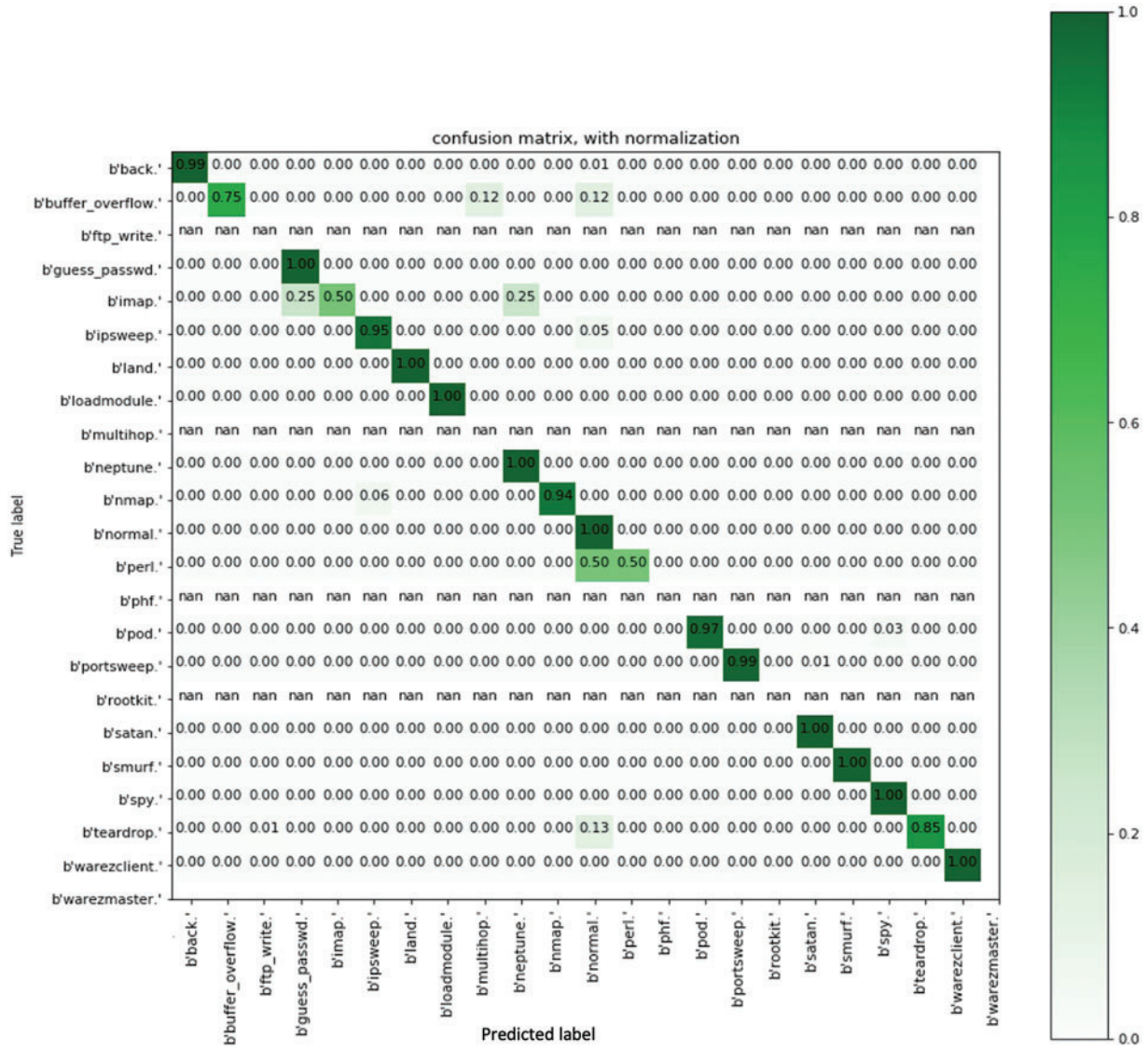


Figure 6: Confusion matrix of SVM algorithm with normalization

To reduce the dimensions of the features, our models also used Principal Component Analysis (PCA). All the targeted values or the good (normal) connection value ratio are shown in the Fig. 8.

Table 1 shows the experimental data analysis with six conventional classifiers. We have utilized Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), Naive Bayes (NB), K-Nearest Neighbor (KNN) and Logistics Regression (LR). We can determine why our system is superior by comparing the proposed system to the existing one. The proposed paper by the authors [14,22,27,30] used different types of machine learning algorithms or techniques, but they did not use any custom data structure for efficiently detecting intrusion. Again, though the authors of the paper [18] used a custom data structure with the Naive Bayes classifier, they did not mention the accuracy of detecting intrusion in databases. On the other hand, this proposed paper utilizes a custom data

structure and SVM classifier with the highest accuracy of 99.92%, indicating that the proposed system is more reliable and effective than others, as seen in Table 2.

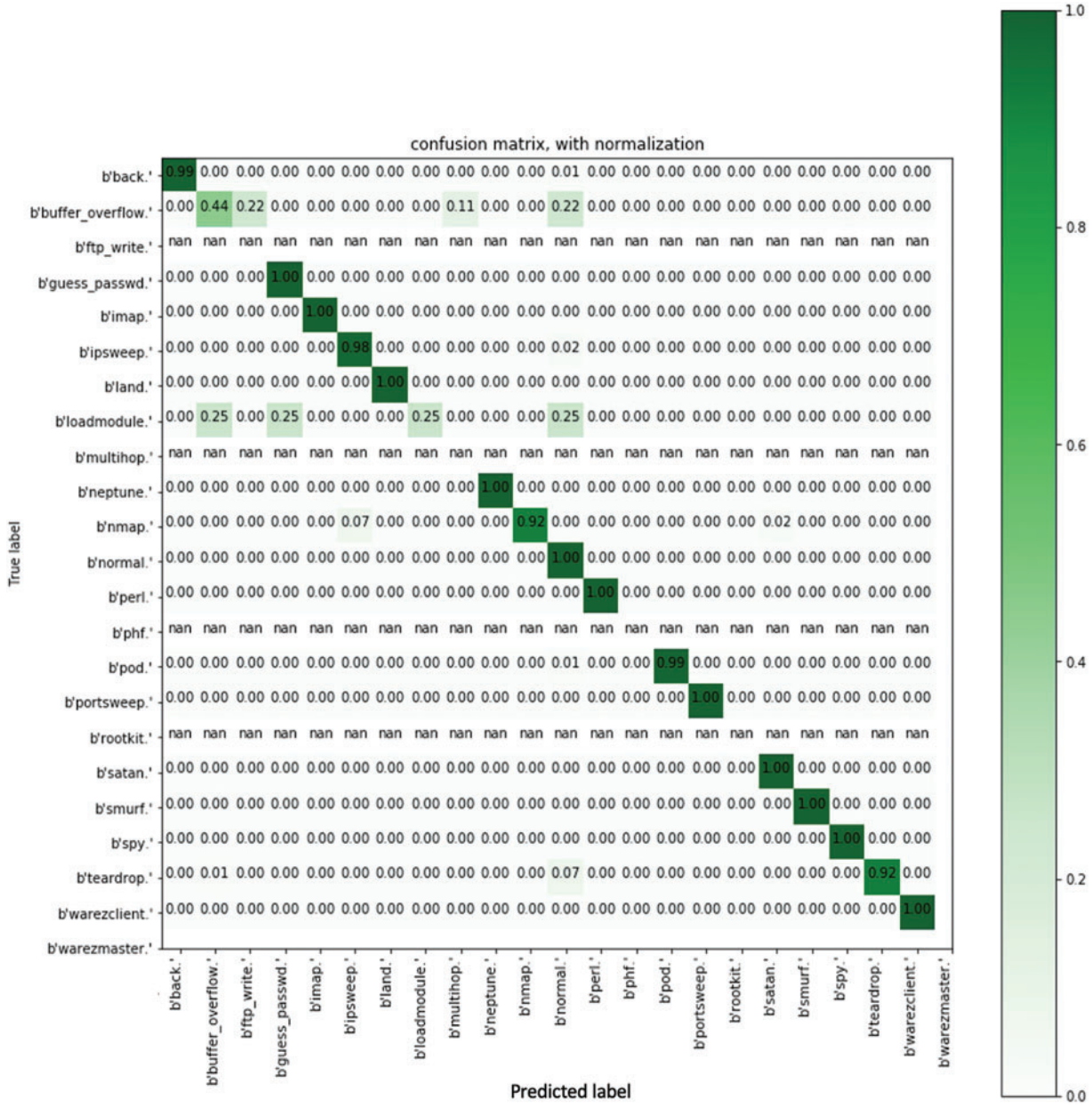


Figure 7: Confusion matrix of SVM algorithm + PCA + OneHotEncoder with normalization

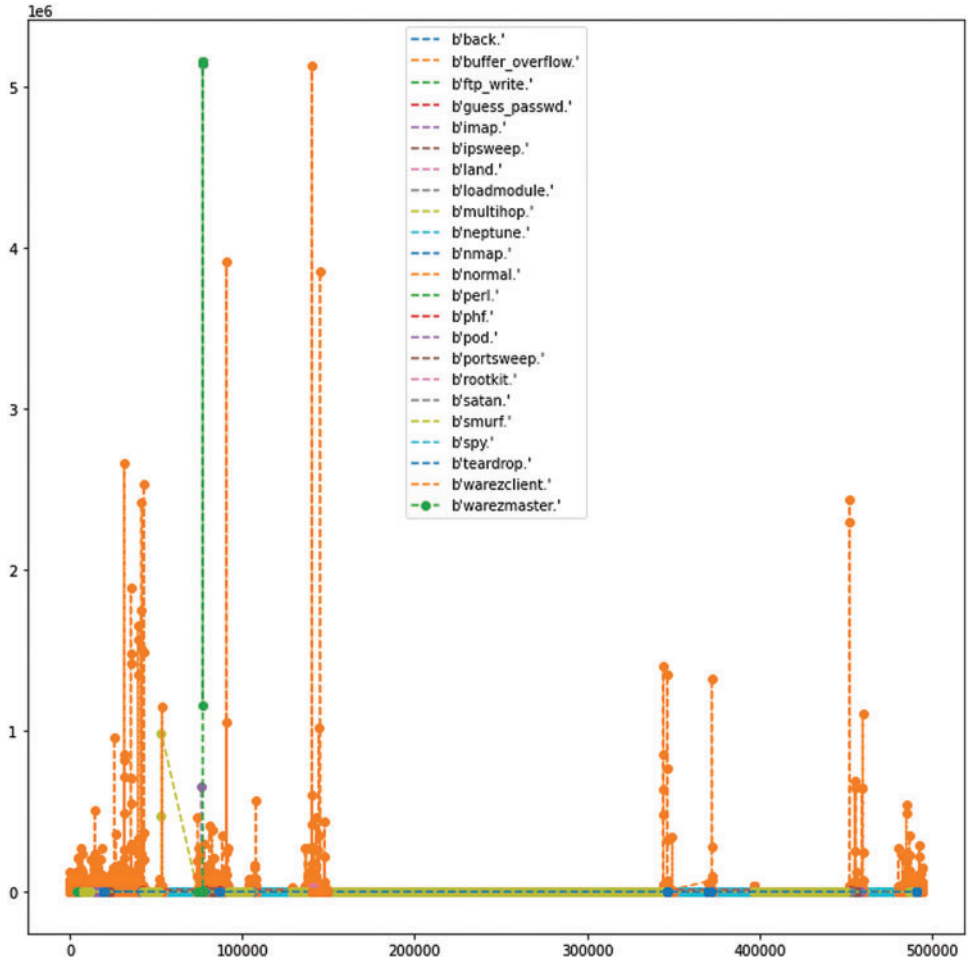


Figure 8: Targeted values or the good (normal) connection value ratio

Table 1: Experimental data analysis with seven classifiers

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
SVC	**99.92**	**99.91**	**99.93**	**99.92**
RF	97.06	97.27	97.00	97.05
DT	95.59	96.02	95.50	95.57
NB	97.06	97.27	97.00	97.05
KNC	95.59	96.02	95.50	95.57
LR	98.53	98.60	98.50	98.53

Table 2: A comparative analysis of proposed and existing systems

Ref. No.	ML algorithm/techniques	Accuracy	Custom data structure
[14]	SVM	48%	No
[22]	CN-LCS	94.64%	No
[27]	SVM	93.50%	No
[30]	DSDRGA	85.43%	No
[18]	NB	–	Yes
Proposed	SVM	99.85%	Yes
		99.92%	

5 Discussion

The suggested Intrusion Detection System (IDS) is a versatile and adaptable transaction-based system employed in Role Based Access Control (RBAC) databases. Even for databases with a large number of users, the system's usability is improved through the implementation of roles, which are used to train the classifier. Additionally, this method utilizes a novel data structure called Pentaplet. The technique offers the advantage of developing a new data structure, Pentaplet, which efficiently manages data storage, thereby enhancing performance despite dynamic changes in the structure and size of the database. The proposed architecture employs machine learning techniques for intrusion detection in the database. Specifically, the system utilizes the SVM algorithm with PCA and *OneHotEncoder* to detect abnormal transactions. The proposed framework achieves an accuracy of 99.85% in the SVM model and 99.92% in the SVM algorithm with PCA and *OneHotEncoder* model. Our method demonstrates significantly better accuracy compared to previous works [14,18,22,27,30]. The data structure used in this system is unique due to its dynamic length feature, which allows it to adapt to different needs. In contrast, the octraplet data structure used in previous work [18] has a fixed length, making it less storage efficient.

6 Conclusions

Protecting sensitive information is crucial in every operational information management. However, safeguards occasionally fail, allowing unauthorized individuals to gain access to private databases. Therefore, intrusion detection systems are utilized in the database to identify hostile actions within the Database Management Systems. In cases where other forms of protection are not feasible or easily exploitable, intrusion detection systems can play a vital role in database restoration [16]. This study proposes a transaction-based anomaly detection solution for RBAC databases, utilizing a Support Vector Machine (SVM) and a novel data structure called Pentaplet. The use of roles to train the classifier makes this method applicable to databases with a large number of users. The developed learning algorithm effectively detects role violations, and our suggested approach aims to reduce the number of false positives by considering support and confidence levels. Furthermore, this work includes a comparative analysis of existing DIDS methods and our proposed DIDS methods.

Future work involves incorporating additional machine learning techniques, such as nature-inspired algorithms like Particle Swarm Optimization (PSO), and expanding the dataset to identify the most accurate intrusion detection algorithms and evaluate attribute value dependencies to improve

the detection rate. Additionally, a method will be provided to assess the scalability of the proposed Pentaplet model through complexity analysis. Moreover, there are plans to integrate Graph Neural Networks (GNNs) into our proposed model in the future [35,36,37]. GNNs are a specialized category of artificial neural networks designed to effectively handle and analyze data with a graph-like structure. Graphs, consisting of interconnected nodes and edges, are used to represent relationships between entities. Nodes represent entities, while edges depict the connections between them. Graphs find applications in various domains, including social networks, chemical structures, and natural language texts. Therefore, it is anticipated that the integration of GNNs will enhance the complexity of our proposed models, particularly in the field of cybersecurity, such as intrusion detection. It is assumed that the inclusion of GNNs will enrich the capabilities of the proposed models, enabling effective detection of intrusions.

Acknowledgement: The author is thankful to the Dean of Scientific Research at Najran University.

Funding Statement: The author is thankful to the Dean of Scientific Research at Najran University for funding this work under the Research Groups Funding Program, Grant Code (NU/RG/SERC/12/6).

Author Contributions: Conceptualization, A. Alazeb; methodology, A. Alazeb; software, A. Alazeb; analysis and interpretation of results, A. Alazeb; writing original draft preparation, A. Alazeb.

Availability of Data and Materials: The data and materials used in this paper is available upon request from the corresponding author.

Conflicts of Interest: The author declare that they have no conflicts of interest to report regarding the present study.

References

[1] P. Bellini, P. Nesi and G. Pantaleo, "IoT-enabled smart cities: A review of concepts, frameworks and key technologies," *Applied Sciences*, vol. 12, no. 3, pp. 1607, 2022.

[2] Y. Otoum, D. Liu and A. Nayak, "DL-IDS: A deep learning-based intrusion detection framework for securing IoT," *Transactions on Emerging Telecommunications Technologies*, vol. 3, no. 33, pp. e3803, 2022.

[3] L. Vigoya, A. Pardal, D. Fernandez and V. Carneiro, "Application of machine learning algorithms for the validation of a new CoAP-IoT anomaly detection dataset," *Applied Sciences*, vol. 13, no. 7, pp. 4482, 2023.

[4] A. Alazeb, B. Panda, S. Almakdi and M. Alshehri, "Data integrity preservation schemes in smart healthcare systems that use fog computing distribution," *Electronics*, vol. 10, no. 11, pp. 1314, 2021.

[5] N. A. Perifanis and F. Kitsios, "Edge and fog computing business value streams through IoT solutions: A literature review for strategic implementation," *Information*, vol. 13, no. 9, pp. 427, 2022.

[6] D. Resul and M. Muhammad, "A review on fog computing: Issues, characteristics, challenges, and potential applications," *Telematics and Informatics Reports*, vol. 10, pp. 100049, 2023.

[7] H. Hamid, R. M. Noor and S. N. Omar, "IoT-based botnet attacks systematic mapping study of literature," *Scientometrics*, vol. 126, pp. 2759–2800, 2021.

[8] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[9] M. Z. Gunduz and R. Das, "Cyber-security on smart grid: Threats and potential solutions," *Computer Networks*, vol. 169, pp. 107094, 2020.

[10] Y. Hu and B. Panda, "Design and analysis of techniques for detection of malicious activities in database systems," *Journal of Network and Systems Management*, vol. 13, pp. 269–291, 2005.

[11] W. Wang, J. Li, N. Zhao and M. Liu, "MEM-TET: Improved triplet network for intrusion detection system," *Computers, Materials & Continua*, vol. 76, no. 1, pp. 471–487, 2023.
[12] A. Bhavani and V. Nithya, "Cryptographic algorithm for enhancing data security in wireless IoT sensor networks," *Intelligent Automation & Soft Computing*, vol. 36, no. 2, pp. 1381–1393, 2023.
[13] E. Jove, J. Aveleira-Mata, H. Alaiz-Moretón and J. L. Casteleiro-Roca, "Intelligent one-class classifiers for the development of an intrusion detection system: The MQTT case study," *Electronics*, vol. 11, no. 3, pp. 422, 2022.
[14] G. Yedukondalu, G. H. Bindu, J. Pavan, G. Venkatesh and A. SaiTeja, "Intrusion detection system framework using machine learning," in *Third Int. Conf. on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, pp. 1224–1230, 2021.
[15] B. D. Caulkins, J. Lee and M. Wang, "A dynamic data mining technique for intrusion detection systems," in *Proc. of the 43rd Annual Southeast Regional Conf.*, Melbourne, Florida, USA, vol. 2, pp. 148–153, 2005.
[16] K. Kurra, B. Panda and Y. Hu, "A multi-version database damage assessment model," in *Int. Workshop on Security in Information Systems*, Angers, France, vol. 2, pp. 100–108, 2013.
[17] M. Doroudian and H. R. Shahriari, "A hybrid approach for database intrusion detection at transaction and inter-transaction levels," in *2014 6th Conf. on Information and Knowledge Technology*, Shahrood, Iran, IEEE, pp. 1–6, 2014.
[18] S. Jayaprakash and K. Kandasamy, "Database intrusion detection system using machine learning," in *2018 Second Int. Conf. on Inventive Communication and Computational Technologies (ICICCT)*, Trichy, India, IEEE, pp. 1413–1416, 2018.
[19] M. Sohrabi, M. M. Javidi and S. Hashemi, "Detecting intrusion transactions in database systems: A novel approach," *Journal of Intelligent Information Systems*, vol. 42, pp. 619–644, 2014.
[20] L. Lyu, J. Jin, S. Rajasegarar, X. He and M. Palaniswami, "Fog-empowered anomaly detection in IoT using hyperellipsoidal clustering," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1174–1184, 2017.
[21] S. W. Kim, S. Park, J. I. Won and S. W. Kim, "Privacy preserving data mining of sequential patterns for network traffic data," *Information Sciences*, vol. 178, no. 3, pp. 694–713, 2008.
[22] S. J. Bu and S. B. Cho, "A hybrid system of deep learning and learning classifier system for database intrusion detection," in *Int. Conf. on Hybrid Artificial Intelligence Systems*, La Rioja, Spain, pp. 615–625, 2017.
[23] H. Mazzawi, G. Dalal, D. Rozenblatz, L. Ein-Dorx, M. Niniox *et al.,* "Anomaly detection in large databases using behavioral patterning," in *2017 IEEE 33rd Int. Conf. on Data Engineering (ICDE)*, San Diego, CA, USA, pp. 1140–1149, 2017.
[24] S. Panigrahi, S. Sural and A. K. Majumdar, "Detection of intrusive activity in databases by combining multiple evidences and belief update," in *2009 IEEE Symp. on Computational Intelligence in Cyber Security*, Nashville, TN, USA, pp. 83–90, 2009.
[25] B. Panda and A. Alazeb, "Securing database integrity in intelligent government systems that employ fog computing technology," in *2020 Int. Conf. on Computing and Data Science (CDS)*, Stanford, CA, USA, pp. 202–207, 2020.
[26] B. Sousa, N. Magaia and S. Silva, "An intelligent intrusion detection system for 5G-enabled Internet of Vehicles," *Electronics*, vol. 12, no. 8, pp. 1757, 2023.
[27] Y. Wang, "A hybrid intrusion detection system," Ph.D. dissertation, Iowa State University, USA, 2004.
[28] G. Edamadaka, C. S. Chowdary, M. J. Kumar and N. R. Sai, "Hybrid learning method to detect the malicious transactions in network data," *IOP Conference Series: Materials Science and Engineering*, vol. 981, no. 2, pp. 22032, 2020.
[29] K. Kurra, B. Panda, W. N. Li and Y. Hu, "An agent based approach to perform damage assessment and recovery efficiently after a cyber attack to ensure E-Government database security," in *2015 48th Hawaii Int. Conf. on System Sciences*, Kauai, HI, USA, pp. 2272–2279, 2015.
[30] I. Singh, S. Sareen and H. Ahuja, "Detection of malicious transactions in databases using dynamic sensitivity and weighted rule mining," in *Int. Conf. on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, Coimbatore, India, pp. 1–8, 2017.

[31] S. Kaddoura, R. A. Haraty, A. Zekri and M. Masud, "Tracking and repairing damaged healthcare databases using the matrix," *International Journal of Distributed Sensor Networks*, vol. 11, no. 11, pp. 914305, 2015.
[32] S. M. Darwish, "Machine learning approach to detect intruders in database based on hexplet data structure," *Journal of Electrical Systems and Information Technology*, vol. 3, no. 2, pp. 261–269, 2016.
[33] K. Yamanishi, J. I. Takeuchi, G. Williams and P. Milne, "On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms," in *ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Boston, MA, USA, pp. 320–324, 2000.
[34] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *2009 IEEE Symp. on Computational Intelligence for Security and Defense Applications*, Ottawa, ON, Canada, pp. 1–6, 2009.
[35] H. Zhu and J. Lu, "Graph-based intrusion detection system using general behavior learning," in *GLOBECOM 2022 IEEE Global Communications Conf.*, Rio de Janeiro, Brazil, pp. 2621–2626, 2022.
[36] W. Jiang, "Graph-based deep learning for communication networks: A survey," *Computer Communications*, vol. 185, pp. 40–54, 2022.
[37] T. Bilot, N. E. Madhoun, K. A. Agha and A. Zouaoui, "Graph neural networks for intrusion detection: A survey," in *IEEE Access*. vol. 11, pp. 49114–49139, 2023.