Machine Learning for Cloud Data Classification and Anomaly Intrusion Detection

Leila Megouache, Abdelhafid Zitouni, Salheddine Sadouni, Mahieddine Djoudi

Machine learning for Cloud data classification and anomaly intrusion detection

Leila Megouache[1*], Abdelhafid. Zitouni[2], Salheddine. Sadouni[3] , Mahieddine. Djoudi[4]

[1] Lire Laboratory, Computer Science Department,University of Constantine2-Abdelhamid Mehri, Algeria, 25000
[2] Lsiacio Laboratory, Constantine 1 Univeristy, Department of Geographical Sciences and Topography. Constantine, Algeria
[3] Techne, Université de Poitiers, TECHNE, Poitiers, FranceFrance

Corresponding Author Email: megouache_leila@yahoo.fr

Keywords:
Artificial intelligence, Extreme learning machine, Cloud, Intrusion detection system, Security, k-means clustering.

ABSTRACT

The sheer volume of applications, data and users working in the cloud creates an ecosystem far too large to protect against possible attacks. Several attack detection mechanisms have been proposed to minimize the risk of data loss backed up to the cloud. However, these techniques are not reliable enough to protect them; this is due to the reasons of scalability, distribution and resource limitations. As a result, Information Technology Security experts may feel powerless against the growing threats plaguing the cloud. For that, we provide a reliable way to detect attackers who want to break into cloud data. In our framework, we have no labels and no predefined classes on historical data, and we wish to identify similar models to form homogeneous groups from our observations. Then, we will use a k-means clustering algorithm to handle unlabeled data, and a combination approach of clustering and classification. We start with a k-means clustering algorithm for generating a labelled dataset from an unlabeled dataset. The labelled dataset is then used to train the extreme learning machine classifier, which will ultimately serve for intrusion detection. The proposed framework is eventually applied to the classic benchmark KDD99 dataset; the numerical results validate both the high accuracy and the time-saving benefit of the proposed approach.
.

1. INTRODUCTION

Throughout Due to the growing development of the Internet of Things (IoT) and digitization, various security incidents such as unauthorized access [1] [2] and malware attack [3] have grown at an exponential rate in recent years.

Cloud computing is now widely adopted and used by large companies to take advantage of the delivery of applications, infrastructures and high storage capacity on the Internet. In a cloud computing environment, multiple users can access a single server to retrieve and update their data without purchasing any licenses for different kinds of applications but also can make more extensive use of cloud computing in one work life [4]. But, as the volume of data and the complexity of cloud operations increase, defining an effective security infrastructure becomes of paramount importance. For example, suppose an attack has occurred at the cloud level, all cloud resources will be permanently affected, and the quality of service will decrease. Therefore, the data protection of all cloud users is damaged. For this, cloud service providers must protect their resources to maintain the quality of resources [5].

Although several solutions exist with adequate security measures for cloud applications, they are still insufficient compared to the speed of threats that emerge every day, and the spammers who keep on inspecting our operations. In addition, as cloud operations is shared between different actors, the interoperability factor also becomes a critical requirement [6].For these reasons, we introduce machine learning for its speed and performance.

Machine learning (ML) allows computers to learn without being explicitly programmed [7]. It is a significantly large and growing field of artificial intelligence. Its purpose is to facilitate human tasks through its speed and automatic reasoning. Insecurity, machine learning is based on data analysis to find patterns. So, that we can better detect malware in encrypted traffic, find internal threats, predict where the "bad neighborhoods" are online to keep users safe while browsing or protect data in the cloud by learning about suspicious user behavior [8]. In machine learning security we often talk about three main types of attacks: poisoning, evasion and inference. In the case of poisoning, an attacker seeks to bias the behaviour of a model by modifying training data. We can take the well-known example of Microsoft Tay, a chatbot designed to interact on social networks with young Americans. It ended up appropriating the vocabulary of its speakers. With evasion, an attacker plays on the input data of the application to obtain a decision different from the one normally expected. And finally, in the inference case, an attacker successively tests different requests on the application to study its behavior [9].There are currently several use cases of ML in the field of cyber security, such as fraud detection, vulnerability detection from predictive models, intrusion detection, static analyzes and the detection of infiltration of data. Doyen Sahoo [3] presented an overview about machine learning techniques utilized in malware URLs detection and categorize feature representation and learning algorithm development in this domain [10]. The learning machine extreme (ELM) is an emerging technology that overcomes some challenges faced by older learning machines, such as latency in processing time and responses to data processing. Otherwise, there are several machine learning algorithms that have been proposed as IDS models such as Support vector machine (SVM) [11] and Artificial neural network (ANN) [12], ELM [13] [14] [15]. ELM provides better generalization performance at a

much faster learning speed and with the least human intervention [16] [17].

Until now, most of existing solutions are based on the "legitimate" and "malicious" connection types connected to an IoT network. In a public wireless network, the attack detection system detects an attack by providing a detection model based on the reputation and trust given to each node [16]. The reputation of each node is compared to a threshold, which determines if the node is good or if it is considered as an attacker. Or the SVELTE system [18] which is a prototype of an intrusion detection system for the Conkiti operating system, Proposed by Raza. It includes a distributed mini firewall to respond to alerts. SVELTE is a hybrid system which has à centralized components and a distributed components between the nodes. It is installed both on the nodes and on the router which links the internal zone of the objects and the rest of the Internet. It is designed primarily to detect attacks on routing protocols.

Considering the existing works in this field to ensure the quality of cloud services such as [19] [20]. The big question we always ask ourselves is: Why in many countries, many big companies still afraid to back up their data in the cloud and away from home, lest it is leaked, lost or damaged? So our objective in this proposed work is to minimize the risk of intrusion at the cloud level by using probability laws and the K-means clustering algorithm for data segmentation and also to know how to use classification techniques to categorize the different attacks that may occur. Intrusion detection by the classification method only is increasingly used. But building a system based on classification and clusters will certainly improve intrusion detection techniques. The KDD 1999 intrusion detection dataset will play a role in our key to solving the problem and is the most widely used by researchers working in the security field. Then our contribution in this paper is to create a framework based on a combination of clustering and classifier. Firstly k-means clustering is used to create a labeled dataset from an unlabeled dataset. The labeled dataset is using to train the ELM classifier that is eventually using to detect intrusion. The experiments with the KDD99 dataset show a high quality of intrusion detection. The organization of this paper is summarized as follows: Firstly, we briefly discuss the concept of data security and the most relevant methods to solve these problems through intelligent decision-making in a distributed environment. We also make a brief discussion of different machine learning tasks in security. Second, we propose an extensible methodology to model user behavior from contextual information. Behaviors follow a probabilistic procedure to filter out malicious operations. Finally, we try to improve data security by combining clustering and classification methods. The results of this method are we brought more precision in the filtering data stored in the cloud, minimized the risk of losing sensitive data and provided a good quality system to our customers.

This paper is presented as follows. The related works are discussed in Section 2. The preliminaries are given in Section 3. The proposed scheme is explained in Section 4, while the result and discussion are introduced in Section 5.finally, the conclusions are given in Section 6.

2. RELATED WORKS

The Due to the very rapid development of IoT, many researchers have proposed their approaches to detecting security attacks in the cloud. Until now, ELM remains an important research topic due to its high efficiency, easy implementation, unification, classification, and regression. By implementing these approaches, we can effectively detect spammers [1], making it a powerful tool for combating unwanted and malicious activities.

In [20], this work integrates different machine learning algorithms: Vector Machine, Naive Bayes and Random Forest support for classification. And was performed on a cloud environment using "Tor Hammer" as an attack tool,but this solution has not shown much efficiency.

In [21],in this article, the authors propose a new firewall system called the Enhanced Intrusion Detection and Classification (EIDC) system for a secure cloud computing environment. EIDC detects and classifies received traffic packets using a new combination technique called the most frequent decision where nodes 11. The past decisions are combined with the current decision of the machine learning algorithm to estimate the final classification of attack categories [22]. This strategy increases the learning performance and the accuracy of the system.

In ‘Using Machine Learning to Secure IoT Systems’, Canedo and Skjellum [23] propose using machine learning within an IoT gateway to help secure the system. The proposal was to use the Machine Learning technique, specifically Artificial Neural Network (ANN) in the gateway and application layers. In the gateway to monitor subsystem components and, in the application, layer to monitor the state of the entire system. After setting up the system with training data and warming it up, the researchers manipulated the sensors to add invalid data for 10 minutes. When invalid data is running on the system, the neural network will be able to detect the differences between valid and invalid data. Later a delay between transmissions was added as a third input to simulate man-in-the-middle attacks. To predict whether the data was valid or invalid for the approximately 360 samples in the test set and summarized that using ANN is very beneficial in making an IoT system more secure.

In [24], the authors present a machine learning program that attempts to maintain privacy across multiple data providers. The proposed system allows all users of the system to verify the accuracy of encrypted data. A one-way proxy re-encryption (UPRE) scheme is used for reducing high computational costs with multiple data providers. The cloud server embeds noise into the encrypted data, allowing analytics to apply machine learning techniques and keep information confidential from cloud providers [25].

In [26], the method proposed in this article based on ML-ELM does not rely on manual feature extraction and selection. Instead, the crude current signals obtained by IKB are sent directly to the network for identification, which highly reduces the complexity of system design. The ML-ELM model first performs layer-by-layer unsupervised learning through the ELM-AE of each hidden layer. Then supervised learning with labels is performed using the traditional ELM algorithm to classify.

In [27], the authors propose an automated approach to cyber security management in IoT systems. This approach provided an autonomous or semi-autonomous (without human intervention) means of early detection and response to cyber-attacks. The authors demonstrate that the proposed

solution can protect against unknown attacks in a web application.

In [28], a hybrid intrusion detection system based on machine learning was proposed. The authors combined support vector machine (SVM) and genetic algorithm (GA) methodologies with a fitness function developed to assess the accuracy of their system. In their scenario, an SVM was used using different values of hyper parameters of the kernel, gamma and degree functions. Their results showed that the proposed model provides symmetry between information security and attack detection and intrusion.

In summary, existing approaches to detecting attacks in a cloud environment generally focus on the design and development of an intrusion detection system at the entrance to the cloud and generally rely on the existence of a single centralized cloud. However, there are many requirements to consider, such as resource limitation, distribution, and system scalability. For this, we believe that it is necessary and imperative to design and develop new approaches that support all aspects of security and present and future attacks.

3. BACKGROUND

3.1 Cloud computing

The Cloud computing is the provision of IT services (servers, storage, databases, software, network management, artificial intelligence) via the Internet [28] [15]. To offer faster and more innovative use, flexible resources and profit at a very high cost and productivity compared to traditional methods. Moreover, there are several types of clouds' which do not necessarily have the same structures and are different in their design and development.

Several models, types and services have evolved to help provide the best solution to our needs. There are three different ways to deploy cloud services for this: on a public cloud, a private cloud or a hybrid cloud [27] [29].

Public cloud:Public clouds are owned and operated by third-party cloud service providers, who provide their computing resources, such as servers and storage, over the Internet.

Private cloud: A private cloud refers to cloud computing resources used exclusively by a single business or organization [30].

Hybrid cloud: Hybrid clouds' combine public and private clouds, linked by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives large flexibility [31].

One of the challenges of cloud development is the emergence of various consumer security issues. Machine Learning (ML) is one of the means used today to secure the cloud. ML techniques are used in a variety of ways to prevent or detect attacks and security breaches in the cloud [30].

.

3.2 Overview of ELM

In many cloud-level security issues, extreme machine learning offers a new way to solve these problems. The Extreme Learning Machine (ELM) is a new machine learning model proposed by Huang [32].It is based on least squares Layered Neural Networks (SLFNN). Nowadays, ELM is an important research topic due to its high efficiency, easy implementation, unification, classification, and regression. And could therefore be implemented in field of detection of social spammers [33].

In this section, we will briefly discuss the basis of ELM. The ELM algorithm can be summarized as follows in 3 steps:

- **Step 1**: Definition of hidden layer node number $\tilde{N}$, randomly assign input weights ***ai*** and hidden layer biases ***bi, (i* = 1, 2, . . .,Ã).**
- **Step 2**: Calculate the hidden layer output matrix ***H***.
- **Step 3**: Calculate the output weight ***β.***

https://arxiv.org/pdf/1409.3924

The simple ELM learning algorithm has a model of the form:

$$\hat{Y} = X2\ \sigma\ (X1\ n)$$

Where **X1** is the matrix of input-to-hidden layer weights, σ is an activation function, and **X2** is the matrix of hidden-layer-to-output weights. The algorithm works as follows:

1. Complete $\mathbf{X_1}$ with Gaussian random noise.

2. Estimate $\mathbf{X_2}$ by the least squares method to match the response matrix of the variables **Y**, use using the pseudo in· +, giving a design matrix **T**:

$$\mathbf{X_2} = \sigma\ (\mathbf{X_1}\ \mathbf{T})^{+}\ \mathbf{Y}$$

ELM algorithm can be explained as follows: Given N arbitrary distinct samples **(xi, *ti*** $)\in \boldsymbol{R}n \times \boldsymbol{R}m$ where ***xi*** is the input sample and ***ti*** is the output sample (label), the output of a SLFN with L hidden nodes and activation function $g(x)$ are calculated as:

$$Oj = \sum_{i=1}^{N} \beta i\ Li = 1\ (wi \cdot xj + bi)(j = 1,2,\cdots, N) \quad (1)$$

Or: $\sum_{i=1}^{N} \beta iL(wi \cdot xj + bi) = tj,\ j = 1,\ldots,N$

Therefore, the computational construction of the ELM algorithm in this study is shown in Figure. 1[31]

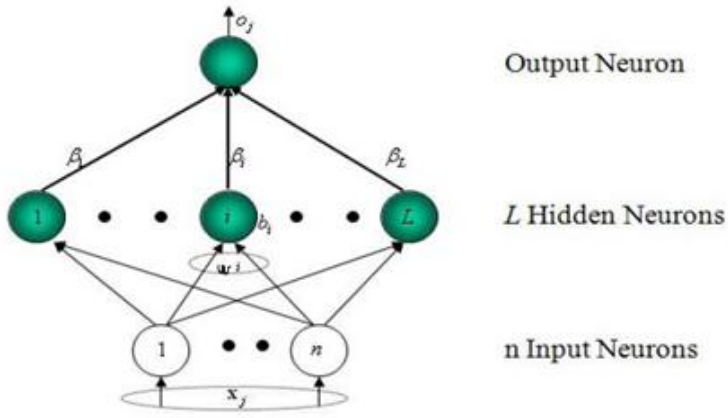


Figure. 1 Neurons network

Where $wi = (wi1, wi2, \ldots, win)$ T, are weights connecting the input and hidden layer, ***bi*** **(*i* = 1,2, ⋯ , *L*)** is the bias of the i -th hidden node, and ***βi*** **= (*βi*1, *βi*2, …, *βim*)*T*** is the output weights between the hidden layer and the output layer.

The nonlinear classification problem can be transformed into the following linear classification problem:

$H\beta = T$ **(2)**

Where $H = \{h\}$ $(i = 1, \ldots, L and j = 1, \ldots, N)$ is the hidden-layer output matrix.

$hij = (wi \cdot xj + bi)$ denotes the output of the Ith hidden neuron concerning xj, $T = [t1, t2, \ldots, tm]$ T is the target matrix (classification labels).

Where **wi** is the weight vector between the input layer and hidden layer, **βi** is the weight vector between the hidden layer and output layer, **bi** is the bias of the I, the hidden node, and L (·) is the activation function of the hidden layer. The node parameters **wi** and **bi** of the hidden layer are randomly assigned. And therefore, only the number of hidden layer nodes l needs to be determined in the ELM model. If the error between the output **Oi** and the target **t** can be approximated to zero, then the following equation can be obtained by:

$$\sum_{j=1}^{n} \| tj - 0j \| = 0 \quad \textbf{(3)}$$

$$H_0(w_1,\ldots,w_L, x_1,\ldots,x_L, b_1,\ldots,b_N) = \begin{pmatrix} L(w_1, x_1, b_1) \cdots L(w_L, x_L, b_1) \\ \cdot \quad\quad\quad \cdot \\ \cdot \quad\quad\quad \cdot \\ L(w_1,x_1,b_N)\ldots L(w_L, x_L, b_N) \end{pmatrix} \quad \textbf{(4)}$$

β is output weight, $\beta = \begin{pmatrix} \beta^T{}_1 \\ ., \\ \beta^T{}_L. \end{pmatrix}_{L\times m}$ and $T = \begin{pmatrix} T_1{}^t \\ . \\ T^t{}_N \end{pmatrix}_{Nxm}$

In most cases, the number of hidden nodes is much smaller than the number of training samples. Namely **($L \ll N$),** with a total of ***L***neurons in the hidden layer [34].

The minimum norm least-square (**LS**) solution to the linear problem (**2**) is: $\hat{\beta} = H^{+} . T$

Where $H+$ is the Moore-Penrose generalized inverse of matrix Has analyzed in [32], ELM using such Moore-Penrose (**MP**) inverse method tends to obtain good generalization performance with highly increased learning speed.

Algorithm 1: Extreme Learning Machine

Input: Number of training samples *n* where
$\{(x\,j, t\,j) / x\,j \in R^n, t\,j \in R^m, and\ j = 1, 2, \ldots, n\}$.
2-Activation function *gi (x)*, and number of hidden nodes *L*.
Output:
3-Step 1: Input weight ai and bias *bi* are initialized randomly, $i = 1, 2, \ldots, L$.
4-Step 2: Hidden layer outputs matrix H is calculated.
5- Step 3: Output weight matrix $\beta$is computed as follows:
$\beta = H\dagger T$,
where $T \in R^{n\times m}$

4. PROPOSED METHODOLOGY

Considering the different existing methods and approaches and their limitations in the detection of attacks in the cloud, we propose at this work a robust framework to efficiently perform attack detection in the cloud environment. Among the existing security attacks, we are interested in network security attacks. For example, the goal of an attacker is to launch an indiscriminate integrity attack that induces high false positive and true negative rates of classifiers, or to launch [35] a targeted privacy violation attack that illegally obtains sensitive data of the targeted user **[**36**].**

Thus, our main task is to maximize the security in the cloud and minimize the risk of data loss. Finally, if a malevolent user attempts to attack the system, it will be immediately stopped. Also, the cloud service provider must provide access only to authenticated users in its database. To verify user authenticity, CSP checks their trust values. If the user's trust value is greater than the threshold value, the user is considered a genuine user. Note that the user trust value depends on their behavior parameters in the cloud [37]. ELM is a learning algorithm for the single hidden layer feed-forward neural networks used in classification and regression. It has a simple and more valid mode, compared to the traditional BP algorithm and is more convenient than the traditional ANN model. Therefore, the learning speed of ELM is much faster than that of BP. ELM will provide a direct solution to the problem and tends to reach not only the smallest training error. But also, the smallest norm

The proposed method is named attacker detection in Cloud, based on the supervised learning (SL) approach. To filter attackers, all data (text, document, and figure) will be tagged. This process is called document markup.

Figure 2 schematizes our approach. The database will be built from the pre-existing data on the cloud. This data is fragmented into multiple subsets, and then extreme machine learning will be run to make predictions and decisions on each subset of data. Combining the results for each ELM helps distinguish legitimate users from non-legitimate users.

Compared to other solutions, such as, fog computing (FC) and mobile edge computing (MEC) (FC/MEC). The solution for ELM is simple and can be found by finding the minimum standard of a problem of least squares, which can finally be transformed into a generalized Moore-Penrose inverse problem involving a matrix [33].

4.1 Dataset construction in Cloud

We will try to gather pre-existing data at the cloud level, which are no classified into non-legitimate users and legitimate users [38] [39]. Unlabeled data collection is the data set containing the most relevant characteristics of multiple cloud user behaviors. However, for the construction of the dataset, the cloud API is used to collect a real dataset from public information. Here we are using K-means clustering which is a type of unsupervised learning used when data is unlabeled [40]. In this step, it suffices to create groups of data represented by the variable K. The algorithm works iteratively to assign each data point to one of the K groups according to the similarity of the characteristics and functionalities

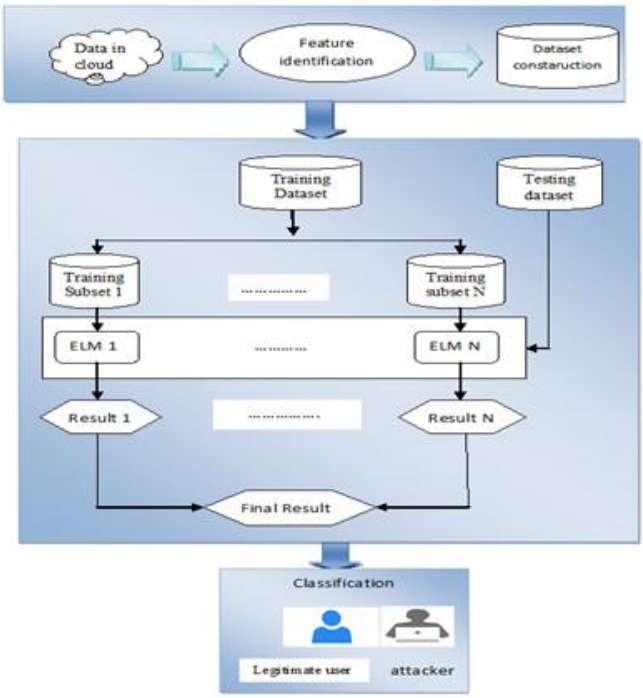
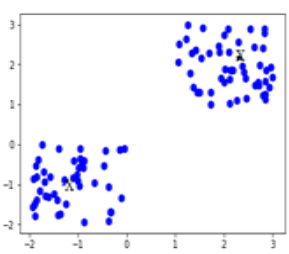


Figure 2. Organization flow of the proposed framework

provided [41]. Anomaly detection based on user behavior is useful, such as the number of times they log in, the history of these movements and all these activities on the cloud will be evaluated. Next, it is necessary to separate valid and monitored activity groups if a data point moves from one group to another; this should be used to detect significant changes in the data. We summarize our approach as follows:

A first selection of data is created to determine spammers and legitimate users at the level of the cloud network [28]. For that, legitimate users is select from the most active clients in the cloud, for example, users who only work in the cloud.

And non-legitimate users isselected from the set of users who were too often involved in malicious activity example, users who share malicious URLs or messages or, who direct to malicious links, and fictitious websites. Then we generate a list of all users (attacker and legitimate users) by exploring the list of subscribed clients. For this a web crawler is used as in [42]. In addition, each user's behavior is tagged. Then two groups of users were created (figure 3 and 4) who are, Legitimate and non-legitimate users.



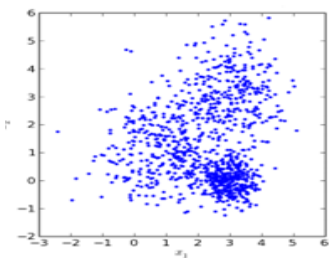


Figure.4 Clustered data **Figure.3** Original unclustered data

Figure 5 shows the difference in proportion between the original data sent by the normal user and the attacker. Most legitimate users deal with private or public data and share information through the cloud with their friends. But at the same time, most attackers steal and spy on other people's data. Here, we have taken a set of random data, that is stored in the cloud. And taking into account the following parameters: their behavior, the size, the number of executions of this data and the execution time. Thus two groups of data are formed from its parameters which are legitimate users and malicious users.

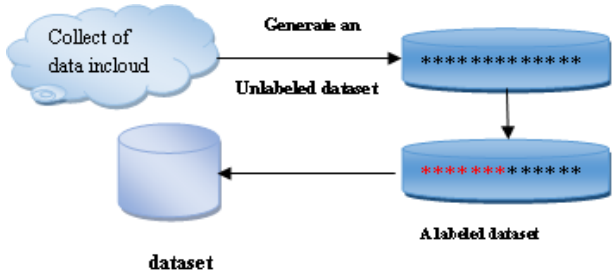


Figure 5. Dataset construction.

The process followed by K-Means Clustering is as follows at fig. 6 [43]:

Step 1- Select the value of "K" which determines the number of clusters.

Step 2- Choose random k points or a centroid to form the cluster. Here we can choose any data point.

Step 3- Assign all the data points to their nearest cluster. Let us use the distance method based on the correlation.

Step 4- Calculate the centroids of the clusters by taking the average of all the data points that belong to each cluster.

4.2 Training and testing phase

In the test phase for the classification of data and the construction of ELM, initially, we will study a limited amount of data, if the result obtained is satisfactory, we will apply the procedure to a large amount of data by the application of the normal law of probability [44]. Each ELM in our work works as follows:

P:the probability of an event occurring.

q: the inverse probability of p.

X: is the number of times an event occurs.

N: the number of experiences.

If **P** is the probability of an event occurring during a malware detection experiment. And if **q = 1- p** is the probability that it does not occur (probability of success), then the probability that this event occurs X times in N experience (i.e. X detection of an attack and N - X no detection) is given by the binomial coefficients [41]:

$$P(X) = \frac{N!}{X!(N-X)!} p^X q^{N-X} = C_N^X p^X q^{N-X}$$

Or X = 0,1, 2, 3,…, N and N! = N(N -1) (N-2) ….1.

And 1, C_N^1, C_N^2 C_N^3 … C_N^X ;

But when the number of data becomes very important, it will extend towards the normal law to carry out our test phase. To test our learning machine, we have developed the following test:

We want to determine that our machine was **90%** efficient at testing a large amount of data in just **1 minute.** Either in a **200-megabyte** data sample, we have validated **160 megabytes** of correct data, or now we determine if our machine learning is effective. The solution is to let **P** be the probability of obtaining the correct data; we must then decide on the two following hypotheses (H):

α : low values.

N: amounts of data
q: is the level of significance which is taken at 0.1
H_0: P = 0.9,and our statement is correct.
H_1: P <0.9,and our statement is false.

We will test for low values of α because we want to know if the proportion of data is too low. If the significance level is taken at 0.1, that is, if the area is grayed, as in the figure. 6, which is equal to 0.1, then α = -2.33. The following decision rule is therefore used:
If: **H_0 is true, μ = NP = 200 (0.9) = 180**
And $\delta = \sqrt{NPq} = \sqrt{(200)(0.9)(0.1)} = 4{,}23$. Then, in reduced centered units:(160 -180)/ 4,23 = -4,73. The value significantly lowers than -2.33, show in figure 6. Therefore, we conclude that our assertion is justified and that the results are very satisfactory.

K-means cluster Algorithm:

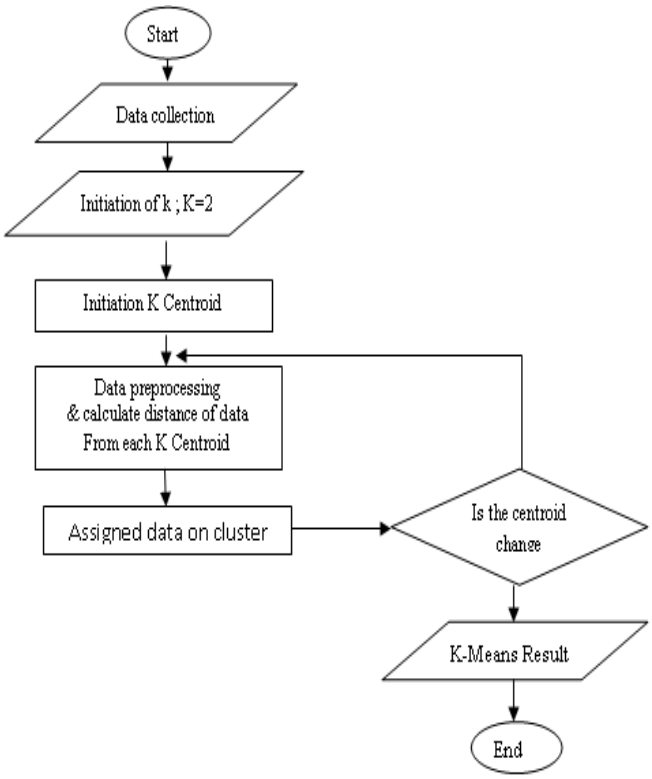


Figure 6.K-means Cluster Algorithm.

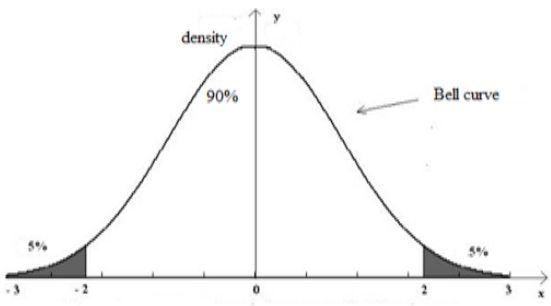


Figure 7. Testing evaluation graph.

The shaded regions **(α)** are critical areas, as shown in the graph in (figure 7).

As has been analyzed in several works, the attack must be performed by injecting erroneous data samples into the training phase to affect the resulting decision function. Purity of the training data and the improvement of the robustness of learning algorithms [45] are two main counter measures that we must take into accounttowards any opponent during the training phase (figure 8).

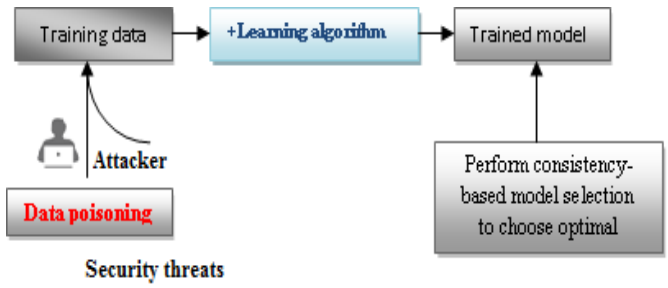


Figure 8. Defensive techniques of machine learning.

Training data that participate in the training phase play an important role in developing a high-performance machine learning model. In general, opponents target training data, resulting in a decrease in the overall performance of the machine learning model (fig.7). For example, a poisoning attack is a typical type of security threat against the training phase [46].

In our framework (figure 9), the training dataset is divided into **K** subsets. Each subset contains the same number of samples and **p-input** features. However, the ELM presents shortcomings for training of big data; like time consumption which is a process of calculating the output matrix of hidden nodes, Moore-Penrose the generalized inverse of a matrix, the Laplace matrix, and matrix multiplications take a long time when forming large-scale datasets.

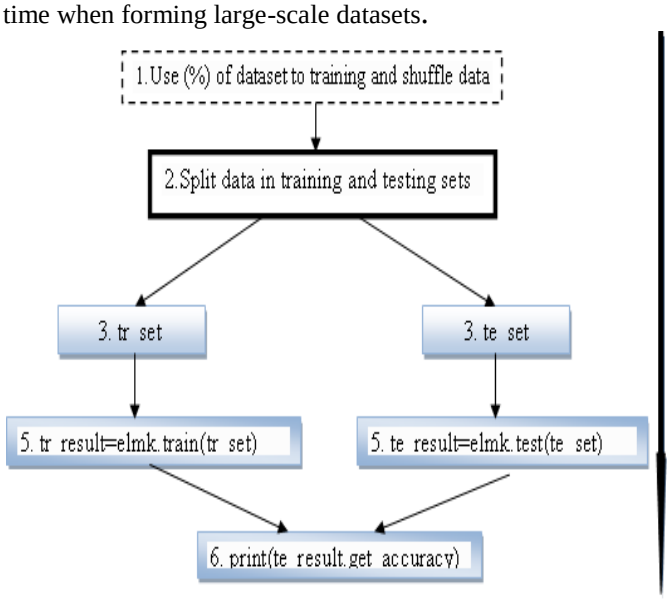


Figure 9. Proposed methodology

The testing and training phase determines the reliability of our machine learning. The results of the training phase in fig.8 who is **tr_result = elmk.train(tr_set),** and the test

phase, **te_result = elmk.test(te_set)** will be combined to obtain at the end, a classification which allows the distinction between legitimate and non-legitimate users by: **print(te_result.get_accuracy).**

Algorithm

1. // training phase

1- initialize training database with **N** samples **(Ai, Yi)**
2- Randomly initialize **W** and biases $\mathbf{x_j}$, **j=,2,…l**
3- Calculate the output weight matrix $\mathbf{T_i}$

4- Calculate**T= Hβ** where $\mathbf{H_0}$=

$$\begin{pmatrix} L(w_1, x_1, b_1) \cdots & L(w_L, x_L, b_1) \\ L(w_1, x_1, b_N) & \cdots L(w_L, x_L, b_N) \end{pmatrix}$$

$$\beta = \begin{pmatrix} \beta^{T_1} \\ . \\ . \\ \beta^{T_L} \end{pmatrix}_{L\times m} \text{ and } \mathbf{T=} \begin{pmatrix} T_1^{\,t} \\ . \\ . \\ T^{t}_{N} \end{pmatrix}_{N\times m}$$

5- Calculate β= H* Y_{all}, where$\mathbf{Y_{all}}$ =(Y1, Y2…, YN)

2. // Detection

a) For each sample **i** calculate **Ti** = $\sum_{i=1}^{N} \mathbf{\beta i\ Li} = \mathbf{1}\ (\mathbf{wi}, \mathbf{xj}, \boldsymbol{bi})(\mathbf{j} = \mathbf{1}, \mathbf{2}, \cdots, \mathbf{N})$
b) Map $\mathbf{T_i}$ **to** $\mathbf{Y_i}$
c)If **Yi** represent attack
Then alert
Else
Remain silent

When the value is calculated by the equation β = H* Y_{all}, that means that, our ELM is trained and ready to detect attacks. It calculates the output for each sample using the equation T= H β. If the planned release represents an attack,it will generate an alert to the cloud administrator. Otherwise, she remains silent. All data will be tagged to pass the malicious user filtering process. This process is called document labelling. We notice that most powerful filtering methods that exist today, there is no automatic way that can be used to assign labels to a large amount of data.

5. EXPERIMENTAL RESULT

With the development of digital technology, and the number of data circulating on the net in very strong growth, at the same time, the security threats on the networks have increased. Considerably nowadays, it is, therefore, necessary to develop more powerful systems to ensure this security. In this article, we explore the capabilities of our ELM on intrusion detection using the KDD Cup 1999 dataset [47]. In addition, the robustness of the state Preserving Extreme Learning Machine (SPELM) is evaluated using a dimensionality reduction technique such as the main component Analysis (PCA) [48]. To evaluate the effectiveness of the experiment results, we consider these metrics:

True positive (TP): represents the number of spammers correctly classified,

False negative (FN): refers to the number of spammers misclassified as non-spammers.

False positive (FP): expresses the number of non-spammers misclassified as spammers.

True negative (TN): is the number of non-spammers classified correctly.

(1) True positive (TP)$TP = \frac{TP}{TP+FN}$ X 100.

(2) False negative (FN) $FN = \frac{FN}{FN+TP}$ X 100.

(3) True negative (TN): $TN = \frac{TN}{TN+FP}$ X 100.

According to the confusion matrix, a set of metrics commonly evaluated in the machine learning field are introduced, including [49]:

Precision (*P*), recall (*R*) and *F*-measure (*F*).

P is the ratio of number of instances correctly classified to the total number of instances and is expressed by the formula:

In the following experiments, we evaluate ELM, Regularized Extreme Learning Machine (RELM), SPELM and support vector machine (SVM) for the detection of malicious user intrusions on the cloud platform.The 1999 KDD Cup dataset is used for intrusion detection. All of our experiments were conducted on a desktop computer, computer with an Intel @ Core i5 Duo CPU E86 @ 3.33 GHz processor and 4 GB of RAM to estimate processing time in MATLAB (R2013a).

5.1 Data set description

The task of the classifier learning competition organized in conjunction with the KDD'99 conference was to learn a predictive model (i.e. a classifier) capable of distinguishing between legitimate and illegitimate connections in a computer network [50] [51]. In KDD Cup 1999, the training set contains a total of four attack categories. In this experiment, we use 30,000 normalized and coded digital data samples. For training and testing. Table 1 shows a confusion matrix obtained by ELM classifiers. It shows that our proposed solution is quite efficient, with 99.2% of non-legitimate users and 99.8% of legitimate users ranked correctly, leaving only a small fraction of non-legitimate users and misclassified legitimate users.

Table 1. Score of legitimate and nonlegitimate user's

	legitimate	non-legitimate
legitimate	99,8	0,2
non-legitimate	0.8	99,2

Furthermore, we compare training and testing time betweenADCELM, RELM, SPELM and SVM. The experiences have been carried out several times to have calculated the mean value of each phase. Regarding the values the test and training phase we observe that our model takes a total of 0.0630 seconds for the test and 0.4374seconds to practice training classification, the experiment results are illustrated in table 2. The results indicate that our ELM is much faster than SVM, SPELM or RELM and is therefore more efficient.

Table 2. Comparison between ELM, RELM, SPELM and SVM

Classifier	Training time (s)	Testing time (s)
Our ELM	0.4374	0.0630
RELM	0,8091	0,1105
SPELM	1,622	0,0721
SVM	3.031	0.501

We compare also training and testing time between ELM, RELM, SPELM and SVM. The experiment results that we have obtained; are illustrated in table 3. The results in figure 10 indicate that our ELM is much faster than other solutions and is, therefore more efficient.

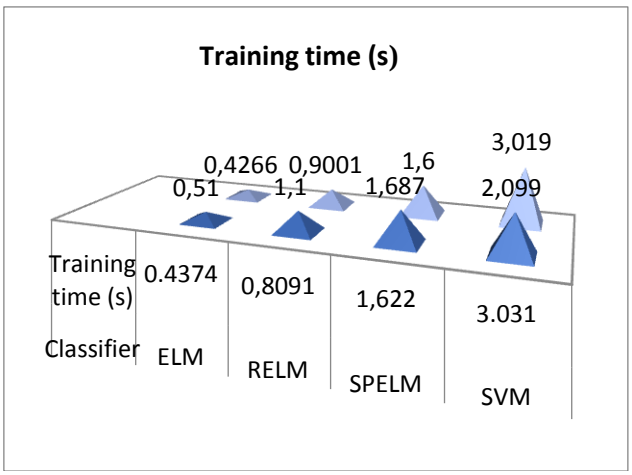


Figure 10. Training time (s) graph.

In the test phase, we calculate the flow time of the operation compared to the other approaches mentioned above. We have redone the calculation in four (4) different periods. And the results are displayed by the following diagram in figure11.

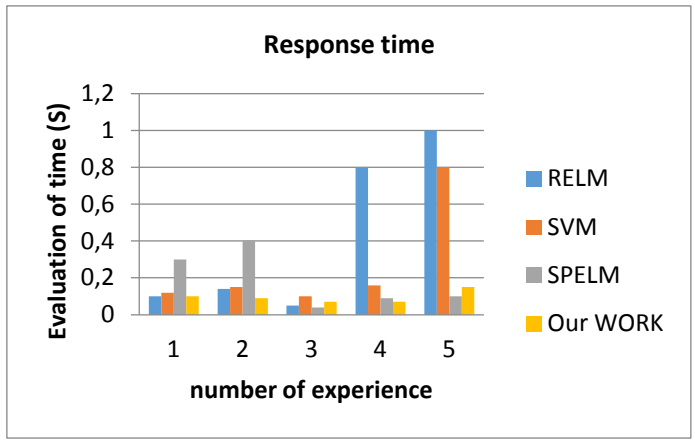


Figure 11. Testing time's graph.

Table 3 Testing accuracy comparison with ELM and RELM without feature dimensionality reduction.

# of training samples in %	SPELM (%)	RELM (%)	Proposed ELM (%)
20	97.52	97.81	97.96
30	98.00	97.86	98.10
40	97.89	97.99	98.02

Figure 11 demonstrates the stability of our system compared to others.

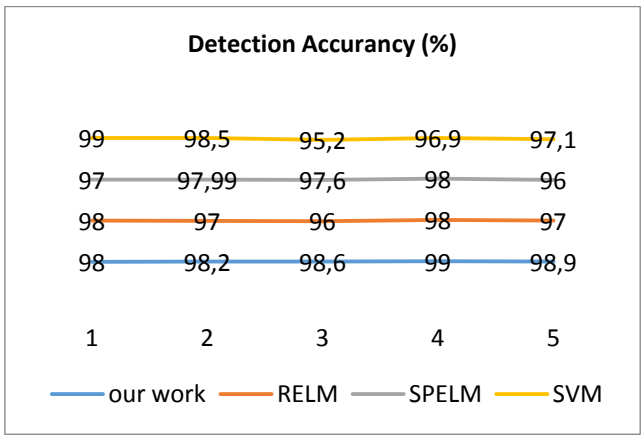


Figure 12. Detection Accuracy Graph

5.2 Discussion

In this article, we have proposed a framework to secure data in Cloud, here we have no labels data, no predefined classes or the Centroid. A k-means clustering algorithm for handling untagged data are used,andan Extreme Learning Machine (ELM) algorithm are applied. The use of ELM by applying the law of least squares to solve the intrusion detection problem in the cloud network.The main advantage of this solutionis the reduced training time and offers a good scalability. However, we have tried to increase the accuracy compared to SVM techniques [52] [53]. All the solutions and

methods have been suggested; for the implementation of a reliable intrusion detection system, had participated in the minimization of loss of control over cloud data. If we can detect at least more than 95% of attack connections and filter them out, we can prevent the attacker from overwhelming the cloud server. In the detection of DDoS attacks [54] [55] , for example, where attackers install malicious program on the network of vulnerable hosts, and controls managers and robots using a command and control mechanism [56] [57] [58].

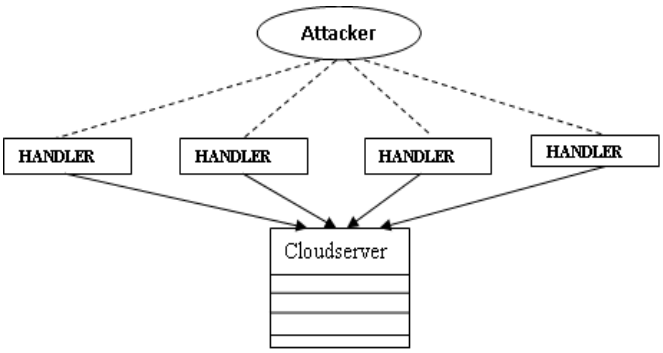


Figure 12. Distributed denial of service attack.

In this case, it is necessary to install an attack detection module between the cloud server and the handler, based on ELM. Comparison of the detection accuracy of our work with other proposed works, are shown in the tables above. Our work is performing well compared to others. However, we need more training time to develop the method of classification as others works.

We also note that false-positive type alerts could generate lot of noise, which sometimes annoys employers in the sector. Let's imagine that more than 200 processes report a false positive alert every 37 seconds. That would require that more than 200 people must sent on-site to detect if there are an anomaly. It seems to us that predicting anomalies using supervised machine learning is the best solution to avoid any potential disaster. And it will be great if we install a system in place to send a signal to the control center in the event of an anomaly. That will help us prevent and stop a devaster problem as quickly as possible before they spread to other linked processes.

Also, in this work, we based ourselves on operational technology (OT) before that of IT technology, with the objective that any application or process developed must be available and used first before being secure. And the machine learning is the best solution in this field. And it cannot in any way be replaced by a human solution.

6. CONCLUSIONS

In this article, we presented a new approach to bring optimal and reliable security to cloud computing based on ELM techniques. It should be noted that ELM, not only improves characteristics related to classification algorithms but also solves the problem of detecting intrusions in the cloud network reliably and efficiently. And this is what has increased its use in many areas nowadays. We proposed a model that can be formed and tested in a very short period, and it can detect attacks with high accuracy. The results demonstrate the effectiveness and efficiency of the proposed method; they show that the proposed method can be applied to larger applications which require both real-time performance and high precision. In our next work, we will use a hybrid method of clustering for better performance instead of a single method.

REFERENCES

[1] Zheng, X., Zhang, X., Yu, Y ., Kechadi, T., Rong, C., (2016).: ELM-based spammer detection in social networks. The Journal of Supercomputing Volume 72 Issue 8August 2016 pp 2991–3005 https://doi.org/10.1007/s11227-015-1437-5- Springer.

[2] Rizal, R. (2020) : Commonwealth Law Bulletin, Maliciousunauthorised access to computer programs and data in Malaysia, Pages 453-461, 2020 vol 47, 2021, Issue 3.

[3] Anidu, A., Obuzor, Z. (2022): Evaluation of Machine Learning Algorithms on Internet of Things (IoT) Malware Opcodes. In: Choo KK.R., Dehghantanha A. (eds) Handbook of Big Data Analytics and Forensics. Springer, Cham. https://doi.org/10.1007/978-3-030-74753-4_12.

[4] Sanchati, R, (2011).: Cloud Computing in Digital and University Libraries. Global Journal of Computer Science and Technology. The journal of Global journal of computer science and technology, volume 11, issue 12, version 1.0, ISSN 0975-4172.

[5] Khilar, P.M., Chaudhari, V., Swain, R.R. (2019): Trust-Based Access Control in Cloud Computing Using Machine Learning. In: Das H., Barik R., Dubey H., Roy D. (eds) Cloud Computing for Geospatial Big Data Analytics. Studies in Big Data, vol 49. Springer, Cham, https://doi.org/10.1007/978-3-030-03359-0_3 .

[6] Babiceanu, R ., Seker, R (2019): Cyber resilience protection for industrial internet of things: A software-defined networking approach. The journal of Computers in Industry, volume 104, pp 47-58, – Elsevier. https://doi.org/10.1016/j.compind.2018.10.004.

[7] Andrew, S., Zhaohao, S., (2022).: Handbook of Research on Foundations and Applications of Intelligent Business Analytics. pages 17. DOI: 10.4018/978-1-7998-9016-4.ch005

[8] machine-learning-security. Accessed on 25/03/2023. Available :http://www.cisco.com/c/en/us/products/security/.html#~how-ml-helps-security.

[9] Xavier, b.: https://www.journaldunet.com/solutions/dsi/1495415-how to-secure - learning machine /. (2020)

[10] Guan, Z., Bian, L., Shang, T., Liu, J. (2018) ,: When machine learning meets security issues: A survey - 2018 IEEE International Conference on Intelligence and Safety for Robotics (ISR) 24_27 Aug

[11] James, G., Witten, D., Hastie,T., Tibshirani, R. (2021).: Support Vector Machines. In: An Introduction to Statistical Learning. Springer Texts in Statistics. Springer, New York, NY https://doi.org/10.1007/978-1-0716-1418-1_9.

[12] Chen, W. H., Hsu, S. H., Shen, H. P., (2005).:"Application of SVM and ANN for intrusion detection," Journal of Computer &. Operations Research., vol. 32, no. 10, pp. 2617–2634.

[13] Fossaceca, J. M., Mazzuchi, T. A., Sarkani, S., (2015). : "Expert Systems with Applications MARK-ELM : Application of a novel Multiple Kernel Learning framework for improving the robustness of Network Intrusion Detection, The journal of Expert Syst. Appl., vol. 42, no. 8, pp. 4062–4080.

[14] Ali, M. H., Zolkipli, M. F., Mohammed, M. A., Jaber, M.M., (2017): "Enhance of extreme learning machine-genetic algorithm hybrid based on intrusion detection system, The journal of Eng. Appl. Sci., vol. 12, no. 16, pp. 4180–4185.

[15] Mohammed, H. A., Mustafa, M.A., (2021).Comparison Between Extreme Learning Machine and Fast Learning Network Based on Intrusion Detection System EasyChair Preprint, N5103 .

[16] Yuan,L., Guang, B,H., Zongben ,X. (2013).:Dynamic Exreme learning machine and Its Approximation_Capability. IEEE Transactions on Cybernetics 43(6) OI:10.1109/TCYB.2013.2239987

[17] Huang, G.B., Wang, D.H., Lan, Y., (2011),: Extreme learning machines: a survey . The journalof machine learning and cybernetics 2:107–122 DOI 10.1007/s13042-011-0019-y– Springer

[18] Wang, T., Wang, P., Cai, S., Zheng, X., Ma, Y., Jia, W., (2021).: Mobile edge-enabled trust evaluation for the Internet of Things. The journal if: Information Fusion, Volume 75, PP 90-1002021, Elsevier https://doi.org/10.1016/j.inffus.2021.04.007.

[19] Raza, S., Wallgren, L., Voigt, T., (2013). : SVELTE: Real-time intrusion detection in the Internet of Things. The journal of: Ad hoc networks, PP 2661-26742013 – Elsevierhttps://doi.org/10.1016/j.adhoc.2013.04.014.

[20] Pooja Rana, Isha Batra, Arun Malik, Agbotiname Lucky Imoize,Yongsung Kim, Subhendu Kumar Pani, Nitin Goyal, Arun Kumar, Seungmin Rho, (2022) Intrusion Detection Systems in Cloud Computing Paradigm: Analysis and Overview Volume 2022. | Article ID 3999039 | https://doi.org/10.1155/2022/3999039.

[21] Nabeel H. Al-A'araji, Safaa O. Al-Mamory, Ali H. Al-Shakarchi, (2021):Classification and Clustering Based Ensemble Techniques for Intrusion Detection Systems: A Survey. Journal of Physics: Conference Series, DOI 10.1088/1742-6596/1818/1/012106

[22] Alzubair, H., Rafik, H., Hong,Y., Ping, L.: (2019). An Efficient Outsourced Privacy Preserving Machine Learning Scheme With Public Verifiability. IEEE access pp(99).. DOI:10.1109/ACCESS.2019.2946202

[23] Wani, A.R., Rana, Q.P., Saxena, U (2019).: Amity International Analysis and Detection of DDoS Attacks on Cloud Computing Environment using Machine Learning Techniques. Amity International Conference on Artificial Intelligence (AICAI). DOI: 10.1109/AICAI45948.

[24] Chkirbene, Z ., Erbad, A., Hamila, R., (2019) : IEEE Wireless, A Combined Decision for Secure Cloud Computing Based on Machine Learning and Past Information, IEEE Wireless Communications and Networking Conference (WCNC), DOI: 10.1109/WCNC44850.

[25] Chkirbene, Z., Erbad, A., Hamila, R., 2019: A Combined Decision for Secure Cloud Computing Based on Machine Learning and Past Information. IEEE Wireless Communications and Networking Conference DOI: 10.1109/WCNC44850.2019 (2019).

[26] Muhammad, N.K., Asha, R., Seyit, C., (2021): "Lightweight Cryptographic Protocols for IoT-Constrained Devices: A Survey", Internet of Things Journal IEEE, vol. 8, no. 6, pp. 4132-4156.

[27] Pan, Y., Naixue, X., Jingli, R., ,(2020)."Data Security and Privacy Protection for Cloud Storage: A Survey", Access IEEE, vol. 8, pp. 131723-131740

[28] Guangquan, Z., Zhiyi, W., Yongcheng, G., Guangxing, N., Zhong, L. W., Bin, Z. (2020):Multi-Layer Extreme Learning Machine-Based Keystroke Dynamics Identification for Intelligent Keyboard Senior Member, IEEE Sensors journal, volume: 21 issue :2.

[29] https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-are-private-public-hybrid-clouds/ Accessed 17 (2023).

[30] Cloud Computing - Overview and Examples Connected Services and Cloud Computing School of Electrical Engineering and Informatics SEEI / STEI Institut Teknologi Bandung ITB Update April (2017).

[31] Johan, S. R.., Jesus, M.T.P (2021).:Framework-based security measures for Internet of Thing: A literature review", Open Computer Science, vol. 11, pp. 346,

[32] Ammar, A., Faisal, A., (2021). : Effective Intrusion Detection System to Secure Data in Cloud Using Machine Learning. https://doi.org/10.3390/sym13122306.

[33] Xingshuo, A., Xianwei, Zhou., Xing, L., Fuhong, L., Lei, Y. (2018).: Sample Selected Extreme Learning Machine Based Intrusion Detection in Fog Computing and MEC. Wireless Communications and Mobile Computing Volume 2018, Article ID 7472095, 10 pages https://doi.org/10.1155/2018/7472095

[34] Qiang, L., Pan, L., Wentao, Z., Wei, C.,Shui, Y., VICTOR, C. (2018).: A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data Driven View . IEEE ACCESS. Digital Object Identifier 10.1109/ACCESS.2018.2805680

[35] Megouache, L., Zitouni, A., Djoudi, M., (2020).: Ensuring user authentication and data integrity in multi-cloud environment. Human-centric Computing and Information Sciences, 10 (1), ff10.1186/s13673-020-00224-yff. ffhal-03125583. Springer.

[36] Ali, B., Manar, A., Qassim, N., Halah, A., (2021).: Machine Learning for Cloud Security: A Systematic Review. Published in: IEEE Access (Volume: 9) ISSN: 2169-3536, DOI: 10.1109/ACCESS.2021.3054129 , PP 20717 – 20735.

[37] Satyakam, B., Pradyut, K.B., (2017).: Implementation of activation functions for ELM based classifiers, Wireless Communications Signal Processing and Networking (WiSPNET) International Conference on, pp. 1038-1042.

[38] Zheng, X., Zhang, X.., Yu, Y., (2016).: ELM-based spammer detection in social networks. J Supercomput 72, 2991–3005. https://doi.org/10.1007/s11227-015-1437-5.

[39] Xingshuo, A., Xianwei, Z., Xing, L., Fuhong, L., Lei, Y., (2018). Sample Selected Extreme Learning Machine Based Intrusion Detection in Fog Computing and MEC; The journal of Wireless Communications and Mobile Computing WILEY; Article ID 7472095, https://doi.org/10.1155/2018/7472095.

[40] Ramy, E., Hassan, A., Amr, B., (2020): A Hybrid Nested Genetic-Fuzzy Algorithm Framework for Intrusion Detection and Attacks, Access IEEE, vol. 8, pp. 98218-98233,.

[41] Pabitr, M. K., Vijay, C., Rakesh, R. S (2018).: Trust-Based Access Control in Cloud Computing Using Machine Learning. Cloud Computing for Geospatial Big Data Analytics p 55-79 Springer. https://link.springer.com/chapter/10.1007/978-3-030-03359-0_3.

[42] Lele, C., Wenbing, H., Fuchun, S., (2015).: A Deep and Stable Extreme Learning Approach for Classification and Regression, In book: Proceedings of ELM-Volume 1DOI: 10.1007/978-3-319-14063-6_13.

[43] Ahuja, L(2018).: Handling web spamming using logic approach. In: International conference on advances in computing and data sciences. Springer, Singapore, pp 380–387.

[44] Gaurav, D., Tiwari, S.M., Goyal, A. (2020): Machine intelligence-based algorithms for spam filtering on document labeling. Soft Comput 24. , Springer 9625–9638 https://doi.org/10.1007/s00500-019-04473-7.

[45] Koloveas, Thanasis, C., Sofia, A., Spiros, S., Christos, T., (2021).: A Machine Learning-Based Framework for Gathering and Leveraging Web Data to Cyber-Threat Intelligence. Journal: Electronics,Volume 10, Number 7, Page 818. DOI: 10.3390/electronics10070818.

[46] Introduction-to-k-means-clustering. Accessed on 25/03/2023 Available: https://blogs.oracle.com/ai-and-datascience/post/.

[47] Mariya, P., Antony, S., Dhandapani, S. (2023):detection/overview IPFS based storage Authentication and access control model with optimization enabled deep learning for intrusion detection. Advances in Engineering Software. Volume 176,

[48] Murray R.S. Book of: Serie Schaum, theorie and application of statistic, Rensselaer Institute (1982).

[49] Christopher, D. O., David, E.C., Matthew, P.T (2017): An empirical machine learning method for predicting potential fire control locations for pre-fire planning and operational fire management International Journal of Wildland Fire 26(7) 587-597 https://doi.org/10.1071/WF16135.

[50] Qiang, L., Pan, L., Wentao, Z., Wei, C., Shui, Y., Victor, C. M. (2018).: A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data Driven View . IEEE Access, Digital Object Identifier 10.1109/ACCESS.2018.2805680.

[51] Abhishek, D., Meet, P., Vaibhav, S., Rudra, M., (2018): Benchmarking datasets for Anomaly-based Network Intrusion Detection: KDD CUP 99 alternatives, 978-1-5386-6227-4/18/$31.00 c IEEE conference.

[52] Thippa, R.G., Praveen, K.R.M..,Kirova, L., Rajesh, K., Dharmendra, S. R, Gautam, S., (2020): Analysis of Dimensionality Reduction Techniques on Big Data, in IEEE Access, vol. 8, pp. 54776-54788, doi: 10.1109/ACCESS.2020.2980942.

[53] Mirza, A. H. (2018): Computer network intrusion detection using various classifiers and ensemble learning," 26th Signal Processing and Communications Applications Conference (SIU), pp. 1-4, doi: 10.1109/SIU.2018.8404704.

[54] Navid, B., Ebrahim, B., Hamid, R. B., Amin, H., Mohsen, S., Zhengyu, L., Mehdi, S., (2022): Locating high-impedance faults in DC microgrid clusters using support vector machines. Journal of Applied Energy, volume 308, 118338.ELSEVIER..

[55] Behal, S ., Kumar, K., (2017).: Detection of DDoS attacks and flash events using novel information theory metrics, journal of Computer Networks, volume 116 page 96-110. Elsevier.

[56] Gopal, S. K., Syed, T.A., (2019). : Distributed denial of service attacks detection in cloud computing using extreme learning machine. Int. J. Communication Networks and Distributed Systems, Vol. 23, No. 3.

[57] Bontupalli, V., Hasan, R., Taha T.M (2014).: Power efficient architecture for network intrusion detection system. In: NAECON 2014-IEEE National aerospace and electronics conference, IEEE, pp 250–254.

[58] Alom, M.Z., Sidike, P ., Taha, T.M ., Asari, V.K. : State Preserving Extreme Learning Machine: A Monotonically Increasing Learning Approach, Journal of Neural Processing Letters 45:703–725 DOI 10.1007/s11063-016-9552-8. (2017).