International Journal of Cloud Applications and Computing
Volume 6 • Issue 4 • October-December 2016

A Cloud Intrusion Detection System Using Novel PRFCM Clustering and KNN Based Dempster-Shafer Rule

Partha Ghosh, Netaji Subhash Engineering College, Maulana Abul Kalam Azad University of Technology, Kolkata, India
Shivam Shakti, Netaji Subhash Engineering College, Maulana Abul Kalam Azad University of Technology, Kolkata, India
Santanu Phadikar, Maulana Abul Kalam Azad University of Technology, Kolkata, India

ABSTRACT

Cloud computing has established a new horizon in the field of Information Technology. Due to the large number of users and extensive utilization, the Cloud computing paradigm attracts intruders who exploit its vulnerabilities. To secure the Cloud environment from such intruders an Intrusion Detection System (IDS) is required. In this paper the authors have proposed an anomaly based IDS which classifies an incoming connection by taking the deviation of it from the normal behaviors. The proposed method uses a novel Penalty Reward based Fuzzy C-Means (PRFCM) clustering algorithm to generate a rule set and the best rule set is extracted from it using a modified approach for KNN algorithm. This best rule set is used in evidential reasoning of Dempster Shafer Theory for classification. The IDS has been trained and tested with NSL-KDD dataset for performance evaluation. The results prove the proposed IDS to be highly efficient and reliable.

KEYWORDS

Anomaly Detection, Cloud Computing, DST (Dempster-Shafer Theory), FCM (Fuzzy C-Means) Clustering, IDS (Intrusion Detection System), KNN, NSL-KDD Dataset, PRFCM (Penalty Reward Based FCM) Clustering

1. INTRODUCTION

For the last decade, Internet has turned out to be an inseparable part of daily human life. With growing number of users there is a need for robust services for development and deployment of software as well as exchange of data. The advent of Cloud computing has given more dimensions to the developers as well as to the users. In basic terms, Cloud computing is the phrase used to describe different scenarios in which computing resource is delivered as a hosted service over the Internet. There are three fundamental types of services offered by the Cloud Service Providers (CSP) - Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) (Subashini & Kavitha, 2011). The Cloud infrastructure makes use of integrated technologies, standard Internet protocols and virtualization techniques. The bugs and vulnerabilities in these technologies render Cloud vulnerable to intrusion as well as traditional attacks (Modi et al., 2013). To protect the Cloud environment from intruders an effective and efficient Intrusion Detection System (IDS) is required. An IDS is deployed in the Cloud network to classify the incoming connections as normal or attack. An intrusion tries to exploit the confidentiality, integrity and availability of resources (Liao, Lin, Lin et al., 2013). There are two major techniques for intrusion detection- Anomaly Detection and Misuse Detection (Patel,

DOI: 10.4018/IJCAC.2016100102

Taghavi, Bakhtiyari, & Júnior, 2013). In case of Anomaly detection, it is a behavior based detection system that defines and characterizes normal behavior of the system. Whenever action deviates from the expected behavior, it is considered as an Anomaly. Therefore, it can detect unknown or novel attacks (Govindarajan & Chandrasekaran, 2011). But since the normal behavior of user differs, the rate of false alarm is high (Özyer, Alhajj, & Barker, 2007). Whereas, Misuse Detection technique is knowledge based detection system where predefined rules or signature of attacks are already formed and that can be used to determine an incoming attack by pattern matching of known attack. Misuse Detection based IDS have higher accuracy than the Anomaly Detection based IDS (Jamdagni, Tan, He, Nanda, & Liu, 2013). However, unknown and variation of known attacks cannot be identified by misuse detection (Ghosh, Mandal, & Kumar, 2015). In this paper the authors have proposed an Anomaly based Intrusion Detection System. Here they use a novel Penalty Reward based Fuzzy C-Means (PRFCM) clustering algorithm to train the IDS which performs better than FCM clustering algorithm. Further, a modified approach for K-Nearest Neighbor (KNN) and Dempster-Shafer Theory (DST) is used in order to classify an incoming connection. Rest of the paper is organized as follows: Section 2 surveys related work in IDS. Section 3 provides a preliminary theory for the proposed system. Section 4 gives a detail of the sample dataset used in the experiment. Section 5 lays out the proposed model. Section 6 and 7 presents the result and conclusion respectively.

2. RELATED WORK

The Cloud environment is vulnerable to many types of attack. To encompass the whole range of attacks, a number of Intrusion Detection Systems are proposed. Lombardi and Pietro (2010) discussed in their paper about the security concerning the Cloud. These security concerns are high owing to its size and complexity of service. A novel architecture for Intrusion Detection System based on virtualization is proposed in their work which proved to be effective in performance. Denning (1987) summarized the model of Intrusion Detection System in which intrusions were identified using the knowledge based on simulated attacks. Although this method was efficient but the system failed to recognize intrusions which varied from the expert knowledge. In order to address the security concerns in Cloud environment several data analysis algorithms were proposed. Lee and Stolfo (2000) outlined a framework for IDS models based on data mining algorithms. Their proposed model used association rule and frequent episode algorithms to mine frequent patterns from audit data which can be used for anomaly detection models. The results show that their model performed as well as the best systems built on manual knowledge approaches. Song and Ma (2009) further proposed the use of data mining in IDS. Their proposed model used K-means algorithm to form clusters of normal and abnormal class. Although the K-means algorithm had simplicity and less complexity but a number of clusters for normal class were mismarked which made it unfit for the job. Rao, Damodaram and Charyulu (2012) proposed Modified and Hashed K-means algorithm for clustering in IDS. Their algorithm overcame the drawback of noise and outliers in K-means algorithm but the model suffered from increase in time complexity. A comparison of FCM and K-means algorithm was performed by Nadiammai and Hemalatha (2012). The results show that FCM clustering outperformed K-means clustering in terms of accuracy and speed. Khazaee and Rad (2013) proposed Fuzzy C-Means clustering for improving the performance of IDS. They performed clustering on 14 reduced features using Chi-Squared feature evaluation to sample the dataset. Their results on KDDCup99 dataset suggest that sampling of data using the FCM clustering algorithm yields a better result than other systems without preprocessing. One of the most common classifiers is the K-Nearest Neighbor algorithm. It uses a distance based selection of K closest instances to classify a given sample. The

use of KNN classifier in IDS was proposed by Y. Liao and Vemuri (2002) using 1998 DARPA BSM audit data to detect the intrusions. One of most remarkable aspect of KNN classifier is low training time as compared to other classifiers. Their results show that a low false positive rate can be achieved. However, their result was said to become less significant for more sophisticated data set. Assigning equal weightage to all the neighboring connections for classification did not seem to be a valid approach. Dudani (1976) proposed to assign a weight to the given nearest connections based on its distance from the sample. In his paper, he proved that the distance weighted rule has lower probability error than the majority rule. This decrease in probability error can be attributed to lower number of ties in distance weighted rule. Keller, Gray and Givens (1985), suggested a fuzzy KNN algorithm to assign importance according to the distinctive nature of labeled samples. They proposed three methods for assigning membership values to each connection using fuzzy KNN decision rule and fuzzy prototype decision rule. Their results show low error rate and a significant decrease in the number of misclassified instances which have high membership to incorrect class. This suggests that their approach can be modified to yield higher accuracy. An IDS was proposed based on Evolutionary Algorithm (EA) using KNN by Govindarajan and Chandrasekaran (2009), which showed that use of other algorithms with KNN tends to decrease false alarm rate and increase efficiency. Their results suggest the use of hybrid KNN algorithm to increase classification accuracy or decrease run time with suitable tradeoff. Su (2011) proposed the use of clustering to improve KNN-based classifier. In their paper, they used clustering to sample the dataset and reduce the training time. They also used genetic algorithm to select an optimal weight vector for weighted KNN classifier. Their results show that the above method can be utilized to increase the classification accuracy and reduce the training time. Artificial Intelligence (AI) in anomaly detection was proposed by the use of evidential reasoning to deal with uncertainty in IDS by Esmaili (1997). It was shown that the evidential reasoning can be used to detect abnormalities in user behavior more effectively. They proposed the use of Dempster-Shafer Theory (DST) to represent and propagate uncertainty in the system. Chen and Venkataramanan (2005) proposed the use of DST to combine evidences for intrusion detection in Mobile Ad Hoc Networks. The DST was compared to Bayesian theory and it had more practical advantage as it required no priori or conditional probabilities. However, the initial assignment of mass in DST was a difficult challenge. The use of KNN rule based on Dempster-Shafer Theory was proposed by Denoeux (1995). In their model the K-nearest neighbors of data were chosen to be items of evidence, upon which DST was applied to combine all the uncertainties for a given hypothesis. Their approach also used DST as a tool to assess the reliability of the resulting classification. In this paper the authors have proposed a new Intrusion Detection System by assessing the benefits and drawbacks of the above works. In the following section, they lay a brief background for their proposed model.

3. PRELIMINARY THEORY

3.1. Fuzzy C-Means Clustering

Clustering is defined as a process of partitioning of dataset into partitions or clusters in a way that every element of the same cluster is as similar as possible. The conventional Hard Clustering algorithm restricts that each data point should only belong to one cluster. However, Fuzzy Set theory proposed by Zadeh (1965) suggested a new clustering algorithm called the Fuzzy C-Means (FCM) algorithm. In FCM clustering every data point is assigned a certain membership to each cluster according to how similar or dissimilar it is from other data points in the cluster. The FCM algorithm introduces fuzziness for each data point called membership. Thus, it can retain more information than Hard Clustering Algorithm. FCM algorithm clusters a dataset into c clusters or partitions such that every element in a cluster are as similar as possible. The clustering is performed by iterative optimization of an objective function J_{FCM} (Bezdek, Ehrlich, & Full, 1984):

$$J_{FCM} = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^{m} d\left(x_i, v_j\right)^2 \quad (1)$$

where, n is the number of connections in dataset and c is the number of clusters. x_i is the i^{th} connection of the dataset, v_j is the cluster center of j^{th} cluster and $d\left(x_i, v_j\right)$ is the Euclidean distance measure. The fuzzification parameter m determines the amount of fuzziness in the membership value. After every iteration the membership value u_{ij} and cluster center v_j are updated using equation (2) and (3) respectively:

$$u_{ij} = \frac{1}{\sum_{l=1}^{c}\left[\frac{d\left(x_i, v_j\right)}{d\left(x_i, v_l\right)}\right]^{\frac{2}{(m-1)}}} \quad (2)$$

$$v_j = \frac{\sum_{i=1}^{n} u_{ij}^{m} x_i}{\sum_{i=1}^{n} u_{ij}^{m}} \quad (3)$$

The membership is the degree of confidence for a data vector to a particular cluster. The membership value is used to determine to which cluster a data vector belongs. With every iteration, the objective function is minimized (Leski, 2003). When the objective function reaches global minima the dataset is said to be clustered. The FCM algorithm often gets stuck at local optima. In their proposed model the authors have used a Penalty Reward based FCM clustering algorithm which overcomes the limitations of FCM.

3.2. K-Nearest Neighbor

K-Nearest Neighbor (KNN) is one of the most fundamental and simple classification method. Fix and Hodges (1951) introduced the K-Nearest Neighbor rule as a non-parametric rule for pattern classification. The KNN algorithm assigns the given data to a class, depending upon the class label of its K-nearest neighbors. The distance between the given data and data in sample space are calculated from which only the K-nearest neighbors are chosen. The KNN algorithm has low accuracy rate. In order to overcome the low accuracy the authors have used a modified approach for KNN algorithm. The algorithm employs a strength measure for every connection in dataset to generate a better and accurate rule set.

3.3. Dempster-Shafer Theory

Dempster-Shafer Theory (DST) is also known as evidence theory or theory of belief functions. Shafer extended the works of Dempster (1966) (1967) (1968) on upper and lower probabilities to express uncertain judgments and published an article as "Mathematical Theory of Evidence" (Shafer, 1976). Shafer termed the lower probability function as Belief and the upper probability function as Plausibility for a given piece of evidence. These values of Belief and Plausibility for all evidences are combined using the Dempster-Shafer (DS) Rule of Combination. The result from the combined evidence determines the selection of correct hypothesis from the frame of discernment. A set of all possible states Ω in a system is known as frame of discernment. Every element in the frame of discernment is known as hypothesis. DST requires a mass to be assigned to all the elements of the power set 2^{Ω}

of frame of discernment. The pieces of evidence are associated with each hypothesis or a set of hypotheses (Kay, 2007). For every element in the power set a mass function is assigned such that $m:2^{\Omega} \to \left[0,1\right]$. There are three important measures in DST:

1. Belief measure is assigned as $bel:2^{\Omega} \to \left[0,1\right]$ and is given by (4):

$$bel\left(H\right) = \sum_{Y \subseteq H; Y \neq \varnothing} m_{\Omega}\left(Y\right) \tag{4}$$

2. Plausibility measure is assigned as $pl:2^{\Omega} \to \left[0,1\right]$ and is given by (5):

$$pl\left(H\right) = \sum_{H \cap Y \neq \phi} m_{\Omega}\left(Y\right) \tag{5}$$

3. Uncertainty measure is given by $pl\left(H\right) - bel\left(H\right)$.

The DS Combination formula is represented by an orthogonal sum of evidences as given by (6):

$$m_{1,2} = m_1 \oplus m_2 \tag{6}$$

The belief towards a particular hypothesis is determined for all the pieces of evidence using this combinational formula. This helps us to determine, up to what degree the evidences support a given hypothesis or not.

The preliminary theory provides a background upon which the authors have developed their Intrusion Detection System for Cloud environment. The FCM clustering acts as a backbone based on which they have proposed a Penalty Reward based FCM clustering algorithm. The authors have used a modified approach for KNN algorithm using the strength term for each connections and then classified the test connections using Dempster-Shafer Theory. In order to analyze the proposed IDS, NSL-KDD dataset has been used. The following section briefly describes the dataset.

4. DATASET DESCRIPTION

In this paper the authors have used NSL-KDD dataset to evaluate the model. "KDDTrain+", "KDDTest+", "20%KDDTraining+" and "KDDTest-21" are the four components of NSL-KDD dataset. For training and testing purpose in the proposed model they have used "KDDTrain+" and "KDDTest+" dataset, which contains 125,973 and 22,544 connections respectively. Each connection has 41 features which are categorized into following four categories: basic features, content features, time-based traffic features and host-based traffic features. In addition to these features an additional feature is included known as the decision feature. The decision feature identifies a connection as normal or attack. The features in NSL-KDD dataset contain numeric as well as non-numeric values. Therefore there is a need to preprocess data i.e. the non-numeric features needs to be converted into numeric form. The range of the feature value is different due to the combination of discreet and continuous values (Ghosh, Debnath, Metia, & Dutta, 2014). Thus the dataset is normalized to make the value of features comparable. This is achieved by using the min-max normalization (Han,

Kamber, & Pei, 2012). Therefore, the final dataset used is preprocessed and normalized, containing attack and normal connections.

5. PROPOSED MODEL

Several models for Intrusion Detection System (IDS) in Cloud computing were introduced by researchers over time. After considering their benefits and shortcomings the authors have proposed their anomaly based IDS. In the proposed model they have divided the IDS into two phases-training phase and testing phase. They have used NSL-KDD dataset in both the phases. The first is the training phase which is divide into two modules-Penalty Reward based Fuzzy C-Means clustering algorithm and validation of clustering. In the first module of training phase a novel Penalty Reward based FCM clustering is performed on the training dataset. In the second module, trustworthiness is assigned to each training connection according to its neighboring connections. The second phase is the testing phase which is also divided into two modules. In the first module of testing phase, best rule set is generated from the rule set obtained from training phase using a modified approach for KNN algorithm. Lastly, in the second module, the evidence from the best rule set are combined using the Dempster-Shafer Rule of Combination to classify the test connection as normal or attack (see Figure 1).

5.1. Training Phase

In the training phase a rule set is generated according to which an incoming connection is classified in the testing phase. For the experiment, authors have used NSL-KDD Train dataset and performed a PRFCM clustering on it to classify the connections in two class labels i.e. normal or attack. Using the class label of neighbors, the connections are further strengthened. Depending upon the membership and strength value of connections, a rule set is generated in the training phase. The following sections give details of the aforementioned methods.

5.1.1. A Novel Penalty Reward based Fuzzy C-Means Clustering (PRFCM)

The FCM algorithm is very sensitive to noise and is easily struck at local optima. In order to overcome this limitation, spatial context of connection is taken into account considering its neighboring connections. A Penalty Reward based FCM algorithm is implemented here which can handle small as well as large amount of noise by adjusting a penalty and reward coefficient. The algorithm takes into account both the feature information and spatial information. The objective function J_{FCM} is modified to incorporate the penalty and reward term by which it can overcome the local optima. The membership function changes but the function for finding the cluster center remains same as that of FCM algorithm. The new objective function of the PRFCM algorithm is defined as follows:

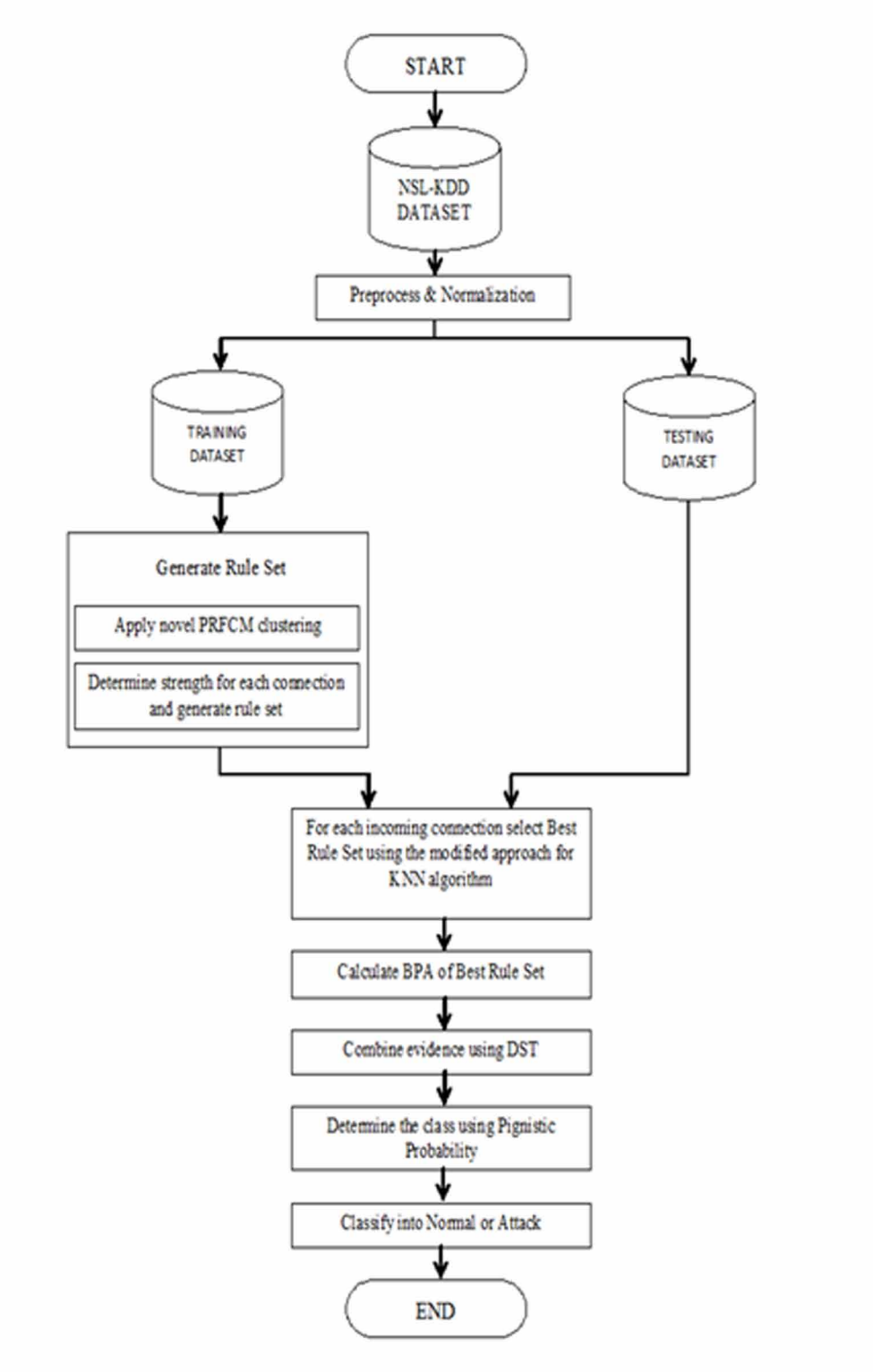
$$J_{PRFCM} = J_{FCM} + J_{PFCM} - J_{RFCM} \tag{7}$$

$$J_{FCM} = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^{m} \cdot d\left(x_i, v_j\right)^2 \tag{8.1}$$

$$J_{PFCM} = \sum_{k=1}^{K}\sum_{i=1}^{n}\sum_{j=1}^{c} W \cdot u_{ij}^{m} \cdot u_{kj} \cdot \ln \alpha_j \tag{8.2}$$

$$J_{RFCM} = \sum_{k=1}^{K}\sum_{i=1}^{n}\sum_{j=1}^{c} W \cdot u_{ij}^{m} \cdot \left(1 - u_{kj}\right) \cdot \ln\left(1 - \alpha_j\right) \tag{8.3}$$

Figure 1. Flow of events of the proposed model



where, the dataset is defined as X = { x_1, x_2, $x_3 \ldots x_n$ } having n number of connections and each connection is a p dimensional vector. The number of clusters is given by c, $2 \le c \le n$. u_{ij} is the degree of membership of a connection x_i to j^{th} cluster center and $0 \le u_{ij} \le 1$. Let L = { l_1, l_2, $l_3 \ldots l_c$ } be the set of class labels into which a connection can be classified. The parameter m is a weighted exponent called the fuzzification parameter. Its value can range from [1, ∞). This value determines the amount of fuzziness in the clustering. When $m = 1$, the process becomes hard clustering. Normally its value is in the range of 1.5 to 3 (Bezdek et al., 1984). $d\left(x_i, v_j\right)$ is the Euclidean distance between the x_i connection and v_j cluster center. The initial cluster centers are randomly initialized and the initial membership values are calculated by membership equation of FCM algorithm using equation (2).

The objective function J_{PRFCM} is minimized by iterative optimization of the function. At every iteration the membership values are updated using equation (9):

$$u_{ij} = \frac{\left[d\left(x_i, v_j\right)^2 + W.\sum_{k=1}^{K} u_{kj}.\ln\alpha_j - W.\sum_{k=1}^{K}\left(1-u_{kj}\right).\ln\left(1-\alpha_j\right)\right]^{-1/(m-1)}}{\sum_{l=1}^{C}\left[d\left(x_i, v_l\right)^2 + W.\sum_{k=1}^{K} u_{kl}.\ln\alpha_l - W.\sum_{k=1}^{K}\left(1-u_{kl}\right).\ln\left(1-\alpha_l\right)\right]^{-1/(m-1)}} \tag{9}$$

Here, $\sum_{j=1}^{c} u_{ij} = 1, \forall i$ and K signifies the number of nearest neighbors to a given connection. The cluster centers are updated using the equation (3).

The membership function incorporates the penalty term and reward term. When a given connection has a higher membership to a particular class and the neighboring connections also have higher membership to that particular class the connection is rewarded and the reward term becomes greater than the penalty term. Whereas when the neighboring connection has a lower membership to that particular class the connection is penalized and the penalty term becomes greater than the reward term. In order to penalize or reward a connection, K nearest neighbor connections are considered for every connection in the data set. The value of K was experimentally determined. It was observed that with increasing value of K the accuracy of the algorithm increased up to a certain extent after which there was a steep fall. This fall indicates that if the value of K is taken to be very large then more number of neighbors determine the result of the connection and hence it is more biased. A bias constant, W is introduced in the penalization-reward term and is tested for different values from 0 to 1. When the value of $W = 0$ the penalization-reward term is nullified and the membership function becomes the same as that of FCM algorithm. With increasing value of W, it was found that the accuracy of the algorithm increased up to a certain limit and then started to decrease. This suggests that if the total impact of the K neighbor is taken into account then the result is highly biased. Therefore, only a part of the impact of neighboring connection is taken into account while giving reward or penalty, to give an unbiased result. Another term introduced was α_j where:

$$\alpha_j = \frac{\sum_{i=1}^{n} u_{ij}^{\ m}}{\sum_{l=1}^{c}\sum_{i=1}^{n} u_{il}^{\ m}} \tag{10}$$

The variable α_j is the fraction of membership values to j^{th} cluster with respect to the membership values to all the clusters for all the connections. The process is iterated until the objective function is minimized or the membership values reach a convergence.

5.1.2. Algorithm for PRFCM Clustering

1. randomly initialize cluster centers
2. initialize membership matrix using equation (2)
3. do:
   calculate value of α_j using equation (10)
   update cluster center using equation (3)
   update membership matrix using equation (9)
4. while: $\left(\max\left|u_{ij}^{(t+1)} - u_{ij}^{t}\right| < \varepsilon\right)$

After the PRFCM clustering a final membership value of every connection for each cluster is assigned. This value only signifies its decision but not the trustworthiness of decision. In the following section the authors introduce a strength value to every connection in the train set considering the decision of its neighbors to generate an efficient rule set.

5.1.3. Strength

The clustering process assigns a membership value to every connection for all the class. The membership is the degree of belongingness of a connection to a particular class whereas strength of a connection is its trustworthiness to that class with respect to its neighbors. The strength signifies that how much a particular connection is trustworthy or valid to serve as a rule set for an incoming connection in testing phase. After clustering a training connection is strengthened by the class label of its nearest neighbor connections. If the neighbors of a connection support same class label then the connection is strengthened or else it is weakened. In the proposed model the authors have taken K nearest neighbors of a training connection and assigned higher strength to connection whose K neighbors belong to the same class as that of the sample connection. The class label of each connection is compared to that of its K -nearest neighbors:

$$strength\left(x_i\right) = \frac{1}{K}\sum_{k=1}^{K} classlabel\left(x_i, x_k\right) \tag{11}$$

Here, $classlabel\left(x_i, x_k\right)$ is a function which returns 1 when connection x_i belongs to the same class as x_k else 0. Therefore the strength is a fraction of total number of neighbors with same class labels as that of connection to the number of K neighbors.

5.1.4. Algorithm for Finding Strength of Connections in Train Set

1. for each connection in the training data set find K-nearest neighbors
2. find the class label of sample connection and class label of its neighbor
3. *for every neighbor: if* $\left(classlabel\left(x_i, x_k\right) == 1\right)$ *counter = counter+1*
4. strength(x) = counter/K

At the end of training phase a membership values for each connection to all the clusters and its strength is assigned. These strength and membership values are used to generate rule set

$R = \{r_1, r_2, r_3 \ldots r_n\}$. This rule set R is further used in the testing phase to generate best rule set to classify an incoming connection.

5.3. Testing Phase

In the testing phase the authors have used the NSL-KDD Test data set. A modified approach for K-Nearest Neighbor algorithm is used to generate best rule set from the rule set generated in the training phase. Thereafter, Dempster-Shafer Rule of Combination is used to fuse multiple evidences from best rule set and the final decision is made using the pignistic probability equation on the test dataset.

5.3.1. Best Rule Set Generation

In order to classify an incoming connection a best rule set is generated from the rule set R, formed in the training phase. A modified approach for KNN algorithm is used to generate the best rule set R′, where R′⊆ R. The best rule set R′ is generated by taking the nearest rules from the incoming connection. The algorithm assigns a weight to every connection in best rule set according to the distance of it from an incoming connection. The value of weight is inversely proportional to the distance. If the distance between incoming connection and a connection in rule is large, then the connection is said to have "less" influence and if the distance is small the connection is said to have "more" influence. The distance of all the connections in the rule set R to an incoming connection IC is calculated and the most informative or the best rule set R′ is selected based on their distance measure. A weight is generated for every connection in the best rule set using the equation given below:

$$w_i = \frac{d\left(x_{farthest}, IC\right) - d\left(x_i, IC\right)}{d\left(x_{farthest}, IC\right) - d\left(x_{closest}, IC\right)} \tag{12}$$

where, $x_{farthest}$ is the farthest rule from IC and $x_{closest}$ is the nearest rule. $d\left(x_i, IC\right)$ is the Euclidean distance measure between the connection x_i and incoming connection IC. In addition to this weight, the strength of each connection of best rule set is also incorporated in the effective weight matrix. Thus, the effective weight is product of weight and strength of each connections of R′:

$$\omega_i = w_i \cdot strength\left(x_i\right) \tag{13}$$

A confidence value is now assigned to every connection in the best rule set R′. This confidence value is proportional to the membership value. β_{ij} is the confidence value of x_i connection to l_j class:

$$\beta_{ij} = \frac{u_{ij}}{\sum_{l=1}^{c} u_{il}} \tag{14}$$

5.3.2. Algorithm for Finding Best Rule Set R'

1. *for every connection in rule set R calculate distance to incoming connection* IC
2. find the minimum distance best rule set R′
3. for every connection in R′ calculate weight using equation (12)

4. calculate effective weight ω_i using equation (13)
5. find confidence value β_{ij} using equation (14)

Now, a best rule set R′ with effective weight ω_i and confidence value β_{ij} to act as pieces of evidence in Dempster-Shafer Theory is obtained. This best rule set R′ contains sufficient information to determine the class label of the incoming connection IC.

5.3.3. Dempster-Shafer Rule of Combination

Dempster-Shafer Rule is used to combine the evidences for different hypothesis. The set of class label L is identified as state space θ. This state space θ is also known as frame of discernment in DST. Every element in frame of discernment is known as hypothesis. The DST assigns a mass (probability) for every element in the subset of θ, which is the power set 2^{θ}. This is known as Basic Probability Assignment (BPA). The best rule set R′ generated from above section, serves as pieces of evidence in Dempster-Shafer Theory. In order to assign mass to every element of the power set, the BPA of all the evidences in R′ is to be calculated. The mass function $m\left(H\right)$ is BPA of H which is a subset of θ. H is called a non-focal element if $m\left(H\right)$>0. The mass function satisfies $\sum_{H\subseteq\theta} m\left(H\right)=1$, where $0\le m\left(H\right)\le 1$.

In order to calculate BPA we use the product of confidence value β and effective weight ω. In this paper the authors have taken two hypothesis A (Attack) and N (Normal) in frame of discernment. Then the power set contains $2^{\theta}=\left\{\varnothing,\left\{A\right\},\left\{N\right\},\left\{A,N\right\}\right\}$, where $m\left(\varnothing\right)=0$.

The BPA of A and N are calculated as:

$$m_i\left(A\right)=\omega_i.\beta_{iA} \tag{15}$$

$$m_i\left(N\right)=\omega_i.\beta_{iN} \tag{16}$$

After assigning BPA to the subsets of two hypotheses (signifying the class label) a BPA needs to be assigned to the frame:

$$m_i\left(A,N\right)=1-\sum_{j}^{2^{\theta}} m_i\left(H_j\right) \tag{17}$$

Here, i is the i^{th} piece of evidence or rule from the best rule set.

There are three basic evidential functions which can be obtained from the BPA. They are given below.

5.3.3.1. Belief

The belief $bel\left(H\right)$ is the measure of evidence that the given piece directly supports hypothesis H or its subsets. It forms as a lower bound, indicating the impact of evidence on the hypothesis H. It is given by $bel:2^{\theta}\rightarrow\left[0,1\right]$:

$$bel\left(H\right)=m\left(H\right) \tag{18}$$

5.3.3.2. Plausibility

The plausibility $pl\left(H\right)$ is the measure of evidence that the given piece is not assigned to the hypothesis $\bar{H}$. It serves as an upper bound, indicating the value upto which one doubts the belief. It is given by $pl:2^{\theta} \to \left[0,1\right]$:

$$pl\left(H\right) = 1 - bel\left(\bar{H}\right) \tag{19}$$

5.3.3.3. Uncertainty

The difference between plausibility and belief is known as uncertainty. This is the measure of uncertainty of hypothesis H and is given by $pl\left(H\right) - bel\left(H\right)$.

The Dempster-Shafer Rule of Combination is now used to combine the belief for a hypothesis. The combinational formula for two evidences is represented as an orthogonal sum of evidences in equation (6). The numerator of the equation (20) suggests the accumulated evidence for Y and Z which supports hypothesis H:

$$m_{1,2}\left(H\right) = \frac{\sum_{Y \cap Z = H \neq \varnothing} m_1\left(Y\right) m_2\left(Z\right)}{1 - \gamma} \tag{20}$$

$$\gamma = \sum_{Y \cap Z = \varnothing} m_1\left(Y\right) m_2\left(Z\right) \tag{21}$$

γ is the measure of conflict between masses of the hypotheses. $\left(1 - \gamma\right)$ is the normalization factor which ignores any conflict and allots the conflict mass to the null set. Using the above combination formula the fused mass m(A), m(N) and m(A, N) is found out. Now, a pignistic probability function is used to make the decision using combined weights. The pignistic probability function determines to which class the incoming connection IC can be classified:

$$\rho\left(l_i\right) = m\left(l_i\right) + \frac{m\left(L\right)}{c} \tag{22}$$

Here l_i is the class label, c is the number of classes and $m\left(L\right)$ is the mass assigned to the frame. The value of Pignistic probability falls between 0 and 1 and it satisfies $\sum_{i=1}^{c} \rho\left(l_i\right) = 1$.

The incoming connection IC is assigned to the class having highest value of pignistic probability $\rho\left(l_i\right)$.

Table 1. Confusion matrix for FCM clustering algorithm

	Attack	**Normal**
Attack	36944	21686
Normal	175	67168

5.3.4. Algorithm for Combination of Evidence

1. for every connection in best rule set R′, find BPA using equation (15), (16) and (17)
2. combine evidence using equation (20)
3. give final classification using pignistic probability in equation (22)

Therefore, the testing phase is completed after the incoming connection is classified as normal or attack. The proposed model overcomes the limitations of FCM as well as KNN algorithm and introduced a new algorithm with better results. The proposed anomaly based IDS is implemented on NSL-KDD dataset. Results and detailed analysis are prepared in the following section.

6. RESULT AND ANALYSIS

An IDS with higher efficiency plays an essential role in Cloud environment. The proposed anomaly based IDS proved to be highly efficient. The accuracy of the proposed IDS has been evaluated in two parts. In the first part the accuracy of novel PRFCM clustering algorithm is determined for a fixed value of K and W . In the second part the efficiency of the proposed IDS is determined. The value of K and W is experimentally found out by plotting the accuracy for different value of K and W .

Figure 2 gives a graph of accuracy of the proposed novel PRFCM clustering algorithm for different values of K . It has been observed that for a constant value of W the accuracy rises with increasing value of K up to $K = 3$ after which it starts to fall, thus $K = 3$ is taken for the experiment.

Figure 3 gives a graph of accuracy of PRFCM for different values of W . It has been observed that for a constant value of K the accuracy rises with increasing value of W up to $W = 0.5$ after which it starts to fall, thus $W = 0.5$ is taken for the experiment.

The two phases of the proposed IDS are evaluated using confusion matrix and some metrics. The values of these metrics judge the individual performances in two parts. In the first part Table 1 gives the confusion matrix for FCM clustering and Table 2 gives the confusion matrix of the novel PRFCM clustering algorithm for $K = 3$ and $W = 0.5$. From the results, it is found that the accuracy of novel PRFCM is higher than that of FCM clustering algorithm. The PRFCM algorithm can be reduced to FCM algorithm if the value of W is reduced to 0. When the value of $W = 0$ the penalty

Figure 2. Plot of Accuracy v/s K (number of neighbors)

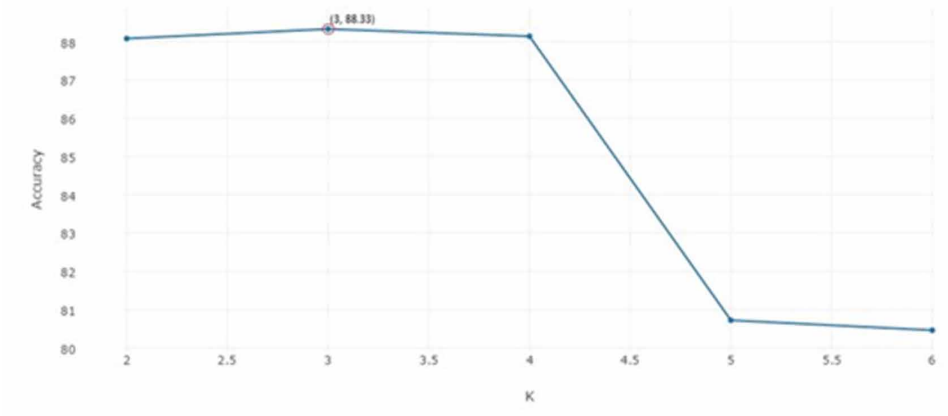


Figure 3. Plot of Accuracy v/s W (bias constant)

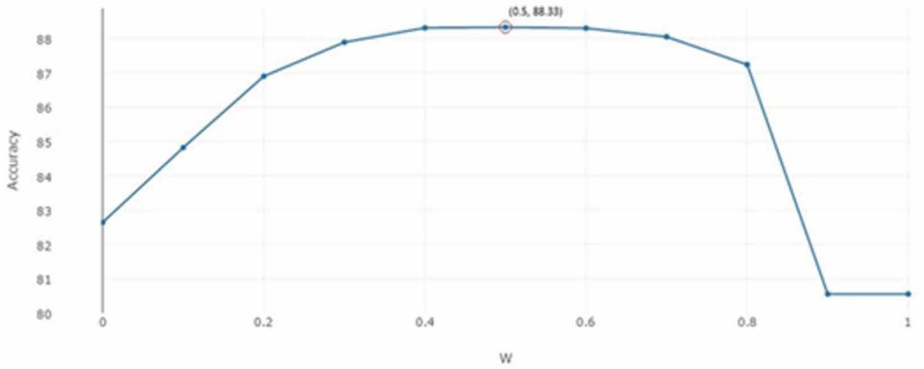


Table 2. Confusion matrix for novel PRFCM clustering algorithm

	Attack	Normal
Attack	44778	13852
Normal	855	66488

reward term goes to zero and hence it becomes FCM algorithm. The use of penalty-reward term in PRFCM clustering algorithm improves the clustering. It overcomes the sensitivity to noise of FCM algorithm by adjusting the penalty-reward terms accordingly. The increase in accuracy of PRFCM from FCM algorithm can be viewed in Figure 4. It is also observed that the detection rate for PRFCM clustering is significantly higher than FCM clustering algorithm. The accuracy of novel PRFCM algorithm was found to be 88.33% whereas that of FCM algorithm was found to be 82.65%. The values determined for all the metrics are given in Table 3.

In the second part the accuracy of proposed model has been evaluated on the test dataset. A confusion matrix was created as in Table 4 and some metrics were used to evaluate the result as in Table 5. With the proposed anomaly based Intrusion Detection System the authors have observed that the false alarm rate is significantly low which results in an increased precision.

Figure 5 shows the values of different metrics in training and testing phase. The low values of false alarm rate suggest that the proposed IDS produces a very less number of false positives. This low false alarm rate thereby increases the precision of the IDS. The increase in efficiency of PRFCM clustering has great impact on the overall increase in accuracy of the proposed model. In addition to this, the use of Dempster-Shafer Theory to combine the evidences along with a modified approach for KNN algorithm also increased the accuracy of the model. By implementing this anomaly based IDS in a Cloud environment the risk of false alarm can be significantly reduced. Thus, the PRFCM clustering algorithm and modified KNN based Dempster-Shafer rule of combination plays a vital role in increasing the efficiency and reliability of the proposed IDS.

Figure 4. Detection Results of FCM and PRFCM Clustering

Table 3. Values of different metrics for FCM and novel PRFCM clustering algorithm

	Accuracy	False Alarm Rate	Detection Rate	True Negative Rate	Precision
FCM	0.82646281	0.0025986	0.630121	0.9974013	0.9952854333
PRFCM	0.88325276	0.0126962	0.763738	0.9873038	0.9812365595

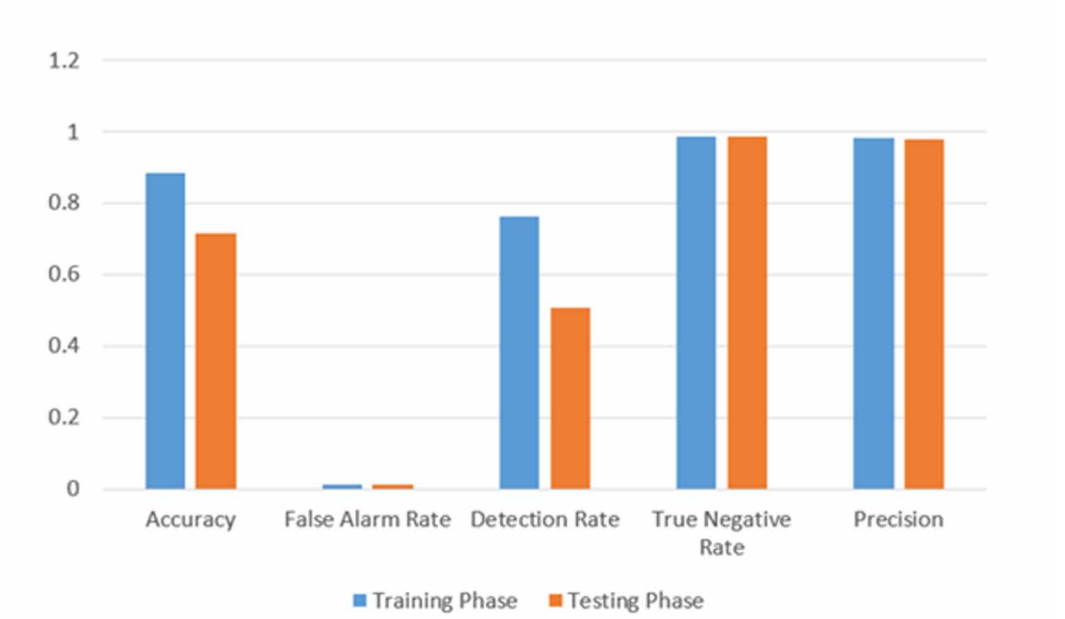
Table 4. Confusion matrix for testing dataset

	Attack	Normal
Attack	6521	6312
Normal	131	9580

Table 5. Values of different metrics for Testing Phase

Accuracy	False Alarm Rate	Detection Rate	True Negative Rate	Precision
0.71420	0.01348985	0.50814307	0.98651014	0.98030667

Figure 5. Detection results of training and testing phase



7. CONCLUSION

With the fast growth of Cloud computing the complexity of its services and number of user increases day by day. The Cloud environment is highly vulnerable to intrusion owing to its vastness. The intruders exploit these vulnerabilities and perform different types of attack in the network. Therefore, there is a need for an efficient Intrusion Detection System. In order to detect novel attacks as well as variations in attacks, an anomaly based model is required. In this paper the authors have proposed a new anomaly based Intrusion Detection System. The anomaly based IDS generates a rule set for normal behavior and classifies an incoming connection according to deviation from that rule set. The authors have introduced a novel Penalty Reward based Fuzzy C-Means (PRFCM) clustering algorithm as an improvement to the existing FCM algorithm in the training phase. The proposed PRFCM algorithm takes into account the spatial information of a connection. It penalizes or rewards a connection based on the class labels of its neighbors. It also overcomes the problem of sensitivity to noise in FCM algorithm. A significant increase in clustering accuracy of novel PRFCM clustering from FCM clustering is recorded in the above experiment. Thereafter the strength of all training connections is found. The authors have also proposed a modified approach for K-Nearest Neighbor (KNN) algorithm to form best rule set and used the Dempster-Shafer Theory (DST) to combine the evidences from best rule set. In the testing phase NSL-KDD test dataset is used to verify the efficiency of the given model. The detailed and part-wise analysis for the complete work is done and is presented in this paper. In their future work the authors look forward towards making an Intrusion Detection Prevention System (IDPS) to provide an extensive security to the Cloud environment. With the proposed work the authors have successfully achieved an efficient and reliable model for Intrusion Detection System in Cloud computing paradigm.

REFERENCES

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The Fuzzy c-Means Clustering Algorithm. *Computers & Geosciences*, *10*(2-3), 191–203. doi:10.1016/0098-3004(84)90020-7

Chen, T. M., & Venkataramanan, V. (2005). Dempster-Shafer Theory for Intrusion Detection in Ad Hoc Networks. In IEEE Computer Society (pp. 35–41).

Dempster, A. P. (1966). New Methods for Reasoning Towards Posterior Distributions based on Sample Data. *Annals of Mathematical Statistics*, *37*(2), 355–374. doi:10.1214/aoms/1177699517

Dempster, A. P. (1967). Upper and Lower Probabilities Induced by a Multivalued Mapping. *Annals of Mathematical Statistics*, *38*(2), 325–339. doi:10.1214/aoms/1177698950

Dempster, A. P. (1968). A Generalization of Bayesian Inference. *Journal of the Royal Statistical Society. Series B. Methodological*, *30*(2), 205–247.

Denning, D. E. (1987). An intrusion-detection model. *IEEE Transactions on Software Engineering*, *13*(2), 222–232. doi:10.1109/TSE.1987.232894

Denoeux, T. (1995). A k-Nearest Neighbor Classification Rule Based on Dempster-Shafer Theory. *IEEE Transactions on Systems, Man, and Cybernetics*, *25*(5), 804–813. doi:10.1109/21.376493

Dudani, S. A. (1976). The Distance-Weighted k-Nearest Neighbor Rule. *IEEE Transactions on Systems, Man, and Cybernetics*, *6*(4), 325–327. doi:10.1109/TSMC.1976.5408784

Esmaili, M. (1997). Dempster-Shafer Theory and Network Intrusion Detection Systems. *Scientia Iranica*, *3*(4), 147–157.

Fix, E., & Hodges, J.L. (1951). Discriminatory analysis. Nonparametric Discrimination: Consistency Properties.

Ghosh, P., Debnath, C., Metia, D., & Dutta, R. (2014). An Efficient Hybrid Multilevel Intrusion Detection System in Cloud Environment. *IOSR Journal of Computer Engineering*, *16*(4), 16–26. doi:10.9790/0661-16471626

Ghosh, P., Mandal, A. K., & Kumar, R. (2015). An Efficient Cloud Network Intrusion Detection System. In Information Systems Design and Intelligent Applications, Advances in Intelligent Systems and Computing (pp. 91–99). India: Springer. doi:10.1007/978-81-322-2250-7_10

Govindarajan, M., & Chandrasekaran, R. M. (2009). Intrusion Detection Using k-Nearest Neighbor. In IEEE ICAC (pp. 13–20). doi:10.1109/ICADVC.2009.5377998

Govindarajan, M., & Chandrasekaran, R. M. (2011). Intrusion detection using neural based hybrid classification methods. *Computer Networks*, *55*(8), 1662–1671. doi:10.1016/j.comnet.2010.12.008

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining: Concepts and Techniques* (3rd ed.). Elsevier Book. doi:10.1007/978-1-4419-1428-6_3752

Jamdagni, A., Tan, Z., He, X., Nanda, P., & Liu, R. P. (2013). RePIDS: A multi tier Real-time Payload-based Intrusion Detection System. *Computer Networks*, *57*(3), 811–824. doi:10.1016/j.comnet.2012.10.002

Kay, R. U. (2007). Fundamentals of the Dempster-Shafer theory and its applications to system safety and reliability modeling. *RTA*, *3*(4), 173–185.

Keller, J. M., Gray, M. R., & Givens, J. A. (1985). A Fuzzy K-Nearest Neighbor Algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-15*(4), 580–585. doi:10.1109/TSMC.1985.6313426

Khazaee, S., & Rad, M. S. (2013). Using fuzzy c-means algorithm for improving intrusion detection performance. *Proceedings of the2013 13th Iranian Conference on Fuzzy Systems (IFSC). IEEE* (pp. 1–4). doi:10.1109/IFSC.2013.6675669

Lee, W., & Stolfo, S. J. (2000). A framework for constructing features and models for intrusion detection systems. *ACM Transactions on Information and System Security*, *3*(4), 227–261. doi:10.1145/382912.382914

Leski, J. M. (2003). Generalized Weighted Conditional Fuzzy Clustering. *IEEE Transactions on Fuzzy Systems*, *11*(6), 709–715. doi:10.1109/TFUZZ.2003.819844

Liao, H. J., Lin, C. H. R., Lin, Y. H., & Tung, K. Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36, 16–24.

Liao, Y., & Vemuri, V. R. (2002). Use of K-Nearest Neighbor classifier for intrusion detection. *Computers & Security, 21*(5), 439–448.

Lombardi, F., & Pietro, R. Di. (2010). Secure virtualization for cloud computing. *Journal of Network and Computer Applications, 34*(4), 1113–1122.

Modi, C., Patel, D., Patel, H., Borisaniya, B., Patel, A., & Rajarajan, M. (2013). A survey of intrusion detection techniques in Cloud. *Journal of Network and Computer Applications, 36*(1), 42–57.

Nadiammai, G. V., & Hemalatha, M. (2012). An evaluation of clustering technique over intrusion detection system.*Proceedings of the International Conference on Advances in Computing, Communications and Informatics - ICACCI '12* (pp. 1054–1060). doi:10.1145/2345396.2345565

Özyer, T., Alhajj, R., & Barker, K. (2007). Intrusion detection by integrating boosting genetic fuzzy classifier and data mining criteria for rule pre-screening. Journal of Network and Computer Applications, 30(1), 99–113.

Patel, A., Taghavi, M., Bakhtiyari, K., & Júnior, J. C. (2013). An intrusion detection and prevention system in cloud computing: A systematic review. Journal of Network and Computer Applications, 36(1), 25–41.

Rao, M. V., Damodaram, A., & Charyulu, N. C. B. (2012). Algorithm for Clustering with Intrusion Detection Using Modified and Hashed K – Means Algorithms. In Advances in Computer Science, Eng. Appl., AISC (Vol. 167, pp. 737–744). Berlin: Springer -Verlag.

Shafer, G. A. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.

Song, C., & Ma, K. (2009). Design of Intrusion Detection System Based on Data Mining Algorithm. *Proceedings of the2009 International Conference on Signal Processing Systems* (pp. 370–373). IEEE Computer Society. doi:10.1109/ICSPS.2009.202

Su, M. Y. (2011). Using clustering to improve the KNN-based classifiers for online anomaly network traffic identification. Journal of Network and Computer Applications, 34(2), 722–730.

Subashini, S., & Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. Journal of Network and Computer Applications, 34(1), 1–11.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338–353. doi:10.1016/S0019-9958(65)90241-X

Partha Ghosh is an Assistant Professor of Information Technology at Netaji Subhash Engineering College, Maulana Abul Kalam Azad University of Technology, Kolkata, West Bengal, India. He has done his MTech in Computer Science and Engineering from Calcutta University in 2003. His research interests include Cloud Computing, Machine Learning, Computer Networks and Security.

Shivam Shakti is a student of B.Tech in Information Technology at Netaji Subhash Engineering College, Maulana Abul Kalam Azad University of Technology, Kolkata, West Bengal, India. His interests include Cloud Computing and Machine Learning.

Santanu Phadikar is an Assistant Professor of Computer Science and Engineering at Maulana Abul Kalam Azad University of Technology, Kolkata, West Bengal, India. He has done his MTech. in Computer Science and Engineering from Calcutta University in 2003. He pursued his PhD from Indian Institute of Engineering Science and Technology, Shibpur, West Bengal, India in 2013. His research area includes Machine Learning, Soft Computing and Cloud Computing.