An Intelligent CRF Based Feature Selection for Effective Intrusion Detection

Sannasi Ganapathy[1], Pandi Vijayakumar[2], Palanichamy Yogesh[1], and Arputharaj Kannan[1]
[1]Department of Information Science and Technology, Anna University, India
[2]Department of Computer Science and Engineering, University College of Engineering Tindivanam, India

Abstract: *As the internet applications are growing rapidly, the intrusions to the networking system are also becoming high. In such a scenario, it is necessary to provide security to the networks by means of effective intrusion detection and prevention methods. This can be achieved mainly by developing efficient intrusion detecting systems that use efficient algorithms which can identify the abnormal activities in the network traffic and protect the network resources from illegal penetrations by intruders. Though many intrusion detection systems have been proposed in the past, the existing network intrusion detections have limitations in terms of detection time and accuracy. To overcome these drawbacks, we propose a new intrusion detection system in this paper by developing a new intelligent Conditional Random Field (CRF) based feature selection algorithm to optimize the number of features. In addition, an existing Layered Approach (LA) based algorithm is used to perform classification with these reduced features. This intrusion detection system provides high accuracy and achieves efficiency in attack detection compared to the existing approaches. The major advantages of this proposed system are reduction in detection time, increase in classification accuracy and reduction in false alarm rates.*

Keywords: *Intrusion detection system, feature selection, false alarms, LA, intelligent CRF, ICRFFSA, LAICRF.*

Received January 31, 2013; accepted November 10, 2013; published online March 8, 2015

1. Introduction

Due to the rapid improvements in the current network technologies, internet applications play a vital role in human life. Moreover, enormous amount of information are provided by the web to the users. At the same time, it also contains a lot of ways to provide risks to the users when they are communicating using the Internet. Such internet attacks are responsible for both harmful and harmless risks to most applications. Because of the threats provided by intrusions, authenticated information is accessed by unauthorized users present in the network.

Therefore, it is necessary to secure the valuable information pertaining to an organization when employees of that organization are communicating using the network. In the current internet world, intrusion detection [2] is one of the high priority and challenging tasks for network administrators and security professionals. Even in the presence of more sophisticated security mechanisms, the attackers come up with newer and more advanced penetration techniques to defeat the installed security systems [4]. Thus, there is a need to provide security to the networks from known vulnerabilities and at the same time it is necessary to take steps to detect and prevent new and unseen attacks by developing more reliable and efficient intrusion detection systems [2].

Intrusion detection systems can be classified into three categories namely network based, host based and application based systems based on their way of development. It also can be classified as signature based or anomaly based systems based on their attack detection methods [4]. The signature based systems work with known signature patterns that are predefined by the administrator. They focus mainly on the known traffic data and from that the systems analyze the unwanted traffic. In this scenario, the intrusion detection system should have a complete database that consists of all possible attacks, that helps to find the attacks easily. On the other hand, the anomaly based systems are the ones which are having collection of normal data. By analyzing the normal activity, these systems identify the abnormal activities in the network traffic. If the behavior deviates from the normal behavior, then the system concludes that an intrusion has happened in the network traffic. However, the existing IDSs have large false alarm rate and they are not very fast. This is due to the fact that the existing IDSs use all the features present in the data set for classification without analyzing the contribution of each feature for effective classification. When the number of features increases in an IDS, it not only lead to increase in detection time but also become responsible for confusing the classifier during decision making. Therefore, it is necessary to select only those features that significantly contribute in the detection process. This can be achieved by applying an effective feature selection algorithm.

Most of the existing feature selection algorithms are [1, 4, 8] useful for reducing the computational complexity since such algorithms reduce the training and testing time for detecting intrusions. One of the most important and efficient approach for feature selection algorithm for IDSs present in the literature is Conditional Random Fields (CRF) based approach [4]. Gupta *et al.* [5] have also proposed a Layered Approach (LA) using CRF for effective intrusion detection. Though, most attacks present in the Internet are detected by this approach, still it is not sufficient to cater to the security requirements of current internet based applications.

Enormous numbers of intrusion detection systems have been proposed by many researchers in the past [3, 10, 12]. In most of the systems, intruders are detected by using intelligent approaches namely, machine learning, data mining and rule based approaches. Recently, CRF was used for intrusion detection by Gupta *et al.* [5]. In their work, they proved the effectiveness of feature selection algorithm using experiments conducted with the bench mark KDD'99 cup intrusion detection data set. The main advantage of combining feature selection with IDS is that the feature selection performs better in detection accuracy and takes less time to detect the intrusions with the same accuracy rate.

In this paper, we propose an Intelligent CRF based LA (LAICRF) model which is developed by combining an Intelligent CRF based Feature Selection Algorithm (ICRFFSA) and LA based classification algorithm for effective intrusion detection. This model uses intelligent agents which are capable of sensing the environment and perform actions based on the environmental conditions. Moreover, these agents are very much useful for decision making, communication, coordination and also for executing specific tasks efficiently. In this network scenario, intelligent agents are necessary to enhance the reliability of communication and also to make collaboration decisions effectively. The major advantages of combining intelligent agents with CRF are that they are able to analyze the data set efficiently and also to make suitable decisions on attacks using the data and the behavior of malicious users. When features are selected using this algorithm and provided to the classifier for classification, the classification becomes more efficient in the following ways:

- *First*: The training and testing time are reduced.
- *Second*: The communication delay and overheads are handled effectively.
- *Third*: The classification accuracy is increased.
- *Finally*: The false alarm rate is reduced.

The rest of this paper is organized as follows: Section 2 depicts the architecture of the system proposed. Section 3 explains the proposed algorithms and mathematical analysis. Section 4 gives the results and discussion of the proposed work. Section 5 gives the conclusions and future enhancements of this proposed work.

2. System Architecture

The architecture of the proposed system for effective intrusion detection is shown in Figure 1. It consists of four major components namely knowledge base, feature selection module that contains a feature selection agent, intrusion detection module which has the training agent and decision making agents. All these components are responsible for performing intrusion detection effectively.

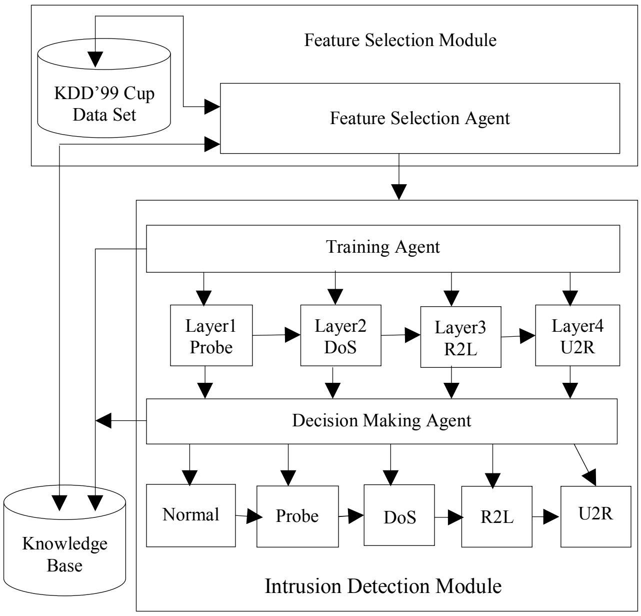

Figure 1. System Architecture

2.1. KDD'99 cup Data Set

The KDD'99 cup data set [6] used in this work is the most used comprehensive data set which is shared by many researchers. In this dataset, 41 attributes are present in each record to characterize network traffic behavior. Among these 41 attributes, 38 are numeric and 3 are symbolic. The features present in KDD data set are grouped into three categories namely basic features, content features and traffic features. Most of the researchers in the past have used only 10 % of data from KDD'99 cup data set for carrying out their experiments.

2.2. Feature Selection Module

This module consists of a feature selection agent and the data set. The feature selection module collects the data from KDD cup'99 data set which has 41 features.

- *Feature Selection Agent*: The feature selection agent selects optimal number of features from these 41 features based on the rules present in knowledge base.
- *Knowledge Base*: The knowledge base contains the properties of all the features as facts. In addition to that, it is capable of adding new rules that are generated by the CRF model during training. The feature selection agent selects appropriate features

by applying the suitable rules from the rules present in the knowledge base. The main advantage of this feature selection module is that it selects optimal number of features from the KDD'99 cup set. It contains various rules to classify the data. These rules are formed based on the LA during training. The rules are used to identify the normal data and attacks. Moreover, the rules are useful for classifying the attacks that are detected during testing.

2.3. Intrusion Detection Module

This module consists of two major components namely training agent and decision making agent. The training agent is responsible for framing layers for Probe, DoS, R2L and U2R attacks. The decision making agent is capable of making decision by testing the data and applying rules. The outputs of this module are either normal or attacks. In case of attacks, they are classified as Probe, DoS, R2L and U2R attacks.

- *Training Agent*: This agent trains the data using the LA based on dataset with reduced features. Moreover, the training agent forms the classification rules which will be stored in the knowledge base. In the LA, four layers are considered for identifying four types of attacks.
- *Decision Making Agent*: The decision making agent is responsible for performing the testing by classifying the data using rules selected from the knowledge base. These rules are generated using the Intelligent Conditional Random Field (ICRF) during the training phase. This ICRF uses a LA for distinguishing the normal records and the four types of attacks namely Probe, DoS, U2R and R2L. In order to fire the rules effectively, the decision making agent performs rule matching and uses forward chaining inference mechanism for effective decision making.

3. Proposed Work

3.1. CRF for Intrusion Detection

CRF are a type of probabilistic system [7] that is used to model the conditional distribution of random variables of any order. Moreover, a CRF is an unbiased and undirected graphical model that can be used to perform sequence labeling [9].

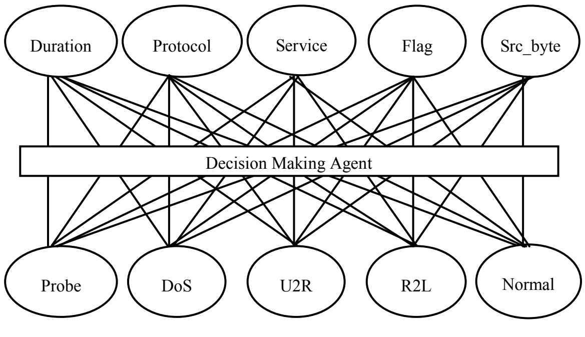

Figure 2. Graphical representation of CRF

Let X_i be a set of random variables over data sequence to be labelled and Y_i be the corresponding label sequence, with i=1, … n. Let G= (V, E) be a graph such that Y= Y_i™(V), so that Y is indexed by the vertices of G. Then, (X_i, Y_i) is a CRF, when conditioned on X_i, the random variables Y_i obey the Markov property with respect to the graph:

$$P(Y_i | X_i, Y_j, j \neq i) = P(Y_i | X_i, Y_j, i \sim j) \tag{1}$$

Where $i{\sim}j$ means that i and j are neighbours in G i.e., a CRF is a random field globally conditioned on X_i [2]. The CRF are given by the relation:

$$P_c(Y_i | X_j) \alpha exp\left(\beta_k f_k (e, y | e, X_i)\right) + \Sigma \gamma_k g_k (i, y \mid i, X_j) \tag{2}$$

Here, X_i is the data sequence, Y_i is a label sequence. Then, the features f_k and g_k are selected by the user. For example, a boolean edge feature f_k might be true if the observation X_i is tcp which is returned by the decision agent. The tag Y_{i-1} is "normal" and tag Y_i is "normal." Similarly, a vertex feature g_k is true if the observation X_i is "service=telnet" and tag Y_i is "attack".

3.2. Intelligent CRF Based Feature Selection Algorithm

Feature selection is the process of selecting appropriate features from the underlying data set such as KDD'99 cup data set for building models [11]. In the CRF based feature selection algorithm, each feature is added to class of values depending on their dependency information. However, this dependency information is computed based on random values. Therefore, to improve the efficiency of feature selection, we propose an intelligent agent and CRF based feature selection in this paper.

We describe our proposed approach for selecting necessary features for every layer and explain how some features were chosen over others. In our system, every layer is separately trained to detect a single type of attack category. We observe that the attack groups are different in their impact and hence, it becomes necessary to treat them differently. Hence, we have selected features for each layer based upon the type of attacks that the layer will detect based on training.

In Probe layer, the session duration and packets to be sent are considered for analysis. DoS layers checks the packets sent by nodes for finding the route through flooding. This is helpful to restrict flooding attacks. This R2L layer uses rules and monitors session duration, types of services requested and number of failed login attempts. The U2R layer monitors the number of files created, the number of commands used through the operating system and protocol type.

In this paper, we propose a new ICRFFSA to perform feature selection automatically by extending the existing CRF based feature selection algorithm [5] in which we select features for every layer randomly. Every layer is individually trained to detect a single type of attack category like DoS, Probe, U2R and R2L. Contribution values are assigned here for all features in that layer. Based on this cumulative contribution value, we set the threshold to find the exact features for all type of attacks. Selected features are stored in the set *F*. The decision agent takes a decision to select that feature to find the particular attack based on the cumulative contribution value of each feature by applying rules. If the particular feature cumulative contribution value is greater than threshold then, agent chooses the feature for identifying the particular attack.

Algorithm 1: Intelligent CRF based feature selection.

```
Input: The set S of all features
Output: F, the set of optimal features

// Let A be the set of features
Begin
   F={ }; // Initialize F to all null set.}
  for i=1 to n do
     Begin
        for j=1 to n do
           Begin
               f=random(S, CRF(s)) //Feature Selection
               CV=CV+Cond.prob($f_i$)//contributed value
               D=DA(CV, Decision)
               if decision==“yes” then F=F∪($f_j$)
                  Val=Check (CV >Threshold($A_i$)) and
                     Constraints (i, j))
                 if (val==true)
                  Display ($A_i$, j, Features(S));
                  Prevent ($A_i$, j);
               Else
                  Stop
           End
     End
End
```

In the existing work, the number of features used for detecting Probe, DoS, R2L and U2R attacks were 5, 9, 14 and 8. In order to minimize the number of features used in our proposed algorithm, we have excluded the features used for probe attack from the features used for detecting DoS, R2L and U2R attacks. Similarly, the features used for Probe and DoS attack have been excluded from R2L and U2R attack features. Thus, from the overall 41 features, we have selected only 5 features for Probe attack, 5 features for DoS attack, 11 features for R2L and 5 features for U2R by using intelligent agents.

These selected features are only used for detecing the four major attacks. Therefore, the main difference between our proposed ICRFFSA and the CRF based feature selection [5] is that we have selected only less number of features for detecting four types of attacks from dataset.

The main purpose of minimizing number of features in our proposed approach is to remove redundant features included to detect various attacks and to minimize the computation time taken to identify the intruders.

3.3. Classification Algorithm using LA

In this paper, we integrated the proposed feature selection algorithm called ICRFFSA with the existing classification algorithm known as LA classifier [5] to perform effective classification. This proposed algorithm receives the trained data with reduced features from the feature selection algorithm and they are validated based on the rules and facts present in the knowledge base.

Algorithm 2: LA based classification.

```
Input: Reduced features from layers
Output: Classified Records

Begin
    Read (n)  // Read the number of layers
    $R_s$={};   // Initialize the rule set $R_s$
    for i=1 to n do
    Begin
        ICRFFSA (DS, RDS)  //DS-Data set and RDS-
            Reduced Data set
       Create_rule ($C_i$, $R_i$, $L_n$);
      $R_s$[i]= $R_s$ [i-1]+ $R_i$
    End
    for j=1 to n do
       Begin
          Constraints (i, j, check_cons(i, j));
          Classify_output ($D_i$, $A_i$, label);
          Check (label==Attack || label==Normal)
          If ($A_i$==True)
          Find_type($A_i$, $C_i$);
         DA($A_i$, Decision);
       End
 End
```

Four types of attacks are identified in this model based on the rules present in the knowledgebase. After identifying the attackers, this classifier also finds the types of attacks.

3.4. Mathematical Analysis

- ***Theorem* 1**. The IDS proposed in this research work to detect Probe attack has $O((T_1)L^2KP)$ time complexity during intrusion detection process for *n* datasets which contains four types of attacks, where T_1 is the number of features used for determining probe attack, *L* is the number of labels, *K* is the number of training instances and *P* is the number of iterations.
- ***Theorem* 2**. The IDS proposed in this research work to detect DoS attack has $O((T_2\text{-}T_1)L^2KP)$ time complexity during intrusion detection process for *n* datasets which contains four types of attacks, where T_2-T_1 is the number of features used for determining DoS attack, *L* is the number of labels, *K* is the number of training instances, and *P* is the number of iterations.
- ***Theorem* 3**. The IDS proposed in this research work to detect R2L attack has $O((T_3\text{-}T_2)L^2KP)$ time complexity during intrusion detection process for n datasets which contains four types of attacks, where T_3-T_2 is the number of features used for determining R2L attack, *L* is the number of labels, *K* is the number of training instances, and *P* is the number of iterations.
- ***Theorem* 4**. The IDS proposed in this research work to detect U2R attack has $O((T_4\text{-}T_3)L^2KP)$ time complexity during intrusion detection process for n

datasets which contains four types of attacks, where T_4-T_3 is the number of features used for determining U2R attack, L is the number of labels, K is the number of training instances, and P is the number of iterations.

Time complexity for all the four types of attacks: Let 'n' be the number of dataset that is used for detecting the four types of attacks. The length of the sequences T_1, (T_2-T_1), (T_3-T_2), (T_4-T_3) denotes the length of the features used in each attack layer. (T_2-T_1) indicates the DoS features without the repetitive occurrence of probe attack features. (T_3-T_2) indicates the R2L features without the repetitive occurrence of probe attack features and DoS attack features. (T_4-T_3) indicates the U2R features without the repetitive occurrence of Probe, DoS, R2L attack features. From this, it is clearly stated that the time complexity to detect an attack from the given dataset is defined as $O((T_4)L^2KP)$. Therefore, the time complexity from the above four theorems is proved as shown below.

$$T(IDS)=O((T_2+(T_2-T_2)+(T_4-T_2))L^2KP)=O(T_4L^2KP)$$

Where $T(IDS)$ is the overall time complexity of the IDS to detect all types of attacks.

4. Results and Discussion

In this work, we used the benchmark KDD'99 cup intrusion data set for carrying out the experiments. This data set is drawn from DARPA intrusion detection system, which is prepared by the MIT Laboratory [6]. The data set contains about five million connection records as training data, and two million connection records as the test data. Each record is unique in the data set with 41 continuous and nominal features plus one class label.

In our experiments, we used 10 percent of the total training data and also 10 percent of the test data (with corrected labels), which are provided separately leading to 4, 94,020 training and 3, 11,029 test instances. The training data are either labeled as Normal, Probing, DoS, R2L, and U2R. Similarly, the test data are also labeled as normal probing, DoS, R2L and U2R.

4.1. Accuracy Calculating Measures

Accuracy of the IDS has been identified in this work using the following metrics, since the factors that are affecting the accuracy are precision, recall and F-measure which are defined below:

- *Precision*: The ratio of the data conditioned to the data relevant to the detection is defined as *Precision*. It is calculated by the formula.

$$Precision=\frac{TP}{TP+FP} \tag{3}$$

- *Recall*: The ratio between data that is relevant to the detection to the data that is successfully detected. It can be calculated by the formula.

$$Recall=\frac{TP}{TP+FN} \tag{4}$$

- *F-Measure*: The fraction of precision versus recall is said to be *F-Measure*. Usually, it the value of beta is set to be 1.

$$F\text{-}Measure=\frac{(1+\beta^2)*Recall*Precision}{\beta*(Recall+Precision)} \tag{5}$$

In this work, all the experiments have been carried out with the WEKA tool as the software. Moreover, the proposed method have been simulated in JAVA (in Intel core i3 with 3GB RAM) for the feature selection, accuracy calculation and the attack detection in the intrusion detection system.

4.2. Detecting the Attacks with All 41 Features

In this work, 10,000 normal records have been selected for conducting the experiments by applying intelligent rules and random probability. Moreover, all the Probe, DoS, R2L and U2R records have been considered in this experiment for training. Totally 15,000 and 64,759 records have been considered in this work for training and testing.

4.3. Detecting the Attacks with Feature Selection

When we carried out the experiments with the same set of data with feature selection, the feature selection algorithm selects the features shown in Table 1. These experiments have been carried out for detecting Probe, DoS, R2L and U2R attacks.

Table 1. List of selected features for all attacks.

Feature Number	Attacks	Feature Name
1	Probe	Duration
2		Protocol_type
3		Service
4		Flag
5		Src_bytes
1	R2L	Duration
5		Src_bytes
10		Hot
11		num_failed_logins
12		logged_in
13		num_compromised
17		num_file_creations
18		num_shells
19		num_access_files
21		is_host_login
22		is_guest_login
23	DoS	Count
34		dst_host_same_srv_rate
38		dst_host_serror_rate
39		dst_host_srv_serror_rate
40		dst_host_rerror_rate
1	U2R	Duration
5		Src_bytes
10		Hot
11		num_failed_logins
12		logged_in
13		num_compromised
17		num_file_creations
18		num_shells
19		num_access_files
21		is_host_login
22		is_guest_login

The results obtained for all these attacks are shown in Table 2 with respect to precision, recall and *F*-

Measure. Though the Layered Intelligent CRF (LICRF) uses the same data set as used in CRF, it increases the classification accuracy due to the use of selected features shown in Table 2.

Table 2. Performance analysis.

Attacks	Approach	Precision (%)	Recall (%)	F-measure (%)	Training Time (sec.)	Testing Time (sec.)
Probe	Proposed LAICRF	88.67	97.97	93.34	6.88	2.02
	Existing LACRF	88.19	97.82	92.73	6.91	2.04
DoS	Proposed LAICRF	99.99	97.23	98.72	25.22	14.83
	Existing LACRF	99.98	97.05	98.50	26.59	15.17
R2L	Proposed LAICRF	94.93	28.32	42.97	5.78	6.32
	Existing LACRF	94.70	27.08	42.08	5.30	5.96
U2R	Proposed LAICRF	55.23	63.56	59.03	0.99	2.86
	Existing LACRF	55.07	62.35	58.19	0.85	2.67

From Table 2, it can be observed that precision, recall and *F-Measure* are improved when the data are classified using the proposed LICRF. This is due to the fact that the Intelligent CRF uses intelligent agents for decision making.

However, the agent based decision making process increases the decision time. However, the proposed approach consumes less time for training and testing when it is compared with the CRF based approach.

4.4. Comparison of Results

In this section, we compare the proposed work with the existing classification methods namely LACRF and decision tree. From the experiments carried out, it has been observed that LAICRF performs better than the other two techniques in terms of detection accuracy. The main reason for this improvement is that the proposed LAICRF not only insist that the observation features to be independent but also uses of intelligent agents to make effective decisions. The comparison results are shown in Table 3.

Table 3. Comparison of detection accuracies.

Approach	Probe (%)	DoS (%)	R2L (%)	U2R (%)
Proposed LAICRF	98.83	97.62	32.43	86.91
Existing LACRF	98.60	97.40	29.600	86.300
C4.5 (Decision Tree)	80.80	97.00	4.600	1.800
Enhanced C4.5	81.5	97.12	12.57	6.24
Multilayer Perception	88.70	97.20	5.600	13.200

From Table 3, it is observed that the LICRF performs significantly better than the previously reported results. The most impressive part of the LICRF is the margin of improvement as compared with decision trees. LICRF provides an impressive attack detection rate of 98.83 percent for Probes, 97.62 percent for DoS, 32.43 percent for R2L and 86.91 percent for U2R. Therefore, we conclude that the proposed LICRFs is effective in detecting the Probe, the U2R and the R2L attacks as well as the DoS attacks.

The main reason for the improvement in detection accuracy for LAICRF is due to many reasons. Firstly, it uses only significant features. Secondly, it uses rules which are fired by the intelligent agents for effective decision making. Thirdly, the classification time is reduced due to feature selection. Fourthly, it uses CRF and LA in addition to rules. Finally, it considers temporal events in performing reasoning.

5. Conclusions and Future Enhancement

In this paper, we have proposed a new intrusion detection system that improves the detection accuracy and time efficiency for building the intrusion detection systems. For this purpose, we proposed a LAICRF model which is developed by combining an ICRFFSA and LA based classification algorithm for effective intrusion detection. In this work, rule and LA based classification methods have been used that significantly reduce the detection time and hence it increases the detection accuracy. Future extension to this work can be the use of temporal models to perform effective temporal reasoning based on time.

References

[1] Amiri F., Yousefi M., Lucas C., Shakery A., and Yazdani N., "Mutual Information-Based Feature Selection for Intrusion Detection Systems," *the Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1184-1199, 2011.

[2] Bace R. and Mell P., "Intrusion Detection Systems," *Computer Security Division, Information Technology Laboratory*, *Nat'l Inst. of Standards and Technology*, 2001.

[3] Chimphlee W., Abdullah A., Sap M., Chimphlee S., and Srinoy S., "A Rough-Fuzzy Hybrid Algorithm for Computer Intrusion Detection," *the International Arab Journal of Information Technology*, vol. 4, no. 3, pp. 247-254, 2007.

[4] Gupta K., Kotagiri R., and Nath B., "Conditional Random Fields for Intrusion Detection," *in Proceedings of the 21st International Conference Advanced Information Networking and Applications Workshops*, Niagara Falls, pp. 203-208, 2007.

[5] Gupta K., Nath B., and Kotagiri R., "Layered Approach using Conditional Random Fields for Intrusion Detection," *IEEE Transactions on Dependable and Secure Computing*, vol. 7, no.1, pp. 35-49, 2010.

[6] KDD Cup 1999 Intrusion Detection Data., available at: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, last visited 2010.

[7] Lafferty J., Mccallum A., and Pereira F., "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," *in Proceedings of the 18th International Conference Machine Learning*, San Francisco, pp. 282-289, 2001.

[8] Li J., Wang H., and Yu J., "Research on the Application of CRFs Based on Feature Sets in Network Intrusion Detection," *in Proceedings of IEEE International Conference on Security Technology*, Hainan Island, pp. 194-197, 2008.

[9] Mccallum A. and Sutton C., "An Introduction to Conditional Random Fields for Relational Learning," *Introduction to Statistical Relational Learning*, vol. 4, no. 4, pp. 267-373, 2006.

[10] Prema L. and Kannan A., "An Active Rule Approach for Network Intrusion Detection with Enhanced C4.5 Algorithm," *the Journal of Communications, Network and System Sciences*, vol. 1, no. 4, pp. 314-321, 2008.

[11] Sindhu S., Geetha S., and Kannan A., "Decision Tree Based Light Weight Intrusion Detection Using a Wrapper Approach," *Expert Systems with Applications*, vol. 39, no. 1, pp. 129-141, 2012.

[12] Wilk T. and Michal K., "Soft Computing Methods Applied to Combination of One-class Classifiers," *Neuro Computing*, vol. 75, no. 1, pp. 185-193, 2012.

Sannasi Ganapathy has completed Ms degree of computer applications from Madurai Kamaraj University in 2005, Ms degree of engineering in computer science and engineering from Anna University, Chennai in 2007. He received his PhD degree from Anna University, Chennai in 2013. He has published more than 20 papers in Journals and Conferences. He has more than 6 years of teaching experience in Engineering colleges. His areas of interest include data mining, network security, theory of computation and artificial intelligence.

Pandi Vijayakumar completed his PhD degree in computer science and engineering in Anna University Chennai in the year 2013. He Completed Ms degree of engineering in the field of Computer science and engineering in Karunya Institute of Technology affiliated to Anna University, in the year 2005. He completed his Bachelor of engineering in Madurai Kamaraj University, Madurai in the year 2002. He is presently working as an Assistant Professor at Anna University Chennai (University College of Engineering, Tindivanam), Chennai, India. His main thrust research areas are Key management in network security and multicasting in computer networks.

Palanichamy Yogesh is working as an Associate Professor in the Department of Information Science and Technology, College of Engineering Guindy Campus, Anna University, Chennai, India. He possesses bachelor and Ms degrees in computer science and engineering from Madurai Kamaraj University, India. He received his PhD degree from Anna University, Chennai, India. He has 5 years of experience in industry and 20 years of experience in academic and research. His areas of interests include data communication networks, mobile computing, multimedia communication and wireless security. Currently, he is working in the areas of internet of things and software defined networking.

Arputharaj Kannan is working as an Professor and Head in the Department of Information Science and Technology, College of Engineering Guindy Campus, Anna University, Chennai, India. He possesses Ms degree in computer science and engineering from Anna University, Chennai, India. He received his PhD degree from Anna University, Chennai, India. He has 8 years of experience in industry and 23 years of experience in academic and research. His areas of interests include data mining, networks security, software engineering and artificial intelligence. He has more than 185 publications in Journals and Conferences.