Middle-East Journal of Scientific Research 24 (4): 997-1003, 2016
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2016.24.04.23243

Fuzzified Ant System for Intrusion Detection in Enterprise Network

[1]L. Dhanabal and [2]S.P. Shantharajah

[1]Assistant Professor [SRG], Department of Computer Applications,
Kumaraguru College of Technology, Coimbatore – 641 049, Tamil Nadu, India
[2]Professor, Department of MCA, Sona College of Technology, Salem – 636 005, Tamil Nadu, India

Abstract: Ant colony algorithm chooses the shortest path based on the pheromones deposited by the ants that are travelling in the same path which then evaporates with the passage of time. Because of this nature the ant colony can only provide a local solution and cannot concentrate the nodes that are far away. Also to address the dynamic nature of optimization problems where certain number of particles is replaced with new ones at different frequencies the change of optimization procedure is needed. Hence to provide a balance between pheromone in changed scenarios and to obtain a global solution we have proposed fuzzified ant system which keeps track of the pheromone information in ant colony optimization and uses fuzzy technique to classifying the results. Proposed algorithm is tested for its efficiency in intrusion detection and results were compared with other traditional algorithms.

Key words: Signature IDS · Pheromone · Fuzzy system · Ant Colony Algorithm · Genetic Algorithm

INTRODUCTION

Computer networks provide an economic solution to various challenging problems including defense applications. Sensitive and mission critical data need to be transferred over the public network in a secure and reliable manner. Most of the general and enterprise networks are not so efficiently and effectively secured from intrusions. This alarming need gives rise to the designing of dynamic intrusion detection systems. The traditional prevention techniques such as user authentication, encryption of data and firewalls serves as the first line of defense for enterprise networks. User authentication can be easily compromised with the help of special tools; firewalls are vulnerable to errors and are ambiguous. Hence these techniques are unable to protect against malicious strange and inside attacks [1].

In general intrusion detection system is categorized in to misuse detection and anomaly detection. Misuse detection can identify intrusion attacks in relation to the known signatures of discovered vulnerabilities. However security experts are needed to define rules or signatures which limit the application of misuse detection to build Intelligent Intrusion Detection System (IDS). In case of anomaly detection, it has the capability to discover novel attacks without prior knowledge in the presence of generalized classification model to extract intrusion pattern and knowledge during training process. But anomaly detection often suffers from the high false positive rate (FPR) in classifying normal network traffic. To overcome this drawback several models has been proposed in recent years [2]. For example statistical approaches, neural networks, predictive analysis, fuzzy logic network and data mining. However each of the techniques suffers from several drawbacks in detecting all kinds of intrusion attempts on their own.

Here we have proposed a Fuzzified ant system applied for intrusion detection in dynamic attack environment. The algorithm first applies the ant colony technique on dataset provided. Ant system at each iteration selects a best rule. For continuous iterations the rules were stored in a rule base, from the rule base rule set is derived. In order to evaluate all the 41 attributes and for generating rules, the system may take more computation time and also to remove the redundant data, feature selection algorithm is applied. Here we have used sequential floating selection mechanism for feature selection. Attributes obtained after the feature selection technique is considered to be the important and used for evaluation. If the attribute combination appears more than

Corresponding Author: L. Dhanabal, Assistant Professor [SRG], Department of Computer Application, Kumaraguru College of Technology, Coimbatore – 641 049, Tamil Nadu, India.

once in the rule set it is eliminated. The remainder of this paper is as follows. Section II describes different author perspective in using ant colony based optimization system in detecting intrusions. Section III presents the proposed methodology. Section IV gives the experimental evaluation of the proposed approach followed by conclusion in Section V.

Related Work: Various techniques have been used in the past decade to detect intrusions. An emotional ant based approach is proposed to identify the possible pre-attack activities and subsequently behave with centralized intrusion detection mechanism. Ants are positioned in relevant locations in interconnected networks and the monitoring is done via the behavior of natural ant colonies [1]. The optimization of both accuracy and interpretability are necessary and should be taken in to account for building an anomaly based IDS. Hence a multi – objective genetic fuzzy intrusion detection system (MOGFIDS) has been proposed which applies an evolutionary agent based computation to generate and evolve accurate and interpretable fuzzy knowledge base for classification [2]

Genetic algorithm uses it crossover and mutation operators to classify the attack packets. Testing data is taken for evaluation and after applying the generic operators it is compared with training data. If there is a deviation particular packet is termed as attack and eliminated from the population. The approach is validated using NSL KDD cup dataset and the performance is measured in terms of detection rate (DR) and False Positive Rate [3]. Multiple ant-colonies are applied instead of single ant-miner. This algorithm works for searching best rules for mislead ant class while searching for rules of success class. Here each ant that belongs to a colony deposits distinct type of pheromone that affects only the ants belonging to same colony. Best rules are searched such that one rule per colony is identified. The best rule is added to rule set [4]. Learning mechanism reduces the weight of the training instances that are correctly classified by the classifier to form new rule. Hence the next rule generation focuses on fuzzy rules that are used for identifying uncovered or misclassified instances. At each iteration fuzzy rule that classifies the current distribution of training samples better than other rules is selected to be included in the classification fuzzy rule base. By doing so, high quality fuzzy if-then rules can be generated [5]. A modified CSOACN (Clustering based Self Organized Ant colony Network) is proposed such that the approach gains modification in both the supervised and unsupervised learning to interact with each other and efficiently. This also minimizes the training data set by allowing new data points added to training set dynamically [6]. A feature vitality based feature selection method is used to identify important features in the dataset. These attributes are then subjected to classification with naïve bayes classifier. The reduced feature classification using proposed feature selection mechanism has proven to be effective than other classifier coupled with commonly used feature selection techniques [7]. The ant colony algorithm can also be further improved by altering the pheromone updating rule in a way to reduce multiple scans in data storage and reduced count of candidate sets. The proposed ACO helps to design rules over frequent patterns in an effective manner [8].Fuzzy support vector machines are utilized for training and testing process. At first the kernel parameter is determined and at each trail in fuzzy vector machine membership parameter is selected. Then features are extracted and these features are used for training and solving decision function. In the testing phase classification is done using fuzzy support vector machine. Fuzzy classification returns the result as normal or attack [9].

Fuzzy Rule Based Systems: Fuzzy logic is a mathematical tool for dealing with uncertainty and mimics human brain to reach a conclusion. Using fuzzy logic, linguistic terms can be constructed such as low, medium, high and more. The pictorial representation of fuzzy logic is given as follows

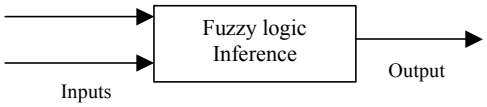

Fig. 1: Representation of Fuzzy logic

Example of linguistic fuzzy variable is shown in Fig.2. Here all rules were applied to the input vector and each rule matches the input pattern but varying degrees. The decision of output is handled by inference method. Due to the transparency of classification process we have used winner based methods, rule that achieves highest degree of match with input to classify the input. The reason behind the selection of fuzzy logic in solving intrusion detection is as follows. (i) Quantitative features are involved (ii) Security. An interval is used to denote the normal and abnormal values. Those values falling outside are regarded as anomalous irrespective of its distance to the interval [10].

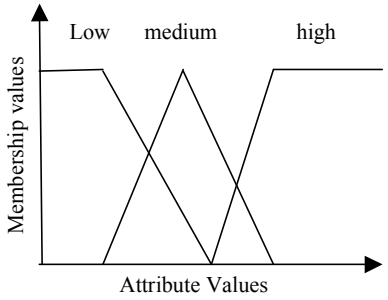

Fig. 2: Example of linguistic variable

Ant Colony Optimisation: Necessary terms for implementation of an ACO are: A problem definition to build solution using probabilistic transition rule. This determines the successor node to an ant for moving next. The transition rule depends on the heuristic value and pheromone level associated with a node. Normally this bias towards the higher probability nodes. Since the solution space is greater it does not guarantee that highest probability will get selected. Heuristic provides guidance to an ant in choosing next node for the path that is building. Constraint satisfaction method that force construction of feasible rules. If a simple if-then rule antecedents are constructed then at most only one fuzzy linguistic term from each variable should be selected. Fitness function to determine quality of solution builds by an ant. This measure how well the instances are classified by rule in training set. Pheromone update rule to specify how to modify pheromone levels of each node in the graph between iterations of ACO algorithm.

If termination condition is false
    For each ant: constructs a new solution
    Evaluate new solutions
    Update pheromone levels
Output best solution

Fuzzy Ant Colony System: Fuzzy Rule Ant colony system runs several ACO algorithms in parallel, each of these have its own problem graph, pheromone levels and heuristic values. After rules are created for each class, all combination of rules are formed in to a rule base and tested on the training set. To remove the redundancy of the data feature selection technique is used. The most effective feature selection technique is the sequential floating search methods.
There are two main categories of floating search methods:

•Forward selection technique
•Backward selection technique

In case of forward search technique the algorithm starts with a null feature set and for each step the best feature that satisfies some criterion function is included with the current feature set. This also verifies the possibility of improvement of the criterion if some feature is excluded. In this case worst feature is eliminated from the set. This method is called as Sequential Backward Selection. In our method rules generated by ACO are eliminated in order to remove the repeated data so we have used SFBS algorithm. Best performing rules in the rule base with the rule describing a specific class is used to update pheromone levels of associated ACO. Block diagram for the proposed architecture is shown in Fig. 3.

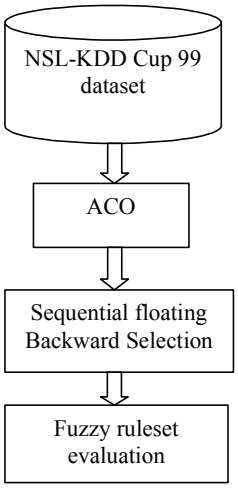

Fig. 3: Block diagram for proposed Fuzzy ACO

The algorithm for constructing fuzzy ant colony system is given below

ALGORITHM
Divide the dataset in to smaller number of classes.
For i = 1 to n
    For each class
        Each ant constructs rule
        Combine rules and form a rulebase
                For each combined rulebase
                Evaluate rulebase
                update pheromone levels
                generate best rulebase as output
        endfor
    endfor

Rule Construction: FACS has the ability to build propositional rules (IF TEMPERATURE is cool AND Wind is Windy THEN Weightlifting), propositional rules

with internal disjunction (IF TEMPERATURE is Cool OR Mild and wind is windy THEN Weightlifting) and propositional rules with negated terms. While creating rule antecedent, ant traverses the graph through each node. Here node represents the term that may be added OUTLOOK = Sunny. For rules with negated terms the graph produces double the nodes eg. OUTLOOK = NOT Sunny. The selection of next node depends on the heuristic value and the pheromone level associated with the node. It is made probabilistically and biased towards terms that have relatively highly heuristic and pheromone values. After the selection and before the term is actually added to rule antecedent the system must ensure that minimum number of appropriate class instances from training set is converted to avoid over fitting to training data. Fuzzy rule that describes specific class is said to be matched with instance if

- The rule and instance belong to same class
- Degree of match between conditional parts of rule and instance is equal to or greater than a pre-defined value of a parameter called construction Threshold.

Heuristic: Heuristic guide ants when selecting terms is based on fuzzy sub-sethood values [6]. A fuzzy set A is assigned with a degree when fuzzy set A is a subset of another fuzzy set B.

$$S(A,B) = \frac{M(A \cap B)}{M(A)} = \frac{\sum_{u \in U} \min\ (\mu_A(u), (\mu_B(u))}{\sum_{u \in U} \mu_{A(u)}} \quad (1)$$

Where u an instance from training set U is, A is a class label and B is the term added to rule antecedent. The heuristic value of a term j-n_j gives the measurement of importance of term in specific class description. Here n denotes the heuristic values and j contains n heuristic values associated. An ACO that finds rule for describing particular class will use the appropriate term heuristic values (associated with the class). In case of negated term heuristic value is the complement of non-negated term $n_{Nor\text{-}j} = 1\text{-}n_jz$.

Updating Pheromone: Here the pheromone is made to deposit in the edges of the graph. Because of the fact that nodes are not important in constructing rule antecedent and are not selected in the order. For example consider the rule given below

If TEMPERATURE IS Mild AND WIND is Windy THEN Weightlifting
IF WIND is windy AND TEMPERATURE is mild THEN Weightlifting

For the initial run of ACO all nodes in the graph have equal amount of pheromone which is set to inverse of the number of nodes. For subsequent iterations the pheromone level changes. At the end of each iteration the rules are evaluated. Let R be the best rule selected in the iteration of particular ACO, the pheromone levels are increased as

$$\tau_j(t+1) = \tau_j(t) + \left(\tau_j(t)\ X\ Q\right), \forall_j \in R \quad (2)$$

At time (t+1) each term j present in rule R gets its pheromone level increased in proportion to quality Q of the rule. Then normalization of pheromone levels of all terms are done by dividing each pheromone level by sum of all pheromone levels that results in decrease of the pheromone levels of terms not in R. The terms that are increased have the chances likely to get selected.

Transition Rule: Ant select terms based on the transition rule that is probabilistic but biased towards terms that have higher heuristic and pheromone levels. The probability that the term j is selected by ant m when building its rule during iteration t is given by

$$P_j^m(t) = \frac{[n_j] \times [\tau_j(t)]}{\sum_{i \in l_m} [n_i] X\ [\tau_j(t)]} \quad (3)$$

Where l_m is the linguistic term set that are considered for inclusion in rule antecedent being built by ant m. For propositional rules with internal disjunction l_m excludes terms that are present in current partial rule antecedent and the terms that are already considered but are below the required number of instances (minInstPerRule). If simple propositional rules or rules that include negated terms, l_m will exclude other variable within the domain of linguistic variables that already have a term present in the rule antecedent.

Rule Evaluation: Instead of evaluating rules separately it is done at the end of each iteration when each class has produced set of rules, a rule describing one class is combined with one rule describing each of the other classes and together to classify the training set. For each instance u,

- Calculate condition match for instance u.
- Instance u is assigned with the class of the rule that has highest condition match.

Rules that obtain highest accuracy in the rule base are used for updating the pheromone levels for the next run in ACO. Let us consider that all rule bases are created and evaluated after an iteration by combining a rule from one class with one rule from each of the other classes. So the total number of evaluations is calculated by

$$Num_{Iterations} * NumAnts^{numClasses} \quad (4)$$

Where $Num_{Iterations}$ is the, *NumAnts* is the, *numClasses* is the number of class labels in the training set. Generally not all possible combinations are formed, only few representatives are selected from each population and used to form different combinations of knowledge or a combination of both.

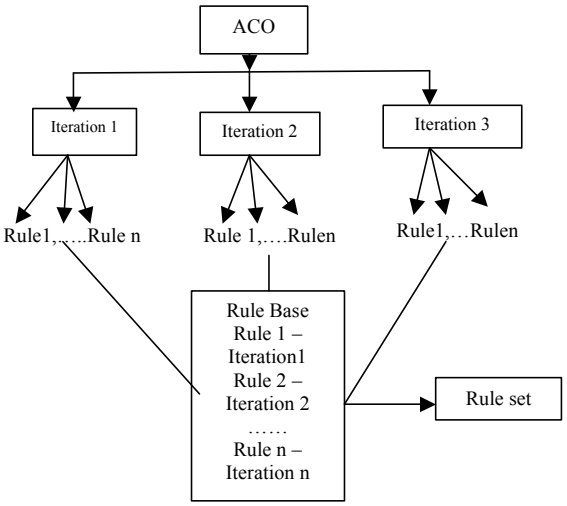

Fig. 4: Rule inducing algorithm

Experiment and Analysis: This section presents the datasets, comparison of fuzzy based ant colony algorithm against other algorithms, parameter settings.

The Datasets: NSL KDD Dataset: This dataset arises from the improvements on KDD'99 dataset [10]. It is a very popular dataset and most widely used for evaluation of intrusion detection system. This consists of 4,900,000 single network connection vectors with 41 attributes and class attribute labeled as normal or attack with exactly one specific attack type. The attacks can belong to one of the following classes. (i). Denial of Service Attack (DoS): This happens when attacker open connections directly or indirectly to computer system thus making it busy and deny legitimate connection. (ii). User to Root Attack (U2R): This class of attack arises when the attacking parties have access to user account gained legally or illegally and exploiting system weakness. (iii) Remote to Local Attack (R2L): When attacker doesn't have account but able to send packets from remote location and exploit system vulnerability to gain access. (iv). Probing Attack: Here the attacker gathers information about network of computer system for finding the way to gain access

Experimental Result: We have downloaded the dataset with all records including intrusion and normal. The records were 125 973 in total with 41 attributes and two classes normal, attack. It is necessary to identify the important attributes for classification. Hence we have used sequential floating selection and reduced 41 attributes to 12 attributes. The reduced attributes are given as follows:

Protocol_type(2),Service(3),
dst_bytes(6),
same_srv_rate(29),
dst_host_same_srv_rate(34),
dst_host_diff_srv_rate(35),
dst_host_same_src_port _rate(36),
dst_host_srv_diff_host_rate(37),
dst_host_serror_rate(38),
dst_host_srv_serror_rate(39),
dst_host_rerror_rate(40),
dst_host_srv_rerror_rate(41)

Fig. 5: Attribute reduction

The NSL KDD dataset records roots from three protocols namely tcp, udp and icmp and these have 66 total number of services. Each of these services has different number of categories and hence divided in to smaller categories. For tcp protocol we have 62 services and udp has 5 services and icmp has 6 services. Thus normal records are divided in to 73 categories. From these 73 categories we are going to form fuzzy interference system. In our experiments we have noted that value of des_bytes is also zero for tcp and ftp data services. Also values in count and serv_count are not exceeding 30. After defining ranges as low, medium and high the fuzzy rule in case of ftp and tcp services can be expressed as follows:

If dst_bytes is low and If count is low and if srv_count is low then the intrusion is normal.

The result for protocol is stored separately and a best rule which covers all the condition is chosen and sent to rule base. Likewise for all the 12 features, each feature is divided in to separate categories and rules are mined. Best rule from each feature is selected and stored in rule base. From the rule base a best rule is obtained which can be used for detecting type of attack and also attack or normal.

Performance Metrics: For a better intrusion detection system, the rate of detection must be higher and the rate of false alarm rate must be lower. Hence to estimate the efficiency of our system, two important formulas were used for evaluation

$$Detection\ Rate = \frac{Total\ number\ of\ detected\ attacks}{Total\ number\ of\ attacks\ detection} X\ 100\% \quad (5)$$

$$False\ alarm\ rate = \frac{(Total\ number\ of\ normal\ packets)}{(Total\ number\ of\ misclassified\ packets)} X\ 100\% \quad (6)$$

All the experiments were conducted on a windows based computer with Pentium core processor 2.0 GHZ, 250 HDD, 2GB of RAM. To implement our proposed approach the program was written in Java Runtime Environment 7 (jre 7.0). The proposed approach is compared with the traditional ACO. It is shown that the proposed algorithm achieves higher detection rate also reduces false alarm rate since decision making is done using fuzzy. Fig. 5 shows the detection rate and false alarm rate comparison between ACO and Fuzzy based ACO. Further to validate the proposed approach with some recent proceedings a comparison have been made and results were observed. To compare other approaches we have used Accuracy. Accuracy is defined as the proportion of data correctly classified that is True Positive (TP) and True Negative (TN).

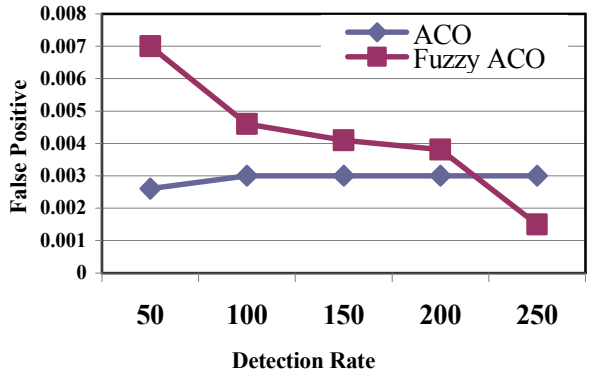

Fig. 5: Detection rate vs False Positive Rate comparison between ACO and Fuzzified ACS

Table 1: Accuracy Comparison of Algorithm

Algorithm	Recall	Precision	F-Measure	Accuracy (%)
C4.5	98.38	74.75	84.96	92.04
Naïve Bayes	55.47	43.33	48.65	76.45
SVM	97.99	74.15	83.94	92.54
ACO	81.88	73.42	71.43	91.83
Fuzzy ACO	96	73.42	89.29	94.33

CONCLUSION

In this paper we have proposed a Fuzzified ant colony system for intrusion detection. For initial evaluation ACO is made to run on the training dataset. The aim of this work is to utilize the fuzzy decision system in detecting transmitted packets as normal or attack. For each run ACO produces a rule as output. For all runs each best rule is stored in a rule base. If more than one rule reflects the previous iteration rules, then those are removed and process continues. As a result a set of best rule are collected in rule base. These rules are used by fuzzy system in classification of intrusion. The proposed methodology is suitable for both anomaly and misuse detection this technique is scalable and accurate since false positive rate are reduced. Computer simulations on NSL-KDD datasets demonstrate high performance of Fuzzified ant colony system for intrusion detection. Our future directions are to apply various feature selection techniques and test the performance since certain important attributes are removed during particular selection technique.

REFERENCES

1. Soumya Banerjee, Crina Grosan and Ajith Abraham, 2005. IDEAS: intrusion detection based on emotional ants for sensors. Intelligent Systems Design and Applications, 2005. ISDA'05. Proceedings. 5th International Conference on. IEEE.
2. Chi-Ho Tsang, Sam Kwong and Hanli Wang, 2007. Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection, Pattern Recognition, 40(9): 2373-2391.
3. Sazzadul Hoque, Mohammad, 2012. An implementation of intrusion detection system using genetic algorithm. arXiv preprint arXiv, pp: 1204-1336.
4. Kolias, Constantinos, Georgios Kambourakis and M. Maragoudakis., 2011. Swarm intelligence in intrusion detection: A survey, computers & security, 30(8): 625-642.

5. Abadeh, M. Saniee, 2007. Intrusion detection using a hybridization of evolutionary fuzzy systems and artificial immune systems, Evolutionary Computation, IEEE Congress on. IEEE.
6. Feng, Wenying, 2014. Mining network data for intrusion detection through combining SVMs with ant colony networks, Future Generation Computer Systems, 37: 127-140.
7. Mukherjee, Saurabh and Neelam Sharma, 2012. Intrusion detection using naive Bayes classifier with feature reduction, Procedia Technology, 4: 119-128.
8. Sundaramoorthy, Suriya and S.P. Shantharajah, 2014. An improved ant colony algorithm for effective mining of frequent items, Journal of Web Engineering, 13(3-4): 263-276.
9. Yanjun Long, Jianquan Ouyang and Xinwen Sun, 2013. Network Intrusion Detection Model based on Fuzzy Support Vector Machine, Journal of Networks, 8(6): 1387-1394.
10. Kosko, Bart, 1986. Fuzzy entropy and conditioning. Information Sciences, 40(2): 165-174.