IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

VOL. E100-A NO. 1
JANUARY 2017

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY
The Institute of Electronics, Information and Communication Engineers
Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

IEICE TRANS. FUNDAMENTALS, VOL.E100–A, NO.1 JANUARY 2017

PAPER Special Section on Cryptography and Information Security

Another Fuzzy Anomaly Detection System Based on Ant Clustering Algorithm*

Muhamad Erza AMINANTO[†a)], **HakJu KIM**[†b)], **Kyung-Min KIM**[†c)], ***Nonmembers***, ***and*** **Kwangjo KIM**[†d)], ***Member***

SUMMARY Attacks against computer networks are evolving rapidly. Conventional intrusion detection system based on pattern matching and static signatures have a significant limitation since the signature database should be updated frequently. The unsupervised learning algorithm can overcome this limitation. Ant Clustering Algorithm (ACA) is a popular unsupervised learning algorithm to classify data into different categories. However, ACA needs to be complemented with other algorithms for the classification process. In this paper, we present a fuzzy anomaly detection system that works in two phases. In the first phase, the training phase, we propose ACA to determine clusters. In the second phase, the classification phase, we exploit a fuzzy approach by the combination of two distance-based methods to detect anomalies in new monitored data. We validate our hybrid approach using the KDD Cup'99 dataset. The results indicate that, compared to several traditional and new techniques, the proposed hybrid approach achieves higher detection rate and lower false positive rate.

key words: *unknown attacks, unsupervised learning, ant clustering algorithm, fuzzy logic*

1. Introduction

In recent years, various schemes have been proposed for computer network protection from malicious party. Intrusion Detection System (IDS) has emerged as one of the most common parts for every network security infrastructures [1]. IDS is usually classified into misuse detection and anomaly detection [2]. Misuse detection techniques usually utilize signature-based approach to detect attacks. The approach is intended to identify known attack patterns. Although misuse detection techniques are most commonly used in practice [2], these techniques have three significant drawbacks [3]. Firstly, it is incapability to detect unknown attacks since it considers known signature of attacks. Secondly, it is burdensome when signatures of attacks need to be updated frequently to maintain the performance of misuse detection. Thirdly, it is difficult to develop appropriate signatures for misuse detection because attackers usually combine previous attacks, so called polymorph attacks [3]. There are two possible ways to solve these drawbacks. The first one is to generate several signatures that cover all possible variation of attacks. Another one is to generalize the signatures which leads to higher False Positive Rate (FPR) [3]. On the other hand, anomaly detection focuses on detecting unusual activity patterns in the observed data [2]. Anomaly detection approach usually deals with statistical analysis and data mining problems [4], which are able to detect novel attacks without prior knowledge since the classification model has the generalization ability to extract intrusion pattern and knowledge during the training phase [4].

It is difficult and inefficient to obtain bulk of labeled network connection records for supervised training. The clustering analysis has emerged as an anomaly intrusion detection approach in recent years [4]. Clustering is an unsupervised data exploratory technique that partitions a set of unlabeled data patterns into groups or clusters such that patterns within a cluster are similar to each other but dissimilar to other clusters' pattern [4]. Ant Clustering Algorithm (ACA) is one of the most widely used clustering approaches which is originated from swarm intelligence. ACA is an unsupervised learning algorithm that is able to find near-optimal clustering solution without predefined number of clusters needed [4]. However, ACA is rarely used in intrusion detection as the exclusive method for classification. Instead, ACA is combined with other supervised algorithms such as Self Organizing Map (SOM) and Support Vector Machine (SVM) in order to provide better classification result [1]. Based on Karami et al. [5] experiments, fuzzy logic approach can be used to improve classification result.

In this paper, we propose a novel hybrid IDS scheme based on ACA and fuzzy logic approach. Our proposed scheme comprises two phases, training and classification. We apply ACA for training phase and fuzzy logic approach for classification phase. We choose fuzzy approach as classification phase, because fuzzy approach can reduce the FPR with higher reliability in determining intrusion activities [5]. The experimental results on the KDD Cup'99 dataset demonstrate that our scheme can provide accurate and robust clustering and classification solution with high Detection Rate (DR) and low FPR. Our contribution in this paper is twofold. First, we examine the hybrid IDS approach published by Karami et al. [5] with different clustering algorithms. We employ ACA instead of Particle Swarm Optimization (PSO) and K-means algorithm. Second, we adopt Karami's fuzzy rule [5] with different fuzzy membership functions. In the end of this paper, we compare our proposed scheme perfor-

Manuscript received March 29, 2016.
Manuscript revised July 25, 2016.

[†]The authors are with the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea.

[*]This paper was presented in part at 2016 Symposium on Cryptography and Information Security, Kumamoto, Japan, Jan. 19–22, 2016.

a) E-mail: aminanto@kaist.ac.kr
b) E-mail: ndemian@kaist.ac.kr
c) E-mail: saza12345@kaist.ac.kr
d) E-mail: kkj@kaist.ac.kr (Corresponding author)

DOI: 10.1587/transfun.E100.A.176

mance with another similar work but different algorithms, Hosseinpour et al. [6]. Our experiment results show that our novel proposed scheme can outperform the previous scheme.

This paper is organized as follows: Section 2 provides previous publications which inspire us to work on this problem. Section 3 explains the background of this paper such as IDS, ACA, Fuzzy logic and KDD Cup'99. Section 4 describes our proposed method. Section 5 contains experimental results and analysis. Finally we conclude in Sect. 6.

2. Related Work

There are many different IDS schemes that use hybrid approaches to integrate the ant-based clustering model with other machine learning and soft computing algorithms [4]. They include the cellular automata [7], K-means algorithm [8], self-organizing map [9], fuzzy C-mean algorithm [10] and fuzzy if-then rule system [11]. Those schemes except Abadeh et al. [11], are different from our proposed scheme since they are not using fuzzy if-then rule system. Fuzzy if-then rule system can outperform other algorithms because it is able to construct a model from crisp value with proper meaning. In addition, a boundary between normal and attack instances is not clear. So, fuzzy logic is suitable for the unclear boundary. Meanwhile, our proposed scheme differs from Abadeh et al. [11] by the goal of the IDS, which their intention is to aim misuse detection while we aim anomaly detection.

One of the most recent hybrid IDS was proposed by Karami et al. [5] at 2014. Unlike Karami's [5] work which focuses on Content-Centric Networks, we aim ordinary networks. They proposed a hybrid IDS system using PSO-K-means algorithm and fuzzy approach. Basically, their scheme contains two phases, training and classification. They applied a novel combination of PSO and K-means algorithm for training phase in order to provide better clustering result. However, according to Kolias et al. [1], ACA-based IDS provides higher DR than other IDS schemes, including PSO and K-means algorithm. Thus, in this paper, we investigate the effectiveness of using ACA instead of PSO and K-means algorithm as a clustering method. For the classification phase, Karami et al. [5] utilized fuzzy if-then rules to give a fuzzy detection of normal and abnormal results in the new monitoring data set that does not appear in the training set. They claimed that by using fuzzy rules, FPR can be reduced. In addition, Hosseinpour et al. [6] published a similar hybrid approach which contains clustering and detection phase. Hosseinpour et al. [6] use Artificial Immune System (AIS) combined with K-means clustering and Density Based Spatial Clustering of Applications with Noise (DBSCAN) while we use ACA combined with fuzzy logic approach.

3. Preliminaries

In this section we describe general overview of related terms such as IDS, ACA, fuzzy logic approach, and KDD Cup'99 Dataset.

3.1 IDS

According to the guidance from National Institute of Standards and Technology (NIST) [12], intrusion detection is defined as "the process of monitoring the events occurring in a computer system or network and analyzing them for signs of possible incidents, which are violations or imminent threats of violation of computer security policies, acceptable use policies, or standard security practices". IDS is a system that is designed to perform all the procedures relevant to intrusion detection [1]. There are many varieties of techniques and frameworks that are implemented in IDS. In general, IDSs are comprised of:

- A set of sensors that collects both malicious and normal data from the monitored system [1]. Sensors may be part of the system or external devices depends on the type of IDS.
- An analyzer engine that collects all data from sensors and analyses them. The engine usually located in central point. The engine has capability to reconfigure the protected system properly if the results of the analysis indicate an intrusion occurred [1].
- A report system that alerts the responsible party when suspicious events occurred [1].

The IDS based on misuse detection contains signatures of known attacks. The list of signatures is utilized by the analyzer engine during the data analysis step and must be frequently updated to include the signatures of the latest attacks. In addition, several IDSs might have response engine [1]. The response engine might be able to take actions automatically or manually by the command of the administrator.

There are many different classifications of the existing IDS based on different criteria. One distinction can be made in terms of the location of the active sensing components of the IDS. Based on this attribute, the IDS can be classified into host-based and network-based [1]. In host-based approaches the sensor components are installed on each host that requires protection. Meanwhile, a network-based IDS monitors the network that contains the hosts of interest. This type of IDS is usually installed on multiple dedicated machines, which are possibly different from the protected hosts, and monitors the network traffic.

Other categorization is based on the adopted data analysis approach. In this case, IDS may fall into one of the two main groups: misuse detection and anomaly detection [1]. The first approach examines the activity of the entire infrastructure for patterns of misuses known beforehand, usually referred to as attack identities. On the other hand, anomaly detection approaches analyze the behavior of the protected system over time toward extracting an approximate estimation of what behavior is considered normal or legitimate. Any action that significantly deviates from that kind of behavior is considered an attack.

In general, an IDS must be able to identify intrusions

with high accuracy. At the same time, an IDS should be able to distinguish between legitimate and intrusive actions. These two criteria have been associated with two performance evaluation variables: DR and FPR. Kolias et al. [1] defined DR as the ratio of the number of correctly detected attacks to the total number of attacks. Meanwhile, FPR is defined as the ratio of the number of normal connections that are classified incorrectly as attacks to the total number of normal connections [1]. An IDS usually tries to maintain high DR and keep FPR as low as possible in the same time.

3.2 ACA

ACA simulates ant random walks on a two-dimensional grid which is all data objects are spread randomly [13]. Unlike the dimension of the input data, each data instance is randomly projected onto a cell of the grid. A grid cell can indicate the relative position of the data instance in the two-dimensional grid. The general idea of ACA is to keep similar items in their original N-dimensional space. Vizine et al. [13] assumed that each site or cell on the grid can be resided by at most one object, and one of the two following situations may occur: (i) one ant holds an object i and evaluates the probability of dropping it in its current position; (ii) an unloaded ant evaluates the probability of picking up an object. An ant is selected randomly and can either pick up or drop an object at its current location [13].

The probability of picking up an object increases by disparity among objects in the surrounding area and *vice versa*. In contrast, the probability of dropping an object increases by high similarity among objects in the surrounding area. Vizine et al. [13] defined $d(i,j)$ in Eq. (1) as the Euclidean distance between objects i and j in their N-dimensional space. The density distribution function for object i, at a particular grid location, is defined by Eq. (1) as follows:

$$f(i) = \begin{cases} \frac{1}{s^2} \sum_j (1 - d(i,j)/\alpha) & f(i) > 0 \\ 0 & Otherwise, \end{cases} \quad (1)$$

where s^2 is the number of cells in the surrounding area of i and α is a constant that depicts the disparity among objects. The $f(i)$ might reach maximum value when all the sites in the surrounding area are occupied by similar or even equal objects. The probability of picking up and dropping an object i is given by Eqs. (2) and (3), respectively:

$$P_{pick}(i) = (\frac{k_p}{k_p + f(i)})^2, \quad (2)$$

$$P_{drop}(i) = \begin{cases} 2f(i) & f(i) < k_d \\ 1 & Otherwise, \end{cases} \quad (3)$$

where the parameters k_p and k_d are threshold constants of the probability of picking up and dropping an object, respectively. A loaded ant considers the first empty cell in its local area to drop the object, since the current position of the ant may be pre-occupied by another object [13].

Tsang et al. [4] define two variables: intra-cluster and inter-cluster distance in order to measure ACA performance. High intra-cluster distance means better compactness. Meanwhile, high inter-cluster distance means better separateness. A good ACA should provide minimum intra-cluster distance and maximum inter-cluster distance in order to presents the inherent structures and knowledge from data patterns.

3.3 Fuzzy Approach

Fuzzy approach is a method of representing the ambiguity and imprecision of a logic that is usually only 1 and 0 in digital form. This property of fuzzy set is appropriate to be exploited as anomaly detector for two main reasons [14]:

1. The anomaly detection problem usually includes several numeric attributes in collected data and various derived statistical measurements. Constructing models on numeric data directly might cause many errors in detection.
2. The security term itself involves fuzziness, because the boundary between normal and abnormal is not well defined [5].

Fuzzy logic is usually occupied together with other popular data mining techniques in order to detect outlier. Malicious behavior is naturally different from normal behavior, then abnormal behavior might be considered as outlier. Fuzzy logic can help to construct more abstract and flexible pattern for intrusion detection and thus greatly increase the adaption ability of the detection system [5]. Therefore, the fuzzy approach can achieve reliable intrusive activities detection with a quite low FPR, since we can classify properly any data instance based on the distance to other clusters. The distance of any data instance to clusters represents a similarity, the nearer the distance means that the data instance is similar to that cluster.

3.4 KDD Cup'99 Dataset

KDD Cup'99 dataset has been the most widely used dataset for the evaluation of anomaly detection methods [15]. The dataset is based on the data captured in DARPA'98 IDS evaluation program. KDD Cup'99 dataset consists approximately 4,900,000 single connection instances. Table 1 shows the packet distribution of KDD Cup 99 dataset [16]. Each instance contains 41 features and is labeled as either normal or attack instance. The dataset provides four distinct attack types as follows:

1. **Probing Attack**: an attacker attempts to collect information about computer networks in the purpose of bypassing the security controls. An example of probing attack is port scanning.
2. **Denial of Service (DoS) Attack**: an attack in which the attacker prevents legitimate users from accessing authorized data. The attacker made computing resources too exhausted to handle legitimate requests by flooding the

Table 1 Packet distribution of KDD Cup'99 dataset.

Type	# of Packets	Proportion (%)
Normal	972,781	19.86
Probe	41,102	0.84
DoS	3,883,370	79.28
U2R	52	0.00
R2L	1,126	0.02
Total	4,898,431	100

network with unnecessary packet requests. An example of DoS attack is syn flood attack.

3. **User to Root (U2R) Attack**: an attacker starts the attack with accessing to a normal user account on the system. Then, the attacker exploits the vulnerability to gain root access to the system. An example of U2R attack is *xterm* exploitation.
4. **Remote to Local (R2L) Attack**: This kind of attack is executed by an attacker who has the ability to send packets to a machine over a network but does not have an account on that machine. The attacker exploits some vulnerabilities to gain local access as a user of that machine remotely. An example of R2L attack is *ftp_write* exploitation.

4. Our Approach

This section describes the details of our approach. Basically, our approach consists of two main phases, training and classification. Similar to other approaches, our scheme is illustrated in Fig. 1. Each phase is also described as follows:

4.1 Training Phase

The training phase implements ACA in order to clusters the network traffic. ACA incorporates several initialization steps. Thus, it needs several input parameters such as the size of grid area, the number of ants, the size of local area, and threshold constant. After the clustering phase finished, we refer to Kim et al. [17] for labelling each data instance according to resulted clusters. The training phase passes this labeled dataset to the Fuzzy Inference System (FIS) in the classification phase.

4.2 Classification Phase

The labeled dataset from the training phase are sent to the second phase for anomaly detection when new data arrives. In the classification phase, a fuzzy decision approach is applied to detect attacks. We calculate Euclidean distance of each test data to all clusters as an input to the FIS. Equation (4) shows the Euclidean distance of two points x and y, where x_i and y_i represent features of each test data instance and training data instance within cluster, respectively. In this case, N represents total features in KDD Cup'99 dataset [18] which has 41 features on each data instances.

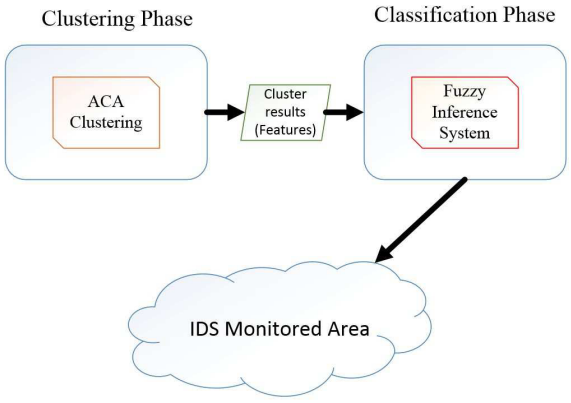

Fig. 1 Our scheme.

$$Distance(x, y) = \sqrt{\sum_{i=1}^{N}(x_i - y_i)^2}. \quad (4)$$

We deploy a combination of two distance-based [5] methods, i.e., nearest to normal and abnormal:

1. **Nearest to Normal**: The distance between a test data instance and each cluster is calculated using average linkage of Euclidean distance. Average linkage approach considers small variances [5], because the approach considers all members in the cluster rather than just a single member. In addition, the average linkage approach tends to be less influenced by the extreme values than other distance methods [19]. A test data instance is classified as nearest to normal when it has minimum average Euclidean distance among clusters labeled as normal cluster and *vice versa*. This distance-based classification allows us to detect whether normal or abnormal traffic by comparing features similarity that is listed in the training data set.
2. **Nearest to Abnormal**: Similar as before, we also calculate average linkage of Euclidean distance in order to find the minimum distance to abnormal cluster. A test data instance is classified as nearest to abnormal when the data instance has minimum average Euclidean distance among clusters labeled as abnormal cluster and *vice versa*.

The proposed fuzzy detection method consists of two inputs (nearest to normal and abnormal), one output, and four main parts: fuzzification, rules, inference engine, and defuzzification [5]. In fuzzification step, a crisp set of input data is converted to a fuzzy set using fuzzy linguistic terms and membership functions. Next, we construct rule base. Afterwards, an inference is made and combined based on the set of rules. In defuzzification step, the results of fuzzy inference are mapped to a crisp (non-fuzzy) output using the output membership functions. Finally, if the crisp output is bigger than a predefined threshold, a test data instance is considered as an abnormal instance, otherwise it is a normal instance.

5. Evaluation

5.1 Performance Measurement

In order to evaluate the performance of our proposed approach, we use DR, FPR and False Negative Rate (FNR). We calculate DR by number of attack instances detected as attacks divided by total of attack instances included in test dataset. We have 393 data of attack instances. FPR is legitimate packet detected as a malicious packet. FPR is calculated by number of legitimate instances detected as attack instances divided by total normal (legitimate) instances included in the data test. We are incorporating 19,268 legitimate instances. Lastly, FNR represents number of attacks that is unable to be detected by our proposed approach. The FNR value can be calculated by one minus DR.

5.2 Clustering Phase

We need to extract the KDD Cup'99 dataset in order to get appropriate traffic data that reflects real network traffic. Also, we need to prepare two sets of data: training and test dataset. Table 2 shows the training dataset that we used as an input to ACA in clustering phase. As mentioned in Sect. 4.1, ACA needs several input parameters, we define the parameters as follows:

- Size of grid area: 600 X 600 size of 2D plane,
- Number of ants: 1000 ants,
- size of local area: 3 X 3 local area,
- Threshold constant: 15.

ACA provides clusters that consolidate similar feature data instances. We label big and small size clusters as normal and attack clusters, respectively. Figure 2 shows the clustering result. The big colony is assumed as benign instances. We prepare the test dataset as shown in Table 3. The dataset is processed by measuring the Euclidean distance between each data instance in the test dataset and all data instances in the training dataset. Then, we define two values: closest to normal and abnormal, as an input parameter to the Fuzzy Inference System (FIS).

5.3 Classification Phase

We use MATLAB fuzzy logic toolbox for FIS-based intrusion detection. The classification phase is structured by following components:

1. Two fuzzy sets of input variables: nearest to normal and abnormal; nearest to normal membership are: Very Close, Close, Average, Far, Very Far; nearest to abnormal membership are: Far, Average, Close.
2. A fuzzy set of output variable: Alarm; alarm membership function: Normal, Less Prone, High Prone, Abnormal.
3. Fuzzy Membership Functions (MF): Figs. 3, 4 and 5 show fuzzy membership function for nearest to normal input, abnormal input and alarm output, respectively.
4. Fuzzy rules: Table 4 shows complete fuzzy rules while Table 10 shows more detailed fuzzy rules.
5. Inference: We use Mamdani fuzzy inference by fuzzy set operation as max and min for OR and AND, respectively [5]. Fig. 7 shows fuzzy inference rule in 3D form.
6. Defuzzifier: We use Center of Gravity algorithm as shown by Eq. (5).

$$CenterOfGravity = \frac{\int_{min}^{max} u * \mu(u)d(u)}{\int_{min}^{max} \mu(u)d(u)}, \quad (5)$$

where u represents the output variable, μ denotes the membership function after accumulation, and *min* and *max* are lower and upper limits for defuzzification, respectively.

Table 2 Our training dataset.

Type	# of Packets	Proportion (%)
Normal	78,101	98.00
Probe	398	0.50
DoS	761	0.96
U2R	35	0.04
R2L	398	0.50
Total	79,602	100

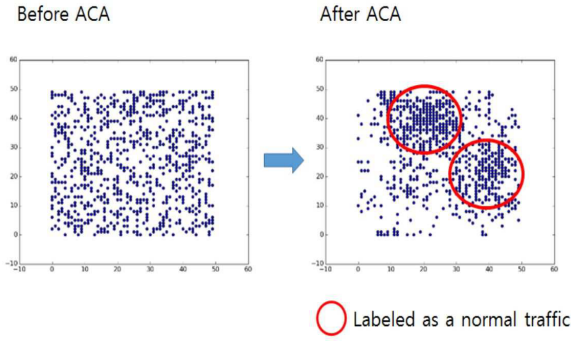

Fig. 2 Clustering result.

Table 3 Our test dataset.

Type	# of Packets	Proportion (%)
Normal	19,268	98.00
Probe	98	0.50
DoS	277	1.41
U2R	17	0.09
R2L	1	0.00
Total	19,661	100

5.4 Experiment Result

This section shows our experimental results. In order to get the best classification phase result, we conduct four different experiments: varying MF positions, MF types, inference rules, and defuzzifier methods.

First, we did experiment with three different MF inputs. Fig. 6 shows the three different inputs. Table 5 shows that MF input (a, b) is the best choice among three different MF

Table 4 Fuzzy rules.

Nearest to Abnormal	Nearest to Normal				
	VeryClose	Close	Average	Far	VeryFar
Close	HighProne	HighProne	Abnormal	Abnormal	Abnormal
Average	LowProne	LowProne	HighProne	HighProne	HighProne
Far	Normal	Normal	Normal	HighProne	HighProne

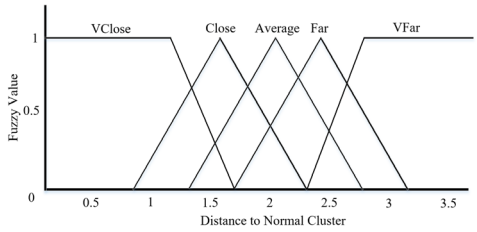

Fig. 3 Membership function for nearest to normal input.

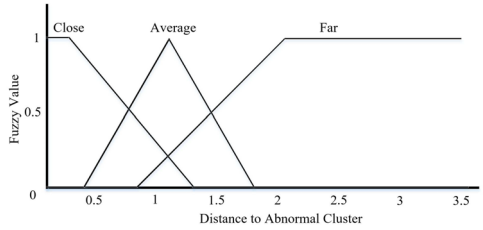

Fig. 4 Membership function for nearest to abnormal input.

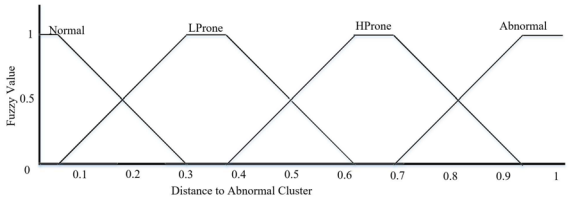

Fig. 5 Membership function for alarm output,

Table 5 Performance with different MF inputs.

Method	FPR (%)	DR (%)
a,b	10.03	92.11
c,d	66.74	95.67
e,f	2.55	0.00

Table 6 Performance with different MF types.

Method	FPR (%)	DR (%)
Trapmf	10.03	92.11
Trimmf	9.37	91.86
Gauss2mf	12.96	92.11
Gbellmf	10.18	93.38

inputs.

Second, we use 4 different MF types: trapmf, trimmf, gauss2mf, and gbellmf. Trapmf represents trapezoidal distribution function. Trimmf represents triangle distribution function. Meanwhile, both gauss2mf and gbellmf represents Gaussian distribution function with different parameters. The selection of those four functions are based on Karami et al. [5] experiments. We cannot compare directly with Karami et al. [5] as the dataset is different, and thus the metrics are totally different which forms the core of the experimentation. In Karami et al. [5] paper, they use Content Centric Network (CCN) dataset which has totally different type with KDD dataset. However, we can compare our works with Karami et al. [5] in term of fuzzy parameter usage. According to Table 6, trapezoidal and triangle distributions gave the best result while in Karami et al. [5], trapezoidal and Gaussian distributions outperformed other distributions. Table 6 shows the effect of different MF types to FPR and DR. Trapezoidal distribution function is the best function among four functions.

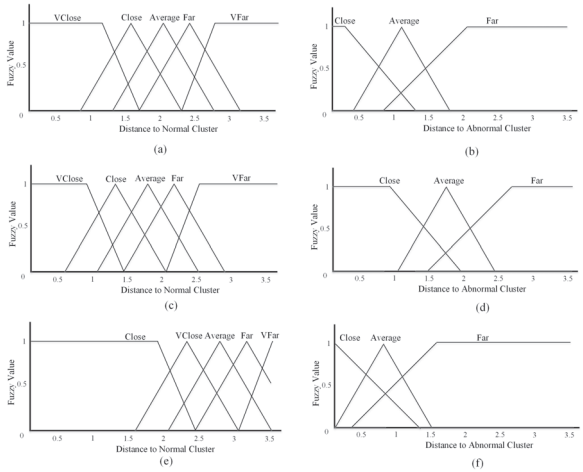

Fig. 6 Three different membership functions. (a)(c)(e) represent different nearest to normal. Meanwhile, (b)(d)(f) represent different nearest to abnormal.

Third, we accommodate different inference rules. The first inference rule shown in Table 4. There are two rules which are not following intuition in Table 4, when nearest to normal far and very far to nearest to abnormal far. Both of them are supposed to be low prone as shown in Table 7. Table 8 shows the effect of different inference rules. Unfortunately, the second rule shown by Table 7 results pretty low DR. The result is make sense since once an instance located far or very far from benign instances, the instance has high probability to be an attack.

Last, we occupy different defuzzifier methods. There are five different defuzzifer methods: Center of Gravity (CoG), bisector, Mean of Max (MoM), Largest of Max (LoM), and Smallest of Max (SoM). Table 9 shows that CoG is the best defuzzifier method in this case.

Recall in the defuzzification step, the results of fuzzy

Table 7 Another fuzzy rules.

Nearest to Abnormal	Nearest to Normal				
	VeryClose	Close	Average	Far	VeryFar
Close	HighProne	HighProne	Abnormal	Abnormal	Abnormal
Average	LowProne	LowProne	HighProne	HighProne	HighProne
Far	Normal	Normal	Normal	LowProne	LowProne

Table 8 Performance with different inference rules.

Method	FPR (%)	DR (%)
Table 4	10.03	92.11
Table 7	66.74	29.26

Table 9 Performance with different defuzzifier methods.

Method	FPR (%)	DR (%)
CoG	10.03	92.11
Bisector	15.4	92.37
MoM	55.08	94.91
LoM	0.00	0.00
SoM	0.00	0.00

Table 10 Some fuzzy rules in proposed system.

IF Normal=*Average* and Abnormal=*Far* **THEN** Alarm=*Normal*
IF Normal=*Close* and Abnormal=*Average* **THEN** Alarm=*LowProne*
IF Normal=*Far* and Abnormal=*Average* **THEN** Alarm=*HighProne*
IF Normal=*VeryFar* and Abnormal=*Close* **THEN** Alarm=*Abnormal*

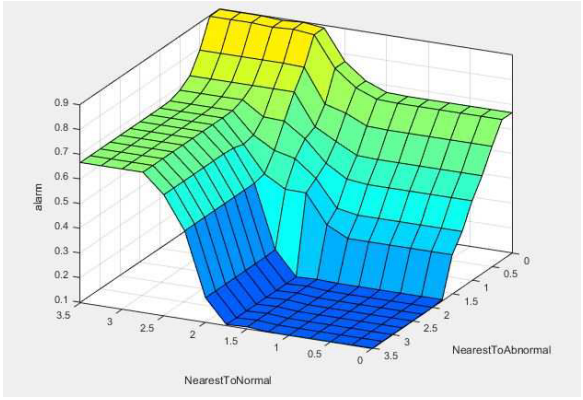

Fig. 7 Fuzzy inference rule in 3D form.

inference are mapped to a crisp (non-fuzzy) output using the output membership functions. If the crisp output is bigger than a predefined threshold (from now on called fuzzy threshold), a test data instance is considered as an abnormal instance, otherwise it is a normal instance. Table 11 shows the performance of our approach using different fuzzy thresholds. We can see that the bigger the fuzzy threshold, the lower the DR. Unfortunately, we also have bigger FPR as a trade-off. We note that 0.65 as fuzzy threshold provide best performance among others with DR = 92.11% and FPR = 10.03%. It means that there are 1,936 legitimate instances detected as an attack. Also, 31 out of 393 attack data instances aren't detected as attacks. Thus, we conclude that 0.65 is the optimal value for the fuzzy threshold.

In order to provide the proper measurement, we compare our scheme with other similar schemes as mentioned by

Table 11 Performance of our proposed scheme.

Fuzzy Threshold	FPR (%)	DR (%)	FNR (%)
0.70	9.40	0.00	100.00
0.65	10.03	92.11	7.89
0.60	20.81	94.91	5.09
0.55	32.35	94.91	5.09
0.30	97.25	98.73	1.27

Table 12 Results comparison.

Method	DR (%)	FPR (%)
AIS+K-means [6]	43.1	15.6
AIS+DBSCAN [6]	58.9	0.8
Our Proposed Scheme	92.11	10.03

Hosseinpour et al. [6]. They proposed a hybrid scheme of AIS and DBSCAN. Similar to our approach, their approach exploits two phases: clustering and detection phase. In addition, they also provide the performance result of another IDS scheme based on AIS and K-means clustering. Table 12 shows the comparison of three different schemes. ACA is a proper algorithm for high density and high dimensional data. Also, ACA is insensitive to initialization step. These properties satisfy the needs of real traffic network, which has high density and high dimensional data. Although ACA needs many input parameters, by combining it with FIS, our proposed scheme is able to achieve significantly higher DR compared to other two schemes. However, our proposed scheme provides quite high FPR. We can vary the parameters and cut it down a little bit. But, if we want the DR to be high, with our scheme the FPR remains a bottleneck. However, we give a comparison with AIS+K-means and AIS+DBSCAN [6] on Table 12. Our scheme is efficiently detecting both known and unknown attacks. This remains a tradeoff whether using less FPR or higher DR is of greater value to the user.

6. Conclusion and Future Work

We propose a novel fuzzy anomaly detection system based on Ant Clustering Algorithm (ACA) and Fuzzy Inference System (FIS). The system contains two phases: the training phase implementing ACA to cluster training dataset; and the classification phase incorporating the FIS. We define our FIS with two distance values as nearest to normal and abnormal clusters. Experimental results show that our scheme is very effective to detect both known and unknown attacks. However, our scheme still provides high FPR. Thus, we will further investigate this issue in the near future.

Acknowledgement

This work was partly supported by IITP funded by the Korea

government (MSIP) (B0101-16-1270) and the NRF funded by the MSIP (No. NRF-2015R1A2A2A01006812).

References

[1] C. Kolias, G. Kambourakis, and M. Maragoudakis, "Swarm intelligence in intrusion detection: A survey," Comput. Secur., vol.30, no.8, pp.625–642, 2011.
[2] P. Laskov, P. Düssel, C. Schäfer, and K. Rieck, "Learning intrusion detection: Supervised or unsupervised?," Image Analysis and Processing, ICIAP 2005, Lecture Notes in Computer Science, vol.3617, pp.50–57, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.
[3] S. Zanero and S.M. Savaresi, "Unsupervised learning techniques for an intrusion detection system," Proc. 2004 ACM Symposium on Applied Computing, SAC'04, pp.412–419, 2004.
[4] C.-H. Tsang and S. Kwong, "Ant colony clustering and feature extraction for anomaly intrusion detection," Swarm Intelligence in Data Mining, Studies in Computational Intelligence, vol.34, pp.101–123, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
[5] A. Karami and M. Guerrero-Zapata, "A fuzzy anomaly detection system based on hybrid PSO-Kmeans algorithm in content-centric networks," Neurocomputing, vol.149, pp.1253–1269, 2015.
[6] F. Hosseinpour, V.P. Amoli, F. Frahnakian, J. Plosila, and T. Hämäläinen, "Artificial immune system based intrusion detection: Innate immunity using an unsupervised learning approach," International Journal of Digital Content Technology and its Applications, vol.8, no.5, pp.1–12, 2014.
[7] P. Albuquerque and A. Dupuis, "A parallel cellular ant colony algorithm for clustering and sorting," Cellular Automata, Lecture Notes in Computer Science, vol.2493, pp.220–230, Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.
[8] N. Monmarché, M. Slimane, and G. Venturini, "Antclass: Discovery of clusters in numeric data by a hybridization of an ant colony with the k-means algorithm," Internal Report, no.213, pp.1–21, 1999.
[9] T. Mikami and M. Wada, "Data visualization method for growing self-organizing networks with ant clustering algorithm," Advances in Artificial Life, Lecture Notes in Computer Science, vol.2159, pp.623–626, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.
[10] P.M. Kanade and L.O. Hall, "Fuzzy ants as a clustering concept," 22nd International Conference of the North American Fuzzy Information Processing Society, NAFIPS 2003, pp.227–232, 2003.
[11] M.S. Abadeh and J. Habibi, "A hybridization of evolutionary fuzzy systems and ant colony optimization for intrusion detection," ISC Int'l Journal of Information Security, vol.2, pp.33–46, 2010.
[12] K. Scarfone and P. Mell, "Guide to intrusion detection and prevention systems (IDPS)," NIST, Special Publication 800-94, 2007.
[13] A.L. Vizine, L.N. de Castro, and E. Hrusch, "Towards improving clustering ants: An adaptive ant clustering algorithm," J. Informatica, vol.29, no.2, pp.143–154, 2005.
[14] H. Izakian and W. Pedrycz, "Agreement-based fuzzy C-means for clustering data with blocks of features," Neurocomputing, vol.127, pp.266–280, 2014.
[15] M. Tavallaee, E. Bagheri, W. Lu, and A.A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," Proc. IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp.1–6, 2009.
[16] H.M. Shirazi, "An intelligent intrusion detection system using genetic algorithms and features selection," Majlesi Journal of Electrical Engineering, vol.4, no.1, 2010.
[17] K. Kim, H. Kim, and K. Kim, "Design of an intrusion detection system for unknown-attacks based on bio-inspired algorithms," Proc. Computer Security Symposium 2015 (CSS 2015), pp.64–70, 2015.
[18] D. Pelleg and A. Moore, "Accelerating exact k-means algorithms with geometric reasoning" Proc. 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'99, pp.277–281, 1999.
[19] J.P. Verma, "Cluster analysis: For segmenting the population," Data Analysis in Management with SPSS Software, pp.317–354, 2012.

Muhamad Erza Aminanto received the B.S. and M.S. degrees in Electrical Engineering from Bandung Institute of Technology (ITB), Indonesia in 2013 and 2014, respectively. He is currently a Ph.D. candidate at School of Computing from Korea Advanced Institute of Science and Technology (KAIST), Korea. His research interests include network security, machine learning application on information security and applied cryptography.

HakJu Kim received the B.S. and M.S. degrees in Computer Science from Korea Advanced Institute of Science and Technology (KAIST) in 2013 and 2015, respectively. He now works for TmaxData company, Korea.

Kyung-Min Kim received the B.S. and M.S. degrees in Computer Science from Sogang University in 2014 and Korea Advanced Institute of Science and Technology (KAIST) in 2016, respectively. He now works for National Security Research (NSR), Korea.

Kwangjo Kim received the B.Sc. and M.Sc. degrees in Electronic Engineering from Yonsei University, Seoul, Korea, in 1980 and 1983, respectively, and the Ph.D. degree from the Division of Electrical and Computer Engineering, Yokohama National University, Japan, in 1991. He was a Visiting Professor with MIT and UCSD, USA, in 2005, and the Khalifa University of Science, Technology and Research (KUSTAR), Abu Dhabi, UAE, in 2012, and an Education Specialist with the Bandung Institute of Technology (ITB), Bandung, Indonesia, in 2013. He is currently a Full Professor with Graduate School of Information Security, School of Computing, KAIST, Korea, the Korean representative to IFIP TC-11, and the honorable President of the Korea Institute of Information Security and Cryptography (KIISC). His current research interests include the theory of cryptology and information security and their practices. Prof. Kim had served as a Board Member of the IACR from 2000 to 2004, the Chairperson of Asiacrypt Steering Committee from 2005 to 2008, and the President of KIISC in 2009. He is a member of IEEE, ACM, IACR and IEICE.