Incremental Hybrid Intrusion Detection Using Ensemble of weak classifiers

Amin Rasoulifard, Abbas Ghaemi Bafghi, Mohsen kahani
Department of Computer Science and Engineering, Faculty of Engineering
Ferdowsi University of Mashhad, Mashhad, Iran
E-mail: rasoulikarbar@yahoo.com, {Ghaemib, kahani}@um.ac.ir

***Abstract*— In this paper, an incremental hybrid intrusion detection system is introduced. This system combines incremental misuse detection and incremental anomaly detection. It can learn new classes of intrusions that are not exist in the training dataset for incremental misuse detection. As the framework has low computational complexity, it is suitable for real-time or on-line learning. Also experimental evaluation on KDD Cup dataset are presented**

KeyWords: incremental learning, ensemble of weak classifiers, hybrid, Learn++

I. INTRODUCTION

With the fast growing of network-based services and sensitive information on the networks, the number and the severity of network-based computer attacks have significantly increased. Although a wide range of security technologies such as information encryption, access control, and intrusion prevention can protect network-based systems, there are still many undetected intrusions.

An intrusion can be defined as "any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource". An IDS can detect and identify intrusion behavior or intrusion attempts in a computer system by monitoring and analyzing network packets or system audit logs, and then sends intrusion alerts to system administrators in real time. Intrusion detection techniques can be categorized into misuse detection and anomaly detection [1].

Misuse detection systems use patterns of well-known attacks or weak spots of the system to identify intrusions. The main shortcoming of such systems[2,3,4] are the necessity of hand-coding of known intrusion patterns and their inability to detect any future(unknown) intrusions not matched with the patterns stored in the system.

Anomaly detection systems, on the other hand, firstly establish normal user behavior patterns (profiles) and then try to determine whether deviations from the established normal profiles can be flagged as intrusions. The main advantage of anomaly detection systems is that they can detect new types of unknown intrusions [5,6,7].

In recent years, the continual emergence of new attacking methods has caused great loss to the whole society. So, the advantage of detecting future attacks has specially led to an increasing interest in incremental learning techniques. The traditional methods commonly build a static intrusion detection model on the prior training dataset, and then utilize this model to predict on new network behavior data. However, the network behavior model does not change continually along with detecting and analyzing process. Thus the initially learnt intrusion detection model can not adapt to the new network behavior pattern, which causes an increase in the false positive rate and decreases the detection precision of the system

In order to improve intrusion detection with high detection rate, with the ability of detection new unknown attacks, and continually adapt model to cope with new network behaviors, we propose a hybrid intrusion detection system which combines the incremental misuse intrusion detection and incremental anomaly detection. In addition, when intrusion detection dataset is so large that whole dataset can't be loaded into the main memory, the original dataset can be partitioned into several subsets, and then the detection model is dynamically modified according to other training subsets after the detection model built on one subset.

Weak classifiers are those that obtain 50 percent classification accuracy on it own training data [16]. *Ensembles* are combinations of several models whose individual predictions are combined in some manner (e.g., averaging or voting) to form a final prediction [12].

Several hybrid intrusion detection systems have been proposed for combining misuse detection and anomaly detection [8,9,10]. We proposed hybrid intrusion detection system based on incremental learning. We use ensemble of weak classifiers for implementing incremental misuse intrusion detection system. Intrusion detection systems using ensemble of weak classifiers generally possesses lower computational complexity than other frameworks which that use strong classifier, because of using weak classifier with lower computational complexity. We use on-line k-mean algorithm for incremental anomaly detection to detect unknown intrusions.

The rest of the paper is organized as follows: related work presented in section II, hybrid system architecture presented in section III, the proposed architecture presented in section IV, KDD Cup Dataset presented in Section V, experimental evaluation presented in section V, comparison to other algorithms presented in section VII computational complexity presented in section VIII and finally we conclude the paper in the conclusion section.

II. RELATED WORK

Hybrid intrusion detection systems are composed of misuse detection and anomaly detection system. They can detect both known intrusions and unknown intrusions. Various methods have been proposed that address the problem of misuse intrusion detection and anomaly detection systems.

ADAM (Audit data analysis and mining) is a hybrid on-line intrusion detection system which uses association rules for detecting intrusions [8]. This framework consists of two phases: training phase and on-line phase. In training phase, the dataset without any class of intrusions is applied to the model and constitute a profile of normal activities as a set of association rules pattern. In on-line phase, ADAM uses sliding window, on-line algorithm that find frequent pattern in the last D connections and compare them with those stored in the normal profile, and discard those that are similar to the pattern of the normal profile. With the rest, ADAM uses a classifier which has been previously trained to classify the suspicious data as a known type of attack, unknown types and a false alarm.

The Next Generate Intrusion Expert System (NIDES) is a hybrid system [9]. It is consists of rule-based misuse detection and anomaly detection that use statistical approaches. This framework employs misuse detection and anomaly detection in parallel for detecting intrusions.

The random forest algorithm used for hybrid intrusion detection system in [10]. It uses ensemble of classification tree for misuse detection and use proximities to find anomaly intrusions. Similar to ADAM it has two phases: on line phase, off-line phase. In the on-line phase the classification trees are used to generate the pattern of known intrusions and in the off-line part of the algorithm, system can detect unknown intrusions and build patterns of known intrusions and add them to the database of known intrusion patterns.

FLIPS is the framework which uses hybrid approach for intrusion prevention systems [11]. The core of this framework is an anomaly-based classifier that incorporates feedback form environment to both tune its model and automatically generate signatures of known malicious activities. It uses PayL as an anomaly detection component. The misuse detection component of this framework is signature-based intrusion detection system which use pattern of intrusions for detection intrusions.

In the proposed incremental hybrid system, we use Learn++ algorithm for the incremental misuse detection component and on-line k-mean algorithm is used for the incremental anomaly detection component.

III. ARCHITECTURE OF HYBRID SYSTEM

To improve the productivity of misuse and anomaly detection, several hybrid intrusion detection systems have been proposed. These frameworks combine misuse and anomaly detection to achieve the detection rate of misuse detection and to detect unknown intrusions. There are three ways to combine misuse and anomaly detection: use anomaly detection at first then misuse detection, use misuse and anomaly detection in parallel and use misuse detection and then anomaly detection afterwards.

Some hybrid intrusion detection systems use anomaly detection at first to detect suspicious activities and then use misuse detection to detect attacks from suspicious activities [8,11]. Suspicious activities are those that deviate from the profile of normal activities. The framework of this approach is shown in Fig.1. The observed activities applied to the anomaly detection to produce suspicious activities and then misuse detection is used to detect attacks. Connections that match to the pattern of attacks are labeled as attack, those that match to false alarm patterns are labeled as normal and the others are labeled as unknown attacks.

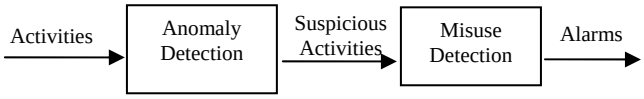

Fig.1: Anomaly detection at first then misuse detection

In order to reduce false positive rate of anomaly-first hybrid frameworks, the anomaly detection component must have high detection rate and the misuse detection component must have high false positive rate. So these frameworks are not suitable for hybrid intrusion detection systems.

Some hybrid intrusion detection systems use misuse detection and anomaly detection component in parallel [9]. In these frameworks both components generate suspicious activities separately. Then, a correlation component is used to elicit intrusions from suspicious activities. The framework of these hybrid systems is shown in Fig.2.

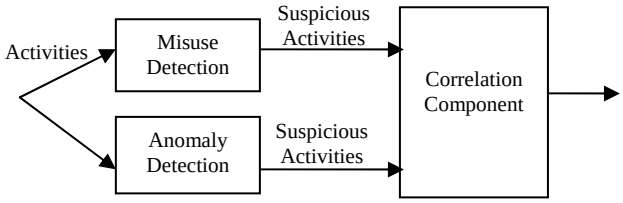

Fig.2: Parallel approach

Other hybrid intrusion detection systems employ misuse detection at first and use anomaly detection afterwards [10]. These frameworks can detect known attacks in real time and generate suspicious activities from the observed activities. Anomaly detection is used to detect unknown intrusions from the suspicious activities. The framework of these hybrid systems is shown in Fig.3.

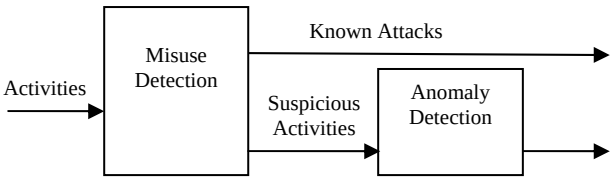

Fig.3: Misuse detection at first then anomaly detection

Our proposed incremental hybrid intrusion detection system uses this type of hybrid intrusion detection systems. This is suitable for detecting known intrusions in real time because of using weak classifiers with lower complexity and has the ability to learn new intrusions incrementally.

IV. The Proposed System Architecture

It is important to increase the detection rate for known intrusions and detect unknown intrusions. It is also important to incrementally learn new intrusions. Due to the fast growing of new intrusions in recent years, Detecting and learning future intrusions will be the main interests in intrusion detection systems. We propose incremental hybrid intrusion detection system which use ensemble of weak classifiers, for incrementally learning new

Intrusion detection systems using ensemble of weak classifiers generally possesses lower computational complexity than other frameworks that use strong classifiers, because of using weak classifiers with suitable parameter to satisfy weak hypotheses. This property is very attractive and promising in intrusion detection systems, because the classifiers should be retained in the short periods in practice.

A. Ensemble of weak classifiers

Ensemble of classifiers developed to improve the classification performance of weak classifiers. In essence, an *ensemble of weak classifiers* are trained using different distributions of training samples, whose outputs are then combined using one method for combining classifiers [12] to obtain final classification rule. This approach exploits the so-called *instability* of the weak classifiers, which allows the classifiers to construct sufficiently different decision boundaries for minor modifications in their training datasets, causing each classifier to make different errors on any given instance. A strategic combination of these classifiers then eliminates individual errors, generating a strong classifier.

B. Proposed hybrid architecture

Misuse detection has high detection rate for known intrusions and cannot detect unknown intrusions. Anomaly detection can detect unknown intrusions but having a low detection rate and high computational complexity. It is also important to incrementally learn new intrusions. In order to obtain intrusion detection with aforementioned techniques, we propose an incremental hybrid system which combines the incremental misuse intrusion detection and incremental anomaly detection. The framework for proposed hybrid intrusion detection system is shown in Fig.4.

The proposed framework divided into two phases: on-line phase and off-line phase. Misuse intrusion detection component is used in the on-line phase. It can learn new class of intrusions that not exist in previous data which used for training the existing classifiers. In other words, it can learn new class of intrusions in supervised mode. It is also suitable for learning known intrusions in on-line mode because of using ensemble of weak classifiers with lower computational complexity.

In the off-line section of our framework, we use on-line k-mean algorithm. It can identify new unknown intrusions and can incrementally learn new instance of data. The new identified intrusions by anomaly detection component must be applied to the misuse intrusion detection component in the next learning phase. Therefore, we must determine the class type of new intrusions. For this reason, another component is used for determining the class type of new intrusions. This component is optional and can be done by the administrator of systems. Any supervised or unsupervised clustering algorithms can be used for this component. In our experiment, we manually determine the class type of intrusions that detected by anomaly detection component.

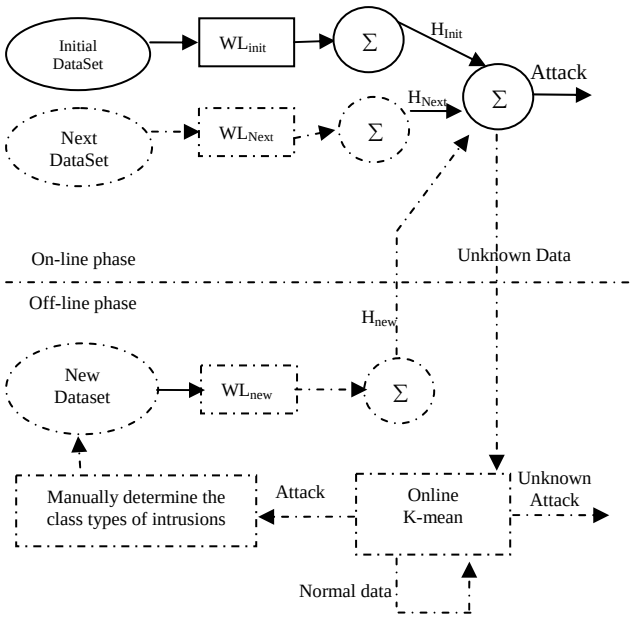

Fig.4: incremental hybrid intrusion detection system

C. Misuse intrusion detection

Misuse detection systems use patterns of well-known attacks or weak spots of the system to identify intrusions. The main shortcomings of such systems are that it cannot detect new unknown intrusions.

The fast growing of new intrusions in computer systems led to an increasing interest in incremental learning algorithms for intrusion detection systems. Learn++ algorithm is an incremental learning algorithm that used ensemble of weak classifiers for incrementally learn new information [13].

We use Learn++ algorithm for incremental misuse detection component of proposed hybrid intrusion detection system. It has an ability to incrementally learn new class of intrusions that not trained as output for existing classifiers. In other words, this algorithm can learn new class of intrusions in supervised mode. Intrusion detection systems using ensemble of weak classifiers generally possesses lower computational complexity than the other frameworks which using strong classifiers, because of using weak classifiers lower computational complexity. The framework of

incremental misuse intrusion detection system is shown in Fig.5 which has the following components:

WL: Learn++ algorithm requires a base classifier for generating a group of **weak Learner** (classifiers) designed before hand. Weak Classifier can obtain a 50% correct classification performance on its own training dataset. We use multi layer perceptron for implementing a weak classifier.

$\sum$: **Weighted Majority voting** which used for calculating the final classification accuracy based on the classification accuracy of the weak classifiers.

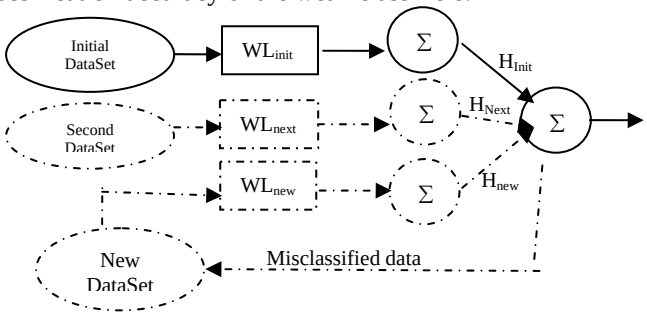

Fig 5: incremental misuse intrusion detection

D. Anomaly detection

Anomaly detection amounts to training models for normal behavior and then classifying as intrusions any network behavior that significantly deviates from the known normal patterns. Clustering algorithms have recently gained attention in intrusion detection systems because of having advantages to find new attacks not seen before. With the fast growing of new attacks in recent years, incremental learning gained interest attention for detecting future intrusions. We use incremental clustering algorithm in proposed hybrid system for incrementally learn new unseen intrusions. We use on-line k-mean algorithm [15]. It has a low time complexity, fast convergence and it is suitable for incremental learning. The pseudo-code for on-line k-mean algorithms is shown in Fig.6.

Algorithm: online k-means (kmo)

Input: A set of N data vectors $X = \{x_1,...,x_N\}$ in $\Re^d$ and number of clusters K.

Output: A partition of the data vectors given by the cluster identity vector

$$Y = \{y_1,...,y_N\}, y_n \in \{1,...,k\}$$

Steps:

1. Initialization: initialize the cluster centroid Vectors $\{\mu_1,...,\mu_K\}$;
2. Loop for M iterations
   For each data vector x_n, set

$$y_n = \arg\min_k \left\| x_n - \mu_k \right\|^2,$$

   Update the centroid μ_{y_n} as

$$y_{y_n}^{(new)} = \mu_{y_n} - \frac{\partial E}{\partial_{\mu_{y_n}}} = \mu_{y_n} + \xi(x_n - \mu_{y_n}),$$

Where ξ is a learning rate usually set to be a small positive number (e.g., 0.05). the number can also gradually decrease in the learning process.

Fig.6: on-line k-mean algorithm

V. KDD CUP DATASET

The KDD cup 1999 intrusion detection contest data (KDD cup 99 intrusion detection dataset) is used in our experiments. This data was prepared by the 1998 DARPA Intrusion Detection Evaluation program by MIT Lincoln Labs (MIT Lincoln Laboratory). Lincoln labs acquired nine weeks for raw TCP dump data. The raw data was processed into connection records, which consist of about 5 million connection records. The data set contains 24 attack types. These attacks fall into four main categories:

Denial of Service (DOS): in this type of attack an attacker makes some computing or memory resources too busy or too full to handle legitimate requests, or denies legitimate users access to a machine. Examples are apache2, Back, Land, Mailbomb, SYN Flood, Ping of death, Process table, Smurf, Teardrop.

Remote to User (R2L): in this type of attack an attacker who does not have an account on a remote machine sends packets to that machine over a network and exploits some vulnerability to gain local access as a user of that machine. Examples are Dictionary, Ftp_Write, Guest, Imap, Named, Phf, Sendmail, and Xlock.

User to Root (U2R): In this type of attacks an attacker start out with access to a normal user account on the system and is able to exploit system vulnerabilities to gain root access to the system. Examples are Eject, LoadModule, Ps, Xterm, Perl, and Fdformat.

Probing: In this type of attacks an attacker scans a network of computers to gather information or find known vulnerabilities. An attacker with a map of machines and services that are available on a network can use this information to look for exploits. Examples are Ispsweep, Mscan, Saint, Satan, Nmap.

The data set has 41 attributes for each connection record plus on class label. R2L and U2r attacks don't have any sequential patterns like DOS and Probe because the former attacks have the attacks embedded in the data packets whereas the later attacks have many connections in a short amount of time. Therefore, some features that look for suspicious behavior in the data packets like number of failed logins are constructed and these are called content features.

An original sample consisting of about 4898431 records obtained from the UCI machine learning repository was used in our study as training set and entire labeled test set is used for testing set. The labeled test dataset includes 311029 records with different distribution from the training set. The distribution of normal and attack types of connection records in these subsets have been summarized in Table 1 and Table2 .

Table1. Distribution of Original Training dataset of KDD Cup 99 dataset

Class	Samples	Sample Percent (%)
Normal	972780	19.85
Dos	3883370	79.27
U2R	52	0.001
R2L	1127	0.023
PROBE	41102	0.83
	4898431	100

Table2. Distribution of Testing dataset of KDD Cup 99 dataset.

Class	Samples	Samples Percent(%)
Normal	60593	19.48
DOS	223298	71.79
U2R	39	0.01
R2L	5993	1.92
PROBE	2377	0.76
NOVEL	18729	6.02
Total	311029	100

VI. EXPERIMENTAL EVALUATION

For simulations, the weak learner used to generate individual hypotheses was a single hidden layer MLP with 41 hidden layer nodes and 4 nodes in output layer. The 4 nodes in output layer correspond to the four class type of intrusions. The mean square error goals of all MLPs were preset to values of 0.02 to prevent over-fitting and to ensure sufficiently weak learning. We note that any neural network can be turned into weak learning algorithms by selecting its number of hidden layers and the number of hidden layer nodes small, and the error goal high, with respect to the complexity of problem.

For validating the effectiveness of incremental hybrid intrusion detection using ensemble of weak classifiers, the following experiments are done.

The original training sample obtained from the KDD Cup dataset was used in our study as training set and entire labeled test set is used for testing set. The labeled test set with different distribution from the training set. We use intrusions of training set to generate the model of incremental misuse component and use normal instances of training set to construct the profile of normal activities.

In order to generate the initial model of our framework, the following scenario is done. In order to make the anomaly detection component, we elicit the normal instances from training dataset for constructing the profile of normal activities. The remaining intrusions instances used as intrusion dataset to make the misuse detection component. We select 10% of intrusion dataset consists of about 400000 instances which contains four class types of intrusions as initial dataset. This dataset used to make an initial model of misuse detection component.

After getting the initial model of our framework, we pursue the following scenario for adding additional model to the initial model. The 90% remaining attack dataset apply to the initial model of misuse detection. The initial model has 21% detection rate on this dataset, the other are clarify as unknown instances. The unknown instances applied to the anomaly detection in order to detect unknown attacks. The anomaly detection component has 76% detection rate and 24% false positive rate on the unknown dataset. The number of clusters equal to 10 in our experiment. Then we manually determine the class types of instances that anomaly detection detects them as attack. These new instances which detected as attack by anomaly detection are candidate for preparing the new intrusions dataset. In order to make the next classifier or model, we use a threshold equal to 100000 instances for constructing the next dataset which randomly selected from the new intrusions dataset. So, we select the next dataset from the new intrusions dataset for generating the next model or classifier (H). The ensemble of existing classifiers is used for misuse detection component in the next iterations.

The above scenario will be done in several iterations for generating the next models based on the new available intrusions dataset. The current model which generated from the new dataset and existing classifiers contribute to getting the final classification accuracy in the next iterations. At the end of aforementioned scenario, 7 classifiers generated from the training dataset and the ensemble of them achieves 97% detection rate on the whole training dataset. The total number of instances that our framework used based on aforementioned scenario, to generate 7 classifiers, are approximately 1000000 records and the remaining instances are removed during construction of model at each iteration based on the ensemble accuracy. Improving the detection rate from 21% to 97% on training dataset means that the existing model can learn new information (new classifier) in order to achieve the higher classification accuracy. This idea can be the main interest of ensemble learning to implement incremental learning based on weak classifiers.

After we get the new model of misuse detection component based on the training dataset, we test the model on testing dataset for evaluating the effectiveness of our incremental intrusion detection system. The results of simulation show that when new unknown data become available (new classifier is generated), the classification performance of ensemble approach on testing dataset increased. In other words, the framework can learn new information in the next iterations when becomes available in current iteration.

Fig.7 has shown that adding the additional training sample for generate classifier (H) caused to improving the detection rate of misuse detection on test dataset from 45.3 to 87.8. This demonstrates its incremental learning capability even when instances of new classes are introduced in subsequent training data. So, our hybrid intrusion detection system can learn new information incrementally when it is been introduced.

The effectiveness of anomaly detection component on the performance classification of hybrid intrusion detection system investigated in the following scenario. We remove the instances of data that correctly classified by misuse detection component from the testing dataset and apply anomaly detection component on remaining instances of testing dataset. These remaining instances are those that misuse detection component clarify them as unknown data. Then incremental anomaly detection applied to remaining dataset to detect anomaly intrusions. Many instances of remaining dataset detected as intrusions that misuse detection component could not clarify the classification output of them.

As indicated in Fig.8 there are instances of data in remaining dataset which detected as intrusions by anomaly detection. These instances were not predicted by misuse detection component. This means that combining misuse detection and anomaly detection can detect more intrusions than each of them individually. These intrusions will be

learned in the next iterations. It is the main interest of ensemble learning using weak classifiers for incremental learning and demonstrates the adaptively and effectively of presented incremental hybrid intrusion detection system.

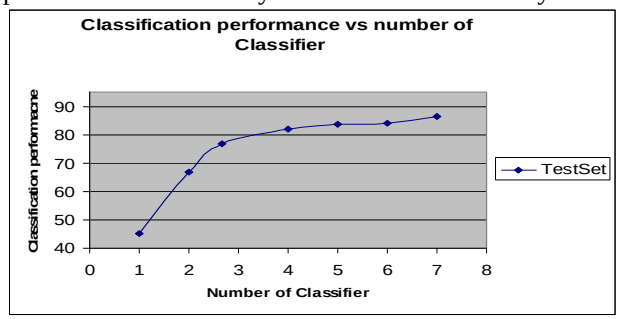

Fig.7: Classification performance vs number of classifier

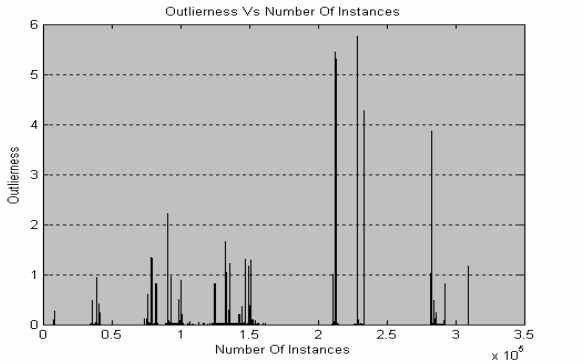

Fig.8: Intrusions detected by anomaly detection

VII. COMPARITION TO OTHER ALGOITHMS

We compare the performance classification of our framework with IEIDSLA [14] framework which running on KDD Cup dataset 99 because the contribution of both papers are based on incremental learnining. We improve the detection rate on test dataset from 45.3% to 87.8% percent while the IEIDSLA improve the performance classification from 77.5% to 84.6%. It can be easily seen that our framework has higher detection rate while it can start with lower detection rate. This means that our framework can start with a small value of data that is available then gradually learn new information when become available.

Table.3 Comparison of Different Algorithms

Framework	Detection Rate(From)	Detection Rate (To)
IEIDSLA	77.5%	84.6%
Ourframwrok	45.3%	87.8%

VIII. COMPUTATIONAL COMPLEXITY

After analysis the LEARN++ algorithm, we calculate that in training phase of our framework, the computational complexity of initial model in the on-line phase is $O(nT_k\alpha)$, where n is the number of instances, T_k is the number of weak hypotheses that must be generated, α is the complexity of BaseClassifier which in our framework is simple multi layer perceptron. For testing phase, computational complexity of our framework is $O(n\alpha')$, where n is the number of test instances, α' is the complexity of BaseClassifier in testing phase. As to ANN, the computational complexity of the training phase depends on the distribution of dataset, and in the worst case it is $O(n^2M^2)$, which is higher than Learn++. M is the number of decision stumps. In a word, Learn++ generally possesses lower computational complexity than strong ANN, especially in training phase.

Clustering algorithms can be divided in two categories [15]: similarity based and centroid based. Similarity algorithms have a complexity at least $O(N^2)$, where N is the number of instances. In contrast centroid-based algorithms have a complexity of $O(NKM)$, where K is the number of clusters, M is the number of batch iteration and N is the number of instances. The on-line k-mean algorithm is a centroid-based which can be a desirable choice for on-line learning. Because it has high clustering quality, relatively lower complexity and fast convergence.

Our framework use simple multi layer perceptron in order to generate weak hypotheses. The complexities of these hypotheses are very lower than the strong classifier that can be constructed with neural network, so the framework has lower complexity than strong classifier.

Any other classification algorithms can be used for generating weak hypotheses. This is my research interest in intrusion detection that we want to work on it in future.

IX. CONCLUSIONS AND FURTURE WORK

In the case that the intrusion detection instances are updated continually and infinitely, the static model learned on the initial training dataset unable to update the profile of model dynamically. For improving the adaptively of the intrusion detection model to network behavior, we present an incremental hybrid intrusion detection model based on ensemble of weak classifiers. The detecting model can incorporate new instances continually, and therefore enhance generalization performance of detecting model.

In this paper, we manually determine the class types of intrusions that detected as attacks by anomaly detection component, in future we want use unsupervised clustering to determine the class types of unknown intrusions automatically.

The research of this paper will have an important significance for building an efficient and applicable intrusion detection system.

REFERENCES

[1] Richard Heady, George Luger, Arthur Maccabe, and Mark Servilla. The architecture of a network level intrusion detection system. Technical Report CS90-20, Department of Computer Science, University of New Mexico, August 1990.

[2] Mounji, A., Charlier, B.L., Zampuniéris, D., & Habra, N. (1995). Distributed audit trail analysis. In D. Balenson & R. Shirey (Eds.), *Proceedings of the ISOC'95 symposium on network and distributed system security* (pp. 102--112), IEEE Computer Society, Los Alamitos, CA.

[3] Lindqvist, U., & Porras, P.A. (1999). Detecting computer and network misuse through the production-based expert system toolset (PBEST). In L. Gong & M. Reiter (Eds.), *Proceedings of the 1999 IEEE symposium on security and privacy* (pp. 146--161), IEEE Computer Socitey, Los Alamitos, CA.

[4] Ilgun, K., Kemmerer, R.A., & Porras, P.A. (1995). State transition analysis: A rule-based intrusion detection approach. *IEEE* Transactions *on Software Engineering*, *21* (3), 181--199.

[5] F. Neri, "Comparing local search with respect to genetic evolution to detect intrusions in computer networks," in Proceedings of the 2000 Congress on Evolutionary Computation, vol. 1, pp. 238–243, Mar-seille, France, July 2000. IEEE, IEEE. Source: IEEE Xplore

[6] J. Guan, D. X. Liu, and B. G. Cui, "An induction learning approach for building intrusion detection models using genetic algorithms," in Proceedings of Fifth World Congress on Intelligent Control and Automation WCICA, vol. 5, pp. 4339–4342. IEEE, June 2004.

[7] C. Kruegel, T. Toth, and E. Kirda, "Service specific anomaly detection for network intrusion detection," in Proceedings of the 2002 ACM symposium on Applied computing, pp. 201–208. ACM, Symposium on Applied Computing, ACM Press New York, NY, USA, Mar. 2002.

[8] Daniel Barbarra, Julia Couto, Sushil Jajodia, Leonard Popyack, and Ningning Wu, "ADAM: Detecting Intrusion by Data Mining", *Proceedings of the 2001 IEEE, Workshop on Information Assurance and Security T1A3 1100 United States Military Academy*, West Point, NY, June 2001.

[9] Debra Anderson, Thane Frivold, And Alfonso Valdes, *Next-Generation Intrusion Detection Expert System (NIDES)-A Summary*, Technical Report SRICLS-95-07, SRI, May 1995.

[10] J. Zhang and M. Zulkernine, "A Hybrid Network Intrusion Detection Technique Using Random Forests," Proc. of the International Conference on Availability, Reliability and Security (AReS), IEEE CS Press, pp. 262-269, Vienna, Austria, April 2006.

[11] M. Locasto, K. Wang, A. Keromytis, and S. Stolfo. Flips: Hybrid adaptive intrusion prevention. In Proceedings of the 8th International Symposium on Recent Advances in Intrusion Detection (RAID), September 2005.

[12] Lei Xu, Adam Krzyzak, Ching Y, Suen, Methods of Combining Multiple Classifier and Their Application to Handwriting Recognition, IEEE TRANSACTION ON SYSTEMS, MAN AND CYBERNETICS, VOL. 22, NO. 3, MAY/JUNE 1992.

[13] Polikar R., Udpa L., Udpa, S., Honavar, V., "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Transactions on System, Man and Cybernetics (C), Special Issue on Knowledge Management*, vol. 31, no. 4, pp. 497-508, 2001.

[14] Wu Yang, Xiao-Chun Yun, Le-Jun Zhang, Using Incremental Learnining Method From Adaptive Network Intrusion Detection, Proceedings of the Fourth International Conference on Machine Learnining and Cybernetics, Guanbzhou, 18-21 August 2005

[15] Shi Zhong, Taghi Khoshgoftaar, and Naeem Seliya Clustering-Based Network Intrusion Detection, International Journal of Reliability, Quality and Safety Engineering.

[16] Y. Freund and R. Schapire, "A decision theoretic generalization of on-line learning and an application to boosting," *Comput. Syst. Sci.*, vol. 57, no. 1, pp. 119–139, 1997.