International Journal of Emerging Trend in Engineering and Basic Sciences (IJEEBS)
ISSN (Online) 2349-6967
Volume 2 , Issue 1(Jan-Feb 2015), PP397-403

A Survey of Intrusion Detection in Dynamic Distributed Network Using Gaussian Mixture Models along with Particle Swarm Optimization and Support Vector Machines

Amol S. Jadhav[1], Prof. Dhanashree Kulkarni [2]
[12]*(computer department, DYPCOE/ , Savitribai Phule Pune University, India)*

Abstract :- Network security becomes more complex due to the arrival of large no. of new type of attacks and lack of dynamic system to detect new type of attacks. In this paper we survey the solution to frequently changing network environment. We define Online Adaboost-based parameterized method. It contain two model, Local model and Global model. In the local model, online Gaussian mixture models (GMMs) and online Adaboost process are used as weak classifiers. A global detection model is constructed by combining the local parametric model. This combination is achieved by using an algorithm based on support vector machines (SVM) and particle swarm optimization (PSO). Experimental results show that the by using online Adaboost process with Local model and Global model increases the detection rate and reduce the false alarm rate.

Keywords: Adaboost, detection rate, false alarm rate, network intrusions, parameterized model.

I. INTRODUCTION

Intrusion detection in network is a crucial part of information security. Now a day's Firewall, Network based intrusion detection system (NIDS) like devices are used to detect and prevent attacks in network infrastructure. Firewall is used to block certain type of attack, but firewall doesn't detect the intrusion within the system. NIDS detect the only known attacks, it fail to detect the new type of attacks. Deep packet inspection (DPI) [1] can incorporate NIDS into firewalls. It can increase the accuracy of intrusion detection system, but it require more time.

This paper focuses on investigation of NIDS. NIDS uses the misuse-based methods that utilize the signature of attack to detect the intrusions in the network. Like this there are lots of techniques to detect the intrusion but they are effectively for well-known attacks only, they are fail to detect the new type of attacks. Currently in network daily new attacks are introduced so the existing systems are unable to prevent or detect the this new type of attacks. In the following we review the work focusing on the online Adaboost-based algorithms related. Machine learning model deals with detection of unknown attacks using the network feature[29]. This work focus on machine learning-based NIDS. The machine learning-based intrusion detection methods can be classified into three classes as follow.

1.1) Statistics-based: It construct statistical models of network connections to determine whether a new connection is an attack. For instance, Denning [2] construct statistical profiles for normal behaviours.
1.2) Data mining-based methods mine rules that are used to determine whether a new connection is an attack. For instance, Lee et al. [3] characterize normal network behaviours using association rules and frequent episode rules [4].
1.3) Classification -based methods constructs a classifier that is used to detect new connection is attack or normal connection. For instance, Mukkamala et al. [5] use the SVM to classify attack or normal connection.

Although there is a research required in the Distributed intrusion detection system (DIDS), especially in the following areas:

1) Network infrastructure and the intrusion training data changed day to day, every day new type of attacks are entered into network infrastructure, due to that size of training data increased over a time and it becomes a very large. Now a previously existing algorithms are almost offline. So it is necessary to use online training which is suitable for dynamic intrusion detectors.

2) In traditional network intrusion detection system, centralized system was used, so due to that lot's of burden occur in central site. Distributed detection system [6], which use local model to shares intrusion detection models learned in local nodes, which reduce the central site load and keep the data privacy. Otey et al. [7] construct a novel distributed algorithm for detecting outliers (including network intrusions). Its limitation is that many raw network data still need to be shared among distributed nodes. There is a requirement for a distributed intrusion detection algorithm to make only a small number of communications between local nodes.

So authors work on this problem and present the solution. We present these publication as a dynamic online solution to the new type of attacks in network. To provide a comprehensive review of how online Adaboost parameterized methods are applied for intrusion detection in dynamic distributed network. This work examines how well a online Adaboost-based algorithms are effective for new type of attacks and how they are used.The rest of the paper is organized as follows: Section II introduce the overview of framework. Section III describe the local detection model. Section IV presents the method for constructing the global detection models. Section V shows the experimental results. Section VI summarizes the paper.

II. DISTRIBUTED INTRUSION DETECTION FRAMEWORK

There have been many survey of the field Dynamic DIDS. In particular Weiming Hu et al. [8] provide a comprehensive review of the online Adaboost-Based parameterized methods for Dynamic distributed network Intrusion detection which contain two models; Local Model and Global Model. Fig.2 gives an overview of framework that consists of the local models, and global models.

2.1. Local Models: Local model is constructed into each node by using weak classifiers and Adaboost-based training. So that each node contains a parametric model that consists of the parameters of the weak classifiers and the ensemble weights.

2.2. Global Models: It is constructed by combining all local parametric models by using PSO and SVM based algorithms. Global models are used to update local models and then updated models are shared by other nodes.

III. ONLINE ADABOOST-BASED LOCAL INTRUSION DETECTION MODELS

The classical Adaboost algorithm [9] carries out the training task in batch mode. By using training set a number of weak classifiers are generated. The final strong classifier is an ensemble of weak classifiers.

Weak Classifiers: Weak classifier consist two types.

3.1. Decision stumps and normal behaviors for classifying attacks.

The limitation of weak classifier is that the decision stumps do not consider the different types of attacks. This cause the influence in the performance of the Ad boost method.

3.2. Online GMMs that model a distribution of values of each factor component for each attack type.Online GMM: For each type of attack or normal samples, we use a GMM. Let s ε {+1, -1,-2,....., -N} be a sample label where +1 represents normal samples and 1,-2,....., -N represent different types of attacks where N is number of different type of attacks, s represent the j^{th} element of sample. The GMM model θ^c_j on the j^{th} feature component for the samples c is:

$$\theta^c_j = \{w^c_j(i), u^c_j(i), \sigma^c_j(i)\}^k_{i=1} \qquad (1)$$

Where = number of GMM components indexed by i, w=weight, μ= mean, and σ= standard deviation. Where the computational complexity of the online GMM for one sample is O(k), which is higher than the decision stumps. Design of the weak classifiers and the strong classifier, as shown in Fig. 1.

Fig. 1. Framework of our algorithm.

The difference between offline Adaboost algorithm and online Adaboost algorithm are as follows.

i) Offline Adaboost algorithms are constructed in one step while online Adaboost algorithm are updated one by one.
ii) In the offline Adaboost algorithm, the sample weights are updated simultaneously. In the online Adaboost algorithm, the sample weights are updated one by one.

iii) In the offline Adaboost algorithm, the number of weak classifiers are not fixed while in online Adaboost algorithm, the number of weak classifiers is fixed, and equal to the dimension of the feature vectors.
iv) Offline Adaboost algorithm is less accurate than the online Adaboost algorithm.
New online Adaboost algorithm overcomes the limitation of traditional online Adaboost algorithm.The performance of algorithm is calculated by using detection rate and false alarm rate. And it depends on the initial weight of the training samples.

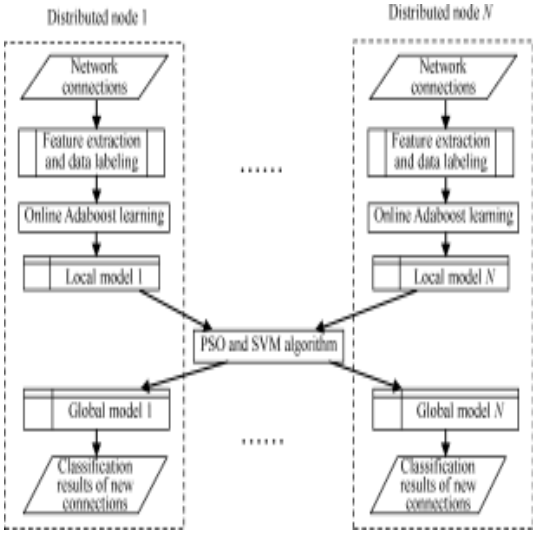

Fig. 2. Overview of the intrusion detection framework

Let t be initial weight of each training sample

$$t=\begin{cases}\dfrac{(M_{normal}+M_{intrusion})*r}{M_{normal}} & \text{for normal connections}\\[2ex] \dfrac{(M_{normal}+M_{intrusion})*(1-r)}{M_{intrusion}} & \text{for network intrusion}\end{cases} \quad (2)$$

where M_{normal} is a number of normal sample, $M_{intrusion}$ is a number of attack sample and r Ɛ (0,1). The value of r depends on the proportion of the normal samples, detection rate and the false alarm rate in specific applications.

IV. METHOD FOR CONSTRUCTING THE GLOBAL DETECTION MODEL

Global detection model is constructed by combining the local parametric detection model from each nodes. Which is then used to detect intrusion on each distributed site.

Kittler et al.[10] develop a different framework for combining the local model like, product rule, the sum rule, the max rule, the mirule, the median rule, and the majority vote rule. But by using this rule local detection model has two problem a) performance gap between the new type of attacks and local detection model. b) Dimension of the vector for similar test sample at the local models. The solution to this problem is combine local model by using PSO and SVM algorithms. PSO[11],[12] is a population search algorithm and the SVM is a learning algorithm, so by using the searching and learning ability of PSO and SVM respectively a global intrusion detection model is constructed in each node. The global intrusion detector constructed in the following simple manner:

$$G(n)=\begin{cases}-1 & \text{if there exist } C(n)=-1\\ 1 & \text{else}\end{cases} \quad (3)$$

C(n) is final strong classifier generated by Adaboost training.

Two things for global detection models are:

i) Global models constructed for all local nodes are uniform.
ii) The computational complexity of the PSO is O(QIA2L2) where I is the number of iterations, and L is the number of the training samples.

V. EXPERIMENTS

We utilize the knowledge discovery and data mining (KDD) CUP 1999 dataset [13]–[15], [16] to test algorithms. It has served as a reliable benchmark data set for many network intrusion detection algorithms. In this data set, each TCP/IP connection was labelled and 41 continues or categorical feature were extracted (41 features including 9 categorical features and 32 continuous features for each network connection). Attacks in the

dataset fall into four main categories. i) denial of service (DOS). ii) user to root (U2R). iii) remote to local (R2L). iv) PROBE.

The number of sample of various types in the training set and in the test set are listed in Table 1.

TABLE I

The KDD CUP 1999 Dataset[1]

Categories	Training data	Test data
Normal	97 278	60 593
DOS	391 458	223 298
R2L	1126	5993
U2R	52	39
Probing	4107	2377
Others	0	18 729
Total	494 021	311 029

Table 2. shows the comparison of online Adaboost-based algorithms with other recently published algorithms for intrusion detection.

TABLE 2
COMPARATIVE STUDY OF VARIOUS ALGORITHMS TESTED ON THE KDD CUP 1999 DATA SET[1]

	Algorithms	Detection rate (%)	False alarm rate (%)
Offline	Clustering [3]	93	10
	K-NN [20]	91	8
	SVM [20]	91-98	6-10
	SOM [5]	89-90.6	4.6-7.6
	Genetic clustering[17]	79	0.30
	Hierarchical SOM [18]	90.94-93.46	2.19-3.99
	Bagged C5 [19]	91.81	0.55
	Offline Adaboost [39]	90.04-90.88	0.31-1.79
Online	Mercer kernel ART [8]	90-94	2.9-3.4
	Our algorithm (decision stumps+traditional Adaboost) [1]	90.13	2.23
	Our algorithm (Online GMMs+our Adaboost) [1]	90.61-91.15	1.17-1.69

VI. CONCLUSION

This paper has presented a survey and comparative study of the Adaboost-based algorithms, that have been proposed towards the improvement of the Dynamic DIDS. We have shown the how online Adaboost algorithm has been helpful for new type of attacks and the way of how it work. The main objective of system is to detect the new intrusion .

Finally we propose the comparative study of algorithms tested on the KDD CUP 1999 dataset.

VII. ACKNOWLEDGEMENTS

The presented paper would not have been possible without college Dr. D. Y. Patil COE, Ambi, Pune. I got support from my family and friends. I thankful to the Prof. Dhanashree Kulkarni who guide me, which help me in improving my work, from this I learnt many new things. Thank you.

REFERENCES

[1] Weiming Hu, Jun Gao, Yanguo Wang, Ou Wu, and Stephen Maybank," Online Adaboost-Based Parameterized Methods for Dynamic Distributed Network Intrusion Detection," IEEE TRANSACTIONS ON CYBERNETICS, VOL. 44, NO. 1, JANUARY 2014

[2] D. Smallwood and A. Vance, “Intrusion analysis with deep packet inspection: Increasing efficiency of packet based investigations,” in Proc. Int. Conf. Cloud Service Computing, Dec. 2011, pp. 342–347.

[3] D. Denning, “An intrusion detection model,” IEEE Trans. Softw. Eng., vol. SE-13, no. 2, pp. 222–232, Feb. 1987.

[4] W. Lee, S. J. Stolfo, and K. Mork, “A data mining framework for building intrusion detection models,” in Proc. IEEE Symp. Security Privacy, May 1999, pp. 120–132.

[5] M. Qin and K. Hwang, “Frequent episode rules for internet anomaly detection,” in Proc. IEEE Int. Symp. Netw. Computing Appl., 2004, pp. 161–168.

[6] S. Mukkamala, G. Janoski, and A. Sung, “Intrusion detection using neural networks and support vector machines,” in Proc. Int. Joint Conf. Neural Netw., vol. 2. 2002, pp. 1702–1707.

[7] S. Parthasarathy, A. Ghoting, and M. E. Otey, “A survey of distributed mining of data streams,” in Data Streams: Models and Algorithms. C. C. Aggarwal (Ed.) New York: Springer, Nov. 2006.

[8] M. E. Otey, A. Ghoting, and S. Parthasarathy, “Fast distributed outlier detection in mixed-attribute data sets,” Data Ming Knowl. Discovery, vol. 12, no. 2–3, pp. 203–228, May 2006.

[9] B. Pfahringer, “Winning the KDD99 classification cup: Bagged boosting,” SIGKDD Explorations, vol. 1, no. 2, pp. 65–66, 2000.

[10] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” J. Comput. Syst. Sci., vol. 55, no. 1, pp. 119–139, Aug. 1997.

[11] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, “On combining classifiers,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 3, pp.226–238, Mar. 1998.

[12] J. Kennedy, “Particle swarm optimization,” in Proc. IEEE Int. Conf. Neural Netw., 1995, pp. 1942–1948.

[13] Y. Shi and R. C. Eberhart, “A modified particle swarm optimizer,” in Proc. IEEE Int. Conf. Evolut. Comput., 1998, pp. 69–73.

[14] S. Stofo et al. The Third International Knowledge Discovery and Data Mining Tools Competition, The University of California, 2002 [Online]. Available: http://kdd.ics.uci.edu/databases/kddCup99/kddCup99.h-tml.

[15] S. Mukkamala, A. H. Sung, and A. Abraham, "Intrusion detection using an ensemble of intelligent paradigms," Netw. Comput. Appl., vol. 28, no. 2, pp. 167–182, Apr. 2005.

[16] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, "The 1999 DARPA off-line intrusion detection evaluation," ACM Trans. Inform. Syst. Security, vol. 34, no. 4, pp. 579–595, Oct. 2000.

[17] J. McHugh, "Testing intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln laboratory," ACM Trans. Inform. Syst. Security, vol. 3, no. 4, pp. 262–294, Nov. 2000.

[18] Z. Zhang and H. Shen, "Online training of SVMs for real-time intrusion detection," in Proc. Adv. Inform. Netw. Appl., vol. 2, 2004, pp. 568–573.

[19] J. Kennedy, "Particle swarm optimization," in Proc. IEEE Int. Conf. Neural Netw., 1995, pp. 1942–1948.

[20] W. Lee and S. J. Stolfo, "A framework for constructing features and models for intrusion detection systems," ACM Trans. Inform. Syst. Security, vol. 3, no. 4, pp. 227–261, Nov. 2000.

[21] S. T. Sarasamma, Q. A. Zhu, and J. Huff, "Hierarchical Kohonenen net for anomaly detection in network security," IEEE Trans. Syst., Man Cybern., Part B: Cybern., vol. 35, no. 2, pp. 302–312, Apr. 2005.

[22] S. J. Han and S. B. Cho, "Evolutionary neural networks for anomaly detection based on the behavior of a program," IEEE Trans. Syst., Man, Cybern.—Part B, vol. 36, no. 3, pp. 559–570, Jun. 2006.

[23]] D. Song, M. I. Heywood, and A. N. Zincir-Heywood, "Training genetic programming on half a million patterns: An example from anomaly detection," IEEE Trans. Evolut. Comput., vol. 9, no. 3, pp. 225–239, Jun. 2005.

[24] W. M. Hu, W. Hu, and S. Maybank, "Adaboost-based algorithm for network intrusion detection," IEEE Trans. Syst., Man, Cybern., Part B: Cybern., vol. 38, no. 2, pp. 577–583, Apr. 2008.

[25] W. M. Hu, W. Hu, and S. Maybank, "Adaboost-based algorithm for network intrusion detection," IEEE Trans. Syst., Man, Cybern., Part B: Cybern., vol. 38, no. 2, pp. 577–583, Apr. 2008.

[26] H. Grabner and H. Bischof, "On-line boosting and vision," in Proc. IEEE Conf. Comput. Vision Pattern Recogn., 2006, pp. 260–267.

[27] Y. Lei, X. Q. Ding, and S. J. Wang, "Visual tracker using sequential Bayesian learning: Discriminative, generative and hybrid," IEEE Trans. Syst., Man Cybern., Part B: Cybern., vol. 38, no. 6, pp. 1578–1591, Dec. 2008.

[28] S. Mabu, C. Chen, N. Lu, K. Shimada, and K. Hirasawa, "An intrusiondetection
model based on fuzzy class-association-rule mining using genetic network programming," IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev., vol. 41, no. 1, pp. 130–139, Jan. 2011.