International Journal of Engineering Research and Development
***e**-ISSN: 2278-067X, **p**-ISSN: 2278-800X, www.ijerd.com*
Volume 3, Issue 11 (September 2012), PP. 56-61

Detection of Intrusion Using an Agent Based Approach

V. Nitesh, S.Siva Shankar Rao

Abstract*—Intrusion detection is one of the high priority and challenging tasks for network administrators and security professionals. There is a need to safeguard the networks from known vulnerabilities and at the same time take steps to detect new and unseen, by developing more reliable and efficient intrusion detection systems. In this paper we propose an agent based intrusion detection that helps to provide rapid response to vulnerabilities. There is a need to safeguard the networks from Known vulnerabilities and at the same time take steps to detect new and unseen, but possible, system abuses by developing more reliable and efficient intrusion detection systems. This system can also be implemented in various homogeneous and heterogeneous environments. Intrusion detection faces a number of challenges, an intrusion detection system must reliably detect malicious activities in a network and must perform efficiently to cope with the large amount of network traffic. In this paper, we address the two issues of Accuracy and Efficiency using Conditional Random Fields and Layered Approach. We demonstrate that high attack detection accuracy can be achieved by using Conditional Random Fields and high efficiency by implementing the Layered Approach.

***Index Terms*—Intrusion detection, Layered Approach, Conditional Random Fields, network security, agent based.**

I. INTRODUCTION

The Intrusion Detection System can be defined as the detection of Intrusion or intrusions attempts either manually or via software expert systems that operate on logs or other information available from the system or the network. An intrusion is nothing but a deliberate, unauthorized attempt to access or manipulate formation or system and to render them unreliable or unusable. Intrusion Detection Systems look for attack signatures, which are specific patterns that usually indicate malicious or suspicious intent. It must be supplemented by other security and protection mechanisms. They are a very important part of our security architecture but do not solve all our problems.

Current state of IDS is much of our interest, because lots of people are still using Firewall and Router logs for Intrusion detection.IDS are not very mature. Mostly signature based IDS are in practice. But it is quickly evolving domain. The important function of IDS is to monitor and analyze user and system activities. IDS asses' integrity of critical system and data files.

Firewall (computing), a technological barrier designed to prevent unauthorized or unwanted communications between computer networks or hosts.

We introduced novel frameworks and developed models which address three critical issues that severely affect the large scale deployment of present anomaly and hybrid intrusion detection systems in high speed networks. The three issues are:

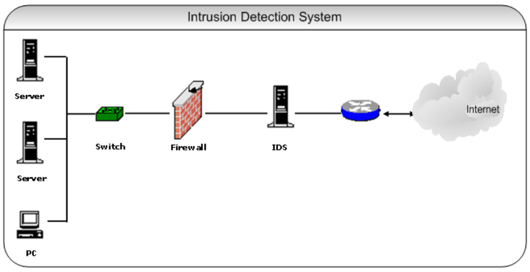
1. Limited attack detection coverage
2. Large number of false alarms and
3. Inefficiency in operation

Another approach for detecting intrusions is to consider both the normal and the known anomalous patterns for training a system and then performing classification on the test data. Such a system incorporates the advantages of both the signature-based and the anomaly-based systems and is known as the Hybrid System.

Conditional models are probabilistic systems that are used to model the conditional distribution over a set of random variables. Such models have been extensively used in the natural language processing tasks. Conditional models offer a better framework as they do not make any unwarranted assumptions on the observations and can be used to model rich overlapping features among the visible observations. Maxent classifiers [1], maximum entropy Markov models [2], and CRFs [3] are such conditional models. The advantage of CRFs is that they are undirected and are, thus, free from the Label Bias and the Observation Bias.

II. RELATED WORK

A number of frameworks have been proposed for building intrusion detection systems, Agent based intrusion detection frameworks are discussed. All of these frameworks suffer from one major drawback; a single intrusion detector used within these frameworks is trained to detect a wide variety of attacks. This results in a large number of false alarms. To ameliorate this, we introduce our Layered Framework for building Intrusion Detection Systems.



2.1 FRAMEWORKS FOR BUILDING INTRUSION DETECTION SYSTEMS

The field of intrusion detection and network security has been around since late 1980s .A number of frameworks have been proposed for building intrusion detection systems. A number of frameworks have been proposed for building intrusion detection systems. Various techniques such as association rules, clustering, naive Bayes classifier, support vector machines, genetic algorithms, artificial neural networks, and others have been applied to detect intrusions. In this section, we briefly discuss these techniques and frameworks. The authors describe a data mining framework for building intrusion detection systems. Agent based intrusion detection frameworks are discussed. All of these frameworks suffer from one major drawback; a single intrusion detector used within these frameworks is trained to detect a wide variety of attacks. This results in a large number of false alarms. To ameliorate this, we introduce our Layered Framework for building Intrusion Detection Systems.

2.1 DATA MINING AND MACHINE LEARNING

Data mining and machine learning methods focus on analyzing the properties of the audit patterns rather than identifying the process which generated them. These methods include approaches for mining association rules, classification and cluster analysis. Classifications methods are one of the most researched and include methods like the decision trees, Bayesian classifiers etc.

Data Mining

Data mining approaches are based on mining association rules and using frequent episodes to build classifiers by discovering relevant patterns of program and user behavior. Association rules and frequent episodes are used to learn the record patterns that describe user behavior. These approaches can deal with symbolic features and the features can be defined in the form of packet and connection details. Mining association rules for intrusion detection has the advantage that they are easy to interpret.

However, they are based upon building a database of rules of normal and frequent items during the training phase. During testing, patterns from the test data are extracted and various classification methods can be used to classify the test data. The detection accuracy suffers as the database of rules is not a complete representation of the normal audit patterns.

Bayesian Classifiers

Naive Bayes classifiers are also proposed for intrusion detection. However, they make strict independence assumption between the features in an observation resulting in lower attack detection accuracy when the features are correlated, which is often the case. Bayesian network can also be used for intrusion detection. However, they tend to be attack specific and build a decision network based on special characteristics of individual attacks. As a result, the size of a Bayesian network increases rapidly as the number of features and the type of attacks modeled by the network increases.

Decision Trees

Decision trees have also been used for intrusion detection. Decision trees select the best features for each decision node during tree construction based on some well defined criteria. One such criterion is the gain ratio which is used. Decision trees generally have very high speed of operation and high attack detection accuracy and have been successfully used to build effective intrusion detection systems.

They, however, consider a data mining approach for mining association rules and finding frequent episodes in order to calculate the support and confidence of the rules. Instead, in our work we define features from the observations as well as from the observations and the previous labels and perform sequence labeling via the conditional random fields to label every feature in the observation. This setting is sufficient to model the correlation between different features in an observation. We train our system using both the normal and the anomalous patterns i.e. we build a hybrid system.

However, network intrusion detection systems must perform very efficiently in order to handle large amount of network data and hence many of the network intrusion detection systems are primarily based on signature matching. When anomaly detection systems are used at network level, they either consider only one feature or assume the features to be independent. However, we propose to use a hybrid system based on conditional random fields and integrate the layered framework to build a single system which can operate in high speed networks and can detect a wide variety of attacks with very few false alarms. We, thus, present the Layered Conditional Random Fields for Network Intrusion Detection.

Secondly, the system fails to model long range dependencies in the observations, which can be easily represented in our model. We integrate the layered framework with the conditional random fields to gain the benefits of computational efficiency, wide attack detection coverage and high accuracy of attack detection in a single system.

2.2 CONDITIONAL RANDOM FIELDS

Conditional models are probabilistic systems which are used to model the conditional distribution over a set of random variables. Such models have been extensively used in natural language processing tasks and computational biology.

Conditional models offer a better framework as they do not make any unwarranted assumptions on the observations and can be used to model rich overlapping features among the visible observations. Maxent classifiers maximum entropy Markov models and conditional random fields are such conditional models.

The Conditional random fields are undirected graphical models; thus, offer us the required framework to build effective intrusion detection systems. The task of intrusion detection can be compared to many problems in machine learning, natural language processing such as part of speech tagging, text segmentation and many others. The conditional random fields have proven to be very successful in such tasks. Hence we explore the suitability of conditional random fields for building robust intrusion detection systems.

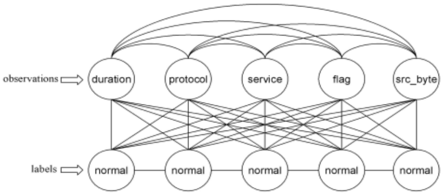


Fig.1.Graphical representation of a CRF.

The task of Intrusion Detection, however, has some major requirements. It has to be an online task and there is no knowledge available for the future observations. Further, once deployed, the system has to deal with large amount of data and thus it must be able to perform fast enough to be effective. Conditional Random Fields satisfy all of these requirements and once the model has been trained and deployed, they are very fast in labeling the data as either normal or as attack. The complexity of a Conditional Random Field is quadratic with respect to the number of labels. This is problematic when the number of labels is large, such as in the language tasks, but in our case we have only two labels; normal and attack. We observe that the training of a Conditional Random Field is expensive but once trained their performance is comparable to that of the Decision Trees and Naive Bayes classifiers. Thus our system is very efficient and can be used online.

III. OUR PROPOSED

We propose a framework for intrusion detection which we call as the LBIDS. To ensure complete network security i.e. to provide confidentiality, integrity and availability, we need a system which is both specific in detecting attacks targeted individually by selecting only a small set of features which are significant to detection for a particular category, as well as is capable to correlate the results to ensure complete network security. The system not only needs to perform this task with high accuracy but also needs to do it at a stage as early as possible as it reduces the effect of the attack and also reduces the computation required by the system.

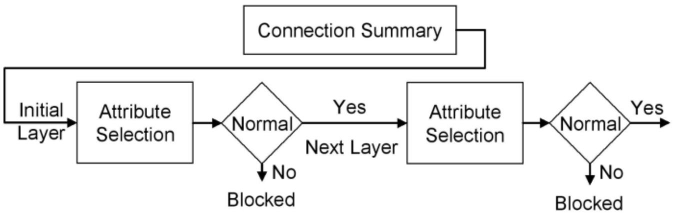


Fig. 2. Layered representation.

Our framework examines different attributes at different layers to effectively identify any breach of security at every layer. This has the advantage that we can effectively divide the computation into smaller parts and if at any stage/layer the system makes a decision that there is an attack, it can simply block that intrusion and save the higher layers from performing any further computation, rather than making a decision by aggregating entire data at a single point as is commonly used in any well known Intrusion Detection System.

Every layer in the LIDS framework is trained separately and then deployed sequentially. We define four layers that correspond to the four attack groups. They are Probe layer, DoS layer, R2L layer, and U2R layer. Each layer is then separately trained with a small set of relevant features. Feature selection is significant for Layered Approach and in order to make the layers independent, some features may be present in more than one layer. The layers essentially act as filters that block any anomalous connection, thereby eliminating the need of further processing at subsequent layers enabling quick response to intrusion. The effect of such a sequence of layers is that the anomalous events are identified and blocked as soon as they are detected.

This ensures that each layer is stand alone and is able to effectively block the type of intrusion which it is meant to block. Sharing of some features from previous layers is necessary to ensure that the layers are linked together. This is important because as we move to any higher layer, various semantic features needs to be related to the non-semantic features such as connection features to ensure better detection capabilities.

We first select four layers corresponding to the four attack groups (Probe, DoS, R2L, and U2R) and perform feature selection for each layer.

Probe Layer

The probe attacks are aimed at acquiring information about the target network from a source that is often external to the network. Hence, basic connection level features such as the “duration of connection” and “source bytes” are significant while features like “number of files creations” and “number of files accessed” are not expected to provide information for detecting probes.

DoS Layer

The DoS attacks are meant to force the target to stop the service(s) that is (are) provided by flooding it with illegitimate requests. Hence, for the DoS layer, traffic features such as the “percentage of connections having same destination host and same service” and packet level features such as the “source bytes” and “percentage of packets with errors” is significant. To detect DoS attacks, it may not be important to know whether a user is “logged in or not.”

R2L Layer

The R2L attacks are one of the most difficult to detect as they involve the network level and the host level features. We therefore selected both the network level features such as the “duration of connection” and “service requested” and the host level features such as the “number of failed login attempts” among others for detecting R2L attacks.

U2R Layer

The U2R attacks involve the semantic details that are very difficult to capture at an early stage. Such attacks are often content based and target an application. Hence, for U2R attacks, we selected features such as “number of file creations” and “number of shell prompts invoked,” while we ignored features such as “protocol” and “source bytes.

IV. ALGORITHM

Training

Step 1: Select the number of layers, n, for the complete system.
Step 2: Separately perform features selection for each layer.
Step 3: Train a separate model with CRFs for each layer using the features select from step 2.
Step 4: Plug in the trained models sequentially such that only the connections labeled as normal are passed to the next layer.

Testing

Step 5: For each (next) test instance perform Steps 6 through 9.
Step 6: Test the instance and label it either as attack or normal.
Step 7: If the instance is labeled as attack, block it and identify it as an attack represented by the layer name at which it is detected and go to Step 5. Else pass the sequence to the next layer.
Step 8: If the current layer is not the last layer in the system, test the instance and go to Step 7. Else go to Step 9.
Step 9: Test the instance and label it either as normal or as an attack. If the instance is labeled as an attack, block it and identify it as an attack corresponding to the layer name.

Our second goal is to improve the speed of operation of the system. Hence, we implement the LIDS and select a small set of features for every layer rather than using all the features. This results in significant performance improvement during both the training and the testing of the system. In many situations, there is a trade-off between efficiency and accuracy of the system and there can be various avenues to improve system performance. Methods such as naive Bayes assume independence among the observed data. This certainly increases system efficiency, but it may severely affect the accuracy. To balance this trade-off, we use the CRFs that are more accurate, though expensive, but we implement the Layered Approach to improve overall system performance.

Our framework has the advantage that it is not specific to any particular type or group of attack as we address the basic features of security and bind them together in a single system rather than creating different system for ensuring each security aspect. Our framework essentially provides a method that can help to reduce the complexity of the system by simply dividing the task into a sequence of tasks. Such a system would be less computational intensive and more accurate.

Attack on confidentiality and integrity of data are emerging trends in network intrusion. Smart agents’ searches for specific information, in a private network connected to the internet. The information required is not simple such as getting the competitors' price checks but is more related to marketing and pricing strategy, actual financial status of an organization etc. The agent is able to hide itself from being discovered by resorting to stealth scan, replicate itself in limited numbers depending upon the situation, transmit the identified information and destroy itself and its traces after completing its task. The whole idea is that such an agent is invisible to the network monitors and it performs its task without being discovered and exposing its identity.

In order to deter such intrusive activities we require building strong defense systems as well as an equally strong legal system that would discourage such attacks. For the legal system to be effective it requires the source to be correctly identified and then collecting strong evidence against the identified source. However, more work is required to correctly identify the source of intrusion as the proposed agent takes every step to hide its identity. One possible method to detect an agent activity is to take a periodic snapshot of activities of the processes in a secure media such as write once only media which can be used for further analysis and for building evidence of intrusion.

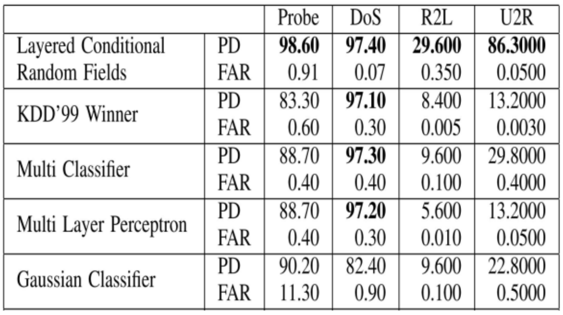
A number of intrusion detection systems have been developed which operate at the network level. However, network intrusion detection systems must perform very efficiently in order to handle large amount of network data and hence many of the network intrusion detection systems are primarily based on signature matching. When anomaly detection systems are used at network level, they either consider only one feature or assume the features to be independent. However, we propose to use a hybrid system based on conditional random fields and integrate the layered framework to build a single system which can operate in high speed networks and can detect a wide variety of attacks with very few false alarms. Further, our system can be used as a standalone system monitoring an entire Network or a single Host or even a single Application running on a particular host.

V. EXPERIMENTAL RESULTS

In this paper, the agent needs to be registered initially. If an intruder tries to login using unmatch passwords, the intruder is blocked. After valid login, still there is chance that the agent or a user can be an intruder. They can be block at sequence layers. At each layers, the respective attacks are identified. When the agent tries to access unauthorized services, the agent will be identified as an intruder. The respective message is being sent to admin at all layers if intruder is being detected. If an unauthorized is trying to send a file, the intruder is blocked. At all layers, the intruder is blocked and the respective intruder is killed. Hence using this agent based, we are able to detect intrusions.

In our comparison of results experiments show that the Layered CRFs perform far better than these techniques. The main reason for this is that the CRFs do not consider the observation features to be independent. In [4], the authors present a comparative study of various classifiers when applied to theKDD'99 data set, and in [6], the authors propose the use of Principle Component Analysis (PCA) before applying a machine learning algorithm. We compare our results from the results presented in these papers in Table 1. The table represents the Probability of Detection (PD) and False Alarm Rate (FAR) in percent for various methods including the KDD '99 cup winners. From the table, we observe that the Layered CRFs perform significantly better than the previously reported results including the winner of the KDD '99 cup and various other methods applied to this data set. The most impressive part of the Layered CRFs is the margin of improvement as compared with other methods. Layered CRFs have very high attack detection of 98.6 percent for Probes (5.8 percent improvement) and 97.40 percent detection for DoS. They outperform by a significant percentage for the R2L (34.5 percent improvement) and the U2R (34.8 percent improvement) attacks.

TABLE 1 : Comparison of Results



		Probe	DoS	R2L	U2R
Layered Conditional Random Fields	PD	**98.60**	**97.40**	**29.600**	**86.3000**
	FAR	0.91	0.07	0.350	0.0500
KDD'99 Winner	PD	83.30	**97.10**	8.400	13.2000
	FAR	0.60	0.30	0.005	0.0030
Multi Classifier	PD	88.70	**97.30**	9.600	29.8000
	FAR	0.40	0.40	0.100	0.4000
Multi Layer Perceptron	PD	88.70	**97.20**	5.600	13.2000
	FAR	0.40	0.30	0.010	0.0500
Gaussian Classifier	PD	90.20	82.40	9.600	22.8000
	FAR	11.30	0.90	0.100	0.5000

VI. CONCLUSIONS

We have addressed the dual problem of Accuracy and Efficiency for building robust and efficient intrusion detection systems. Our experimental results in Section 6 show that CRFs are very effective in improving the attack detection rate and decreasing the FAR. Having a low FAR is very important for any intrusion detection system. Further, feature selection and implementing the Layered Approach significantly reduce the time required to train and test the model. The areas for future research include the use of our method for extracting features that can aid in the development of signatures for signature-based systems. The signature-based systems can be deployed at the periphery of a network to filter out attacks that are frequent and previously known, leaving the detection of new unknown attacks for anomaly and hybrid systems. Sequence analysis methods such as the CRFs when applied to relational data give us the opportunity to employ the Layered Approach. This can further be extended to implement pipelining of layers in multicore processors, which is likely to result in very high performance.

ACKNOWLEDGMENTS

We would like to thank the anonymous referee for helpful Comments

REFERENCES

[1]. A. Ratnaparkhi, "A Maximum Entropy Model for Part-of-Speech Tagging," Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP '96), pp. 133-142, Assoc. for Computational Linguistics, 1996

[2]. A. McCallum, D. Freitag, and F. Pereira, "Maximum Entropy Markov Models for Information Extraction and Segmentation," Proc. 17th Int'l Conf. Machine Learning (ICML '00), pp. 591-598, 2000.

[3]. J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," Proc. 18th Int'l Conf. Machine Learning (ICML '01), pp. 282-289, 2001.

[4]. M. Sabhnani and G. Serpen, "Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context," Proc. Int'l Conf. Machine Learning, Models, Technologies and Applications (MLMTA '03), pp. 209-215, 2003.

[5]. Z. Zhang, J. Li, C.N. Manikopoulos, J. Jorgenson, and J. Ucles. Hide: a hierarchical network intrusion detection system using statistical preprocessing and neural network classification. In Proceedings of the 2001 IEEE Workshop on Information Assurance and Security United States Military Academy, 2001, pages 85–90.

[6]. Y. Bouzida and S. Gombault, "Eigenconnections to Intrusion Detection," Security and Protection in Information Processing Systems,pp. 241-258, 2004.

Author Bibliography

Mr.V.Nitesh received his B.Tech in Computer science and Engineering from SCIENT Institute of Technology, JNTU, Hyderabad and Pursuing M.Tech in Computer science Engineering from Aurora's Technological And Research Institute, JNTU, Hyderabad.

Mr. S.Siva Sankar Rao working as Assoc Professor in the Department of Computer Science and Engineering, in Aurora's Technological And Research Institute with a teaching experience of 9 years.