Volume : 2 | Issue : 2 | Feb 2013 • ISSN No 2277 - 8179

Research Paper

Engineering

KEYWORDS :

Intrusion Detection by Implementing CRF's Using the Technique of Layered Approach

Baji Yadala — M.Tech Student (CSE), Gudlavalleru Engineering College

Shaik Salma Begum — Assistant Professor, Dept of CSE, Gudlavalleru Engineering Colleg

ABSTRACT

To function effectively with high amount of network traffic an intrusion system must be have capability of detecting malevolent activities in a network. Intrusion detection system faces a number of challenges in the real world. Using the techniques of both Conditional Random Fields and Layered Approach we have addressed the Accuracy and Efficiency. Considering examples we exhibited that high attack detection accuracy can be achieved by using Conditional Random Fields and high efficiency by implementing the Layered Approach. Our experimental results on benchmark KDD '99 intrusion data set show that our proposed system based on Layered Conditional Random Fields has better performance as compared to decision trees and the naive Bayes methods. For our methods mathematical testing also provided higher confidence in detection accuracy. Thus we proved that our system is strong and can handle noisy free data providing with high performance.

I. INTRODUCTION

Intrusion detection (ID) is a type of security management system for computers and networks. An ID system gathers and analyzes information from various areas within a computer or a network to identify possible security breaches, which include both intrusions (attacks from outside the organization) and misuse (attacks from within the organization). ID uses vulnerability assessment (sometimes referred to as scanning), which is a technology developed to assess the security of a computer system or network.

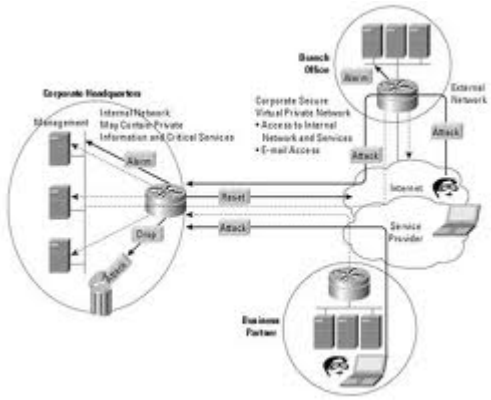

Fig.1: Intrusion Detection System

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. Hybrid systems can be very efficient, subject to the classification method used, and can also be used to label unseen or new instances as they assign one of the known classes to every test instance. This is possible because during training the system learns features from all the classes. The only concern with the hybrid method is the availability of labelled data. However, data requirement is also a concern for the signature- and the anomaly-based systems as they require completely anomalous and attack free data, respectively, which are not easy to ensure.

Intrusion detection functions include:

The functionalities of detecting the intrusions include as below:

- Monitoring and analyzing both user and system activities
- Analyzing system configurations and vulnerabilities
- Assessing system and file integrity
- Ability to recognize patterns typical of attacks
- Analysis of abnormal activity patterns.
- Tracking user policy violations.

ID systems are being developed in response to the increasing number of attacks on major sites and networks, including those of the Pentagon, the White House, NATO, and the U.S. Defence Department. The safeguarding of security is becoming increasingly difficult, because the possible technologies of attack are becoming ever more sophisticated; at the same time, less technical ability is required for the novice attacker, because proven past methods are easily accessed through the Web. Typically, an ID system follows a two-step process. The first procedures are host-based and are considered the passive component, these include: inspection of the system's configuration files to detect inadvisable settings; inspection of the password files to detect inadvisable passwords; and inspection of other system areas to detect policy violations. The second procedures are network-based and are considered the active component: mechanisms are set in place to reenact known methods of attack and to record system responses.

In 1998, ICSA.net, a leading security assurance organization, formed the Intrusion Detection Systems Consortium (IDSC) as an open forum for ID product developers with the aim of disseminating information to the end user and developing industry standards. One preliminary IDS concept consisted of a set of tools intended to help administrators review audit trails. User access logs, file access logs, and system event logs are examples of audit trails.

Fred Cohen noted in 1984 that it is impossible to detect an intrusion in every case and that the resources needed to detect intrusions grow with the amount of usage. Dorothy E. Denning, assisted by Peter G. Neumann, published a model of IDS in 1986 that formed the basis for many systems today. Her model used statistics for anomaly detection, and resulted in early IDS at SRI International named the Intrusion Detection Expert System (IDES), which ran on Sun workstations and could consider both user and network level data. IDES had a dual approach with a rule-based Expert System to detect known types of intrusions plus a statistical anomaly detection component based on profiles of users, host systems, and target systems. Lunt proposed adding an artificial neural network as a third component. She said all three components could then report to a resolver. SRI followed IDES in 1993 with the Next-generation Intrusion Detection Expert System (NIDES). The Multics intrusion detection and alerting system (MIDAS), an expert system using P-BEST and Lisp, was developed in 1988 based on the work of Denning and Neumann. Haystack was also developed this year using statistics to reduce audit trails. Wisdom & Sense (W&S) was a statistics-based anomaly detector developed in 1989 at the Los Alamos National Laboratory. W&S created rules based on statistical analysis, and then used those rules for anomaly detection.

In 1990, the Time-based Inductive Machine (TIM) did anomaly detection using inductive learning of sequential user patterns in Common Lisp on a VAX 3500 computer. The Network Security Monitor (NSM) performed masking on access matrices for

anomaly detection on a Sun-3/50 workstation.[14] The Information Security Officer's Assistant (ISOA) was a 1990 prototype that considered a variety of strategies including statistics, a profile checker, and an expert system. Computer Watch at AT&T Bell Labs used statistics and rules for audit data reduction and intrusion detection.

The signature-based system sare trained by extracting specific patterns (or signatures) from previously known attacks while the anomaly-based systems learn from the normal data collected when there is no anomalous activity. 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. Hybrid systems can be very efficient, subject to the classification method used, and can also be used to label unseen or new instances as they assign one of the known classes to every test instance. This is possible because during training the system learns features from all the classes. The only concern with the hybrid method is the availability of labelled data. However, data requirement is also a concern for the signature- and the anomaly-based systems as they require completely anomalous and attack free data, respectively, which are not easy to ensure. We observe that our proposed system, Layered CRFs, performs significantly better than other systems.

II. BACKGROUND WORK & IDS APPROACHES

This section provides a brief literature review on these technologies and related frameworks. These methods can be broadly divided into three major categories:

A. Statistical Methods

Statistical modeling is among the earliest methods used for detecting intrusions in electronic information systems. It is assumed that an intruder's behaviour is noticeably different from that of a normal user, and statistical models are used to aggregate the user's behavior and distinguish an attacker from a normal user. The techniques are applicable to other subjects, such as user groups and programs. Two statistical models that have been proposed for anomaly detection: NIDES/STAT and Haystack.

B. Pattern Matching

Pattern Matching is the simple type of attack detection technique. It has the simple concept of string matching. Using pattern matching technique, IDSs generally match the text (audit records) or binary sequences against known attack signatures. A pattern matching technique basically looks for a specific attack signature which may be presented in audit record. The limitation of pattern matching approach is that it can recognize only known attacks. It requires continuous updates of attack signatures to identify new attacks. Pattern matching approach is well suited for misuse detection. Snort system is based upon pattern matching.

C. 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.

Decision Trees: Decision trees are one of the most commonly used supervised learning algorithms in IDS due to its simplicity, high detection accuracy and fast adaptation. Decision trees used for intrusion detection select the best features for each decision node during tree construction based on some well-defined criteria.

Artificial Neural Networks: Neural networks are known for good performance in learning system-call sequences. Once the neural net is trained on a set of representative command sequences of a user, the net constitutes the profile of the user, and the fraction of incorrectly predicted events then measures, in some sense, the variance of the user behavior from his profile. They can work effectively with noisy data but they require large amount of data for training and it is often hard to select the best possible architecture for the neural network.

Clustering: For unsupervised intrusion detection, data clustering methods can be applied. These methods involve computing a distance between numeric features and therefore they cannot easily deal with symbolic attributes, resulting in inaccuracy. Addition, clustering methods consider the features independently and are unable to capture the relationship between different features of a single record which results in lower accuracy.

Data Mining: Data mining (DM), also called Knowledge-Discovery and Data Mining, is the process of automatically searching large volumes of data for patterns using association rules. Data mining approaches derive association rules and frequent episodes from available sample data, not from human experts. Using these rules, Lee et. al. developed a data mining framework for the purpose of intrusion detection. In particular, system usage behaviors are recorded and analyzed to generate rules which can recognize misuse attacks. The drawback of such frameworks is that they tend to produce a large number of rules and thereby, increase the complexity of the system.

Bayesian Classifiers: A Bayesian network is a model that encodes probabilistic relationships among variables of interest. This technique is generally used for intrusion detection in combination with statistical schemes, a procedure that yields several advantages, including the capability of encoding interdependencies between variables and of predicting events, as well as the ability to incorporate both prior knowledge and data. However, a serious disadvantage of using Bayesian networks is that their results are similar to those derived from threshold-based systems, while considerably higher computational effort is required.

III. REQUIREMENT FOR INTRUSION DETECTION SYSTEM

Functional Requirements should include:

- Descriptions of data to be entered into the system
- Descriptions of operations performed by each screen
- Descriptions of work-flows performed by the system
- Descriptions of system reports or other outputs
- Who can enter the data into the system
- How the system meets applicable regulatory requirements

IV. SYSTEM IMPLEMENTATION

Other approaches for detecting intrusion include the use of autonomous and probabilistic agents for intrusion detection. These methods are generally aimed at developing a distributed intrusion detection system. To overcome the weakness of a single intrusion detection system, a number of frameworks have been proposed, which describe the collaborative use of network-based and host based systems. Systems that employ both signatures based and behavior-based techniques are discussed in the authors describe a data mining framework for building adaptive intrusion detection models.

Conditional Random field:

The CRFs have proven to be very successful in such tasks, as they do not make any unwarranted assumptions about the data. Hence, we explore the suitability of CRFs for intrusion detection. System may consider features such as "logged in" and "number of file creations." When these features are analyzed individually, they do not provide any information that can aid in detecting attacks. However, when these features are analyzed together, they can provide meaningful information, which can be helpful for the classification task. Taking another example, the connection level feature such as the "service invoked".

Fig.2: Practical approach of CRF

Layered approach for intrusion detection:
The Layer-based Intrusion Detection System (LIDS) in detail. The LIDS draws its motivation from what we call as the Airport Security model, where a number of security checks are performed one after the other in a sequence. Similar to this model, the LIDS represents a sequential Layered Approach and is based on ensuring availability, confidentiality, and integrity of data and (or) services over a network. Fig. 3 gives a generic representation of the framework. 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

The discussed two main requirements for an intrusion detection system; accuracy of detection and efficiency in operation. As discussed in, the CRFs can be effective in improving the attack detection accuracy by reducing the number of false alarms, while the Layered Approach can be implemented to improve the overall system efficiency. Hence, a natural choice is to integrate them to build a single system that is accurate in detecting attacks and efficient in operation. Given the data, we first select four layers corresponding to the four attack groups (Probe, DoS, R2L, and U2R) and perform feature selection for each layer, which is described next. The used domain knowledge together with the practical significance and the feasibility of each feature before selecting it for a particular layer. Thus, from the total 41 features, we selected only 5 features for Probe layer, 9 features for DoS layer, 14 features for R2L layer, and8 features for U2R layer. Since each layer is independent of every other layer, the feature set for the layers is not disjoint We then use the CRFs for attack detection as discussed in However, the difference is that we use only the selected features for each layer rather than using all the 41 features. And now give the algorithm for integrating CRFs with the Layered Approach.

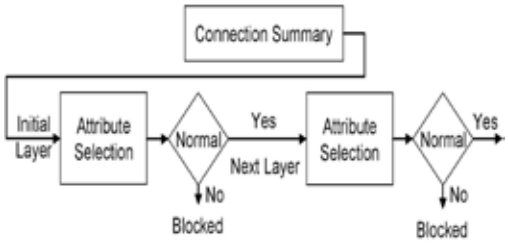

Fig.3: Representation of Layered structure

We select features for each layer based upon the type of attacks that the layer is trained to detect as below.

1. 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.

2. DOS layer:
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" are significant. To detect DOS attacks, it may not be important to know whether a user is "logged in or not."

3. 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 attack.

4. U2R layer (User to Root attacks):
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, 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. DESIGNING & EXPERIMENTAL RESULTS
A use case diagram is a graph of actors, a set of use cases enclosed by a system boundary, communication (participation) associations between the actors and users and generalization among use cases. The use case model defines the outside (actors) and inside (use case) of the system's behavior.

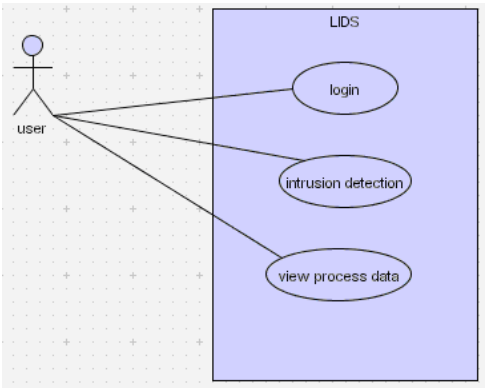

Fig.4: System Usecase representation

An Activity Diagram is a special case of state diagram. An activity diagram is like a flow chart, showing flow of control from activity to activity.

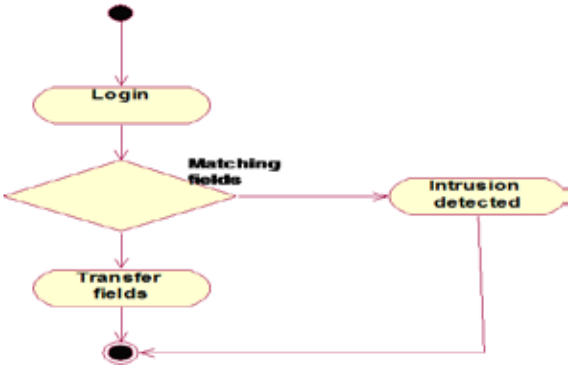

Fig.5: System Activity Representation

The use of benchmark KDD '99 intrusion data set. This data set is a version of the original 1998 DARPA intrusion detection evaluation program, which is prepared and managed by the MIT Lincoln Laboratory [9]. The data set contains about five million connection records as the training data and about two million connection records as the test data. In proposed system use of 10 percent of the total training data and 10 percent of the test data (with corrected labels), which are provided separately. This leads to 494,020 training and 311,029 test instances. Each record in the data set represents a connection between two IP addresses, starting and ending at some well-defined times with a well-defined protocol. Further, every record is represented by 41 different features. Each record represents a separate connection and is hence considered to be independent of any other record. The training data is either labeled as normal or as one of the 24 different kinds of attack. These 24 attacks can be grouped into four classes; Probing, DoS, R2L, and U2R.Similarly, the test data is also labeled as either normal or as one of the attacks belonging to the four attack groups. It is important to note that the test data is not from the same probability distribution as the training data, and it includes specific attack types not present in the training data. This makes the intrusion detection task more

Volume : 2 | Issue : 2 | Feb 2013 • ISSN No 2277 - 8179

realistic. Table 1gives the number of instances for each group of attack in the data set.

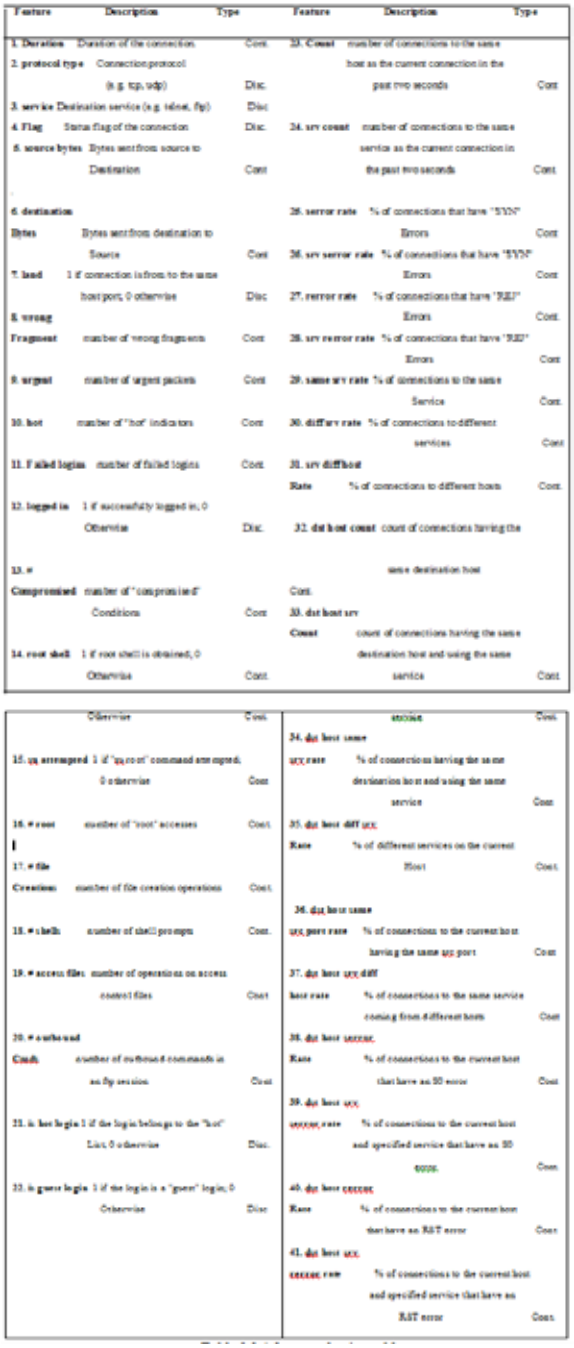

Fig.6: Future selection table

CONCLUSION

Based on the incongruity intrusion detection principle our system is compared with other well known methods. By considering normal mathematical method, data mining and machine learning approaches anomaly-based systems largely detect deviations from the learnt normal data. How ever decision trees and naive Bayes are well verse for their performance but the results of our approach shows that Layered CRFs performs better than those methods and the cause is CRFs do not consider the observation features to be independent. For our methods mathematical testing also provided higher confidence in detection accuracy. Thus we proved that our system is strong and can handle noisy free data providing with high performance

RESULTS COMPARISON

Techniques	Data	Probe (%)	Dos (%)	R2L (%)	U2R (%)
Layered conditional random fields	PD(propablistics dection FAR (false alarm rate)	98.60 0.91	97.40 0.07	29.600 0.350	56.300 0.0500
KDD'99 winner	PD FAR	83.30 0.60	97.10 0.30	8.400 0.005	13.2000 0.0030
Multi-layer perceptron	PD FAR	88.70 0.40	97.20 0.30	5.600 0.010	13.2000 0.0500
Gaussian Classifier	PD FAR	90.20 11.30	82.40 0.90	9.600 0.100	22.8000 0.5000
K-Means clustering	PD FAR	87.60 2.60	97.30 0.40	6.400 0.100	29.8000 0.0006
Leader Algorithm	PD FAR	83.80 0.30	97.20 0.30	1.000 0.003	6.6000 0.0300
Fuzzy ARTMAP	PD FAR	77.20 0.20	97.00 0.30	3.700 0.004	6.1000 0.0010
C4.5(Decision trees)	PD FAR	80.80 0.70	97.00 0.30	4.600 0.005	1.8000 0.0020
Decision tree with principle component analysis	PD FAR	70.40 0.85	97.38 0.12	0.070 0.030	7.0200 0.0001

Fig.7: Result Comparison

REFERENCE

[1] A.K. McCallum, MALLET: A Machine Learning for Language Toolkit, http://mallet.cs.umass.edu, 2010. | [2] L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion Detection with Unlabeled Data Using Clustering," Proc. ACM Workshop Data Mining Applied to Security (DMSA), 2001. | [3] 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. | [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] H. Shah, J. Undercoffer, and A. Joshi, "Fuzzy Clustering for Intrusion Detection," Proc. 12th IEEE Int'l Conf. Fuzzy Systems (FUZZ-IEEE '03), vol. 2, pp. 1274-1278, 2003. | [6] C. Sutton and A. McCallum, "An Introduction to Conditional Random Fields for Relational Learning," Introduction to Statistical Relational Learning, 2006. | [7] R. Bace and P. Mell, Intrusion Detection Systems, Computer Security Division, Information Technology Laboratory, Nat'l Inst. of Standards and Technology, 2001. | [8] D. Boughaci, H. Drias, A. Bendib, Y. Bouznit, and B. Benhamou, "Distributed Intrusion Detection Framework Based on Mobile Agents," Proc. Int'l Conf. Dependability of Computer Systems (DepCoS-RELCOMEX '06), pp. 248-255, 2006. | [9] S. Forrest, S.A. Hofmeyr, A. Somayaji, and T.A. Longstaff, "A Sense of Self for Unix Processes," Proc. IEEE Symp. Research in Security and Privacy (RSP '96), pp. 120-128, 1996. | [10] Y. Gu, A. McCallum, and D. Towsley, "Detecting Anomalies in Network Traffic Using Maximum Entropy Estimation," Proc. Internet Measurement Conf. (IMC '05), pp. 345-350, USENIX Assoc., 2005.