CLASSIFICATION TREE AND KULLBACK-LEIBLER DISTANCE-BASED ANOMALY INTRUSION DETECTION APPROACH

Evgeniya P. Nikolova, Veselina G. Jecheva

Abstract: *In recent years anomaly detection has become an important area for both commercial interests as well as academic research. The intrusion detection process attempts to detect malicious attacks by examining various data collected during processes on the protected system. The present paper proposed an adaptive approach of anomaly based intrusion detection which is grounded on classification trees and relative entropy. The major results of the implemented simulation experiments are presented and discussed as well.*

Keywords*: Intrusion Detection, Anomaly Based IDS, Classification Trees,Relative entropy*

I. Introduction

The importance of information security and intrusion detection, in particular, has been growing over the past few decades, since more and more companies and organizations highly rely on the network communications. As reported by the Computer Emergency Response Team/Coordination Center (CERT/CC) [3], the number of computer attacks has increased exponentially in the past few years. It is the prevalence of such threats that has made intrusion detection systems-the cyberspace's equivalent to the burglar alarm-join ranks with firewalls as one of the fundamental technologies for network security [15]. The purpose of intrusion detection is to detect any illegal activity that happens at the organization’s network. For this reason the intrusion detection system (IDS) monitors the system activities and collects and analyzes data, which describes the users’ behaviour. There are two major types of detection methods: misuse detection, which relies on the knowledge of signatures of known attacks and system vulnerabilities exploits, and anomaly detection, which searches for illegal user activity. The anomaly detection methodology has the advantage that it can detect novel and unfamiliar attacks, since it relies on preliminarily defined profiles of normal user data and looks for significant deviations in the real-time user activity. The actions that divert too much from the profiles of normal user activities are marked as intrusions. Many contemporary IDSs integrate both approaches to benefit from their respective advantages [13].

Anomaly detection methods rely on the assumption that all intrusive activities cause some anomalies in the system data [12]. There are many anomaly detection methods described and applied that differ according to the analyzed data and methods that are enforced to detect deviations from normal behavior: statistical analysis [5], rule-based methods [1, 19], profiling methods [14], etc.

One of the major difficulties in creating an anomaly detector is to select and create the model of normal system behavior. Other issues in the anomaly IDS creation include the proper selection of threshold levels the IDS uses to trigger an alarm, the selection of features to monitor, the inability to train a general model, which could be applied for all systems, the necessity of training and adjustion of the model according to the specific system, which has to be protected, etc.

Among the most successful approaches is the concept of sequence sets of system call patterns to describe the normal system activity and deviations from this baseline [9], [4], [16]). The aim of the proposed method is to analyze the program behavior not the user profiles ([11], [7]). The program behavior monitoring is performed by capturing system calls made by some privileged processes under normal operational conditions. This approach relies on the fact that short sequences of system calls are a reliable discriminator between normal and malicious activity.

The present paper proposes an anomaly detection approach, based on the immune systems, which first collect data patterns representing the appropriate behavior of a service, running on a server, which is possible target of the attackers. Then these systems extract a reference table containing all the known good sequences of system calls. The proposed approach creates as a first step, a classification tree, which contains all data representing legal user activities. The system protection includes monitoring of the sequences of system calls of current system activities and their comparison with those in the database. When the system finds a significant deviation from the preliminarily composed tree, it raises an alarm signal and marks the sequence as anomalous.

II. Description of the Methodology

1. Relative entropy

The relative entropy [8] is a measure of the distance between two distributions and can be used as an indicator to measure the distance between two data points. Thereby we can use it as a measure of the regularity

of tasting data in IDS. The relative entropy or Kullback Leibler distance between two probability mass functions *p(x)* and *q(x)* is defined as follows

$$D(p \parallel q) = \sum_{x \in \mathrm{X}} p(x) \log \frac{p(x)}{q(x)} = E_p \log \frac{p(\mathrm{X})}{q(\mathrm{X})}.$$

with the condition that $0 \log \frac{0}{q} = 0$ and $p \log \frac{p}{0} = \infty$.

The smaller the relative entropy, the closer the two sources are in terms of their probability distributions.

2. The Applied Methodology

The classification tree is a data mining technique for predicting the membership of cases in classes defined by a dependent variable usually of the categorical type [2]. Each case is measured along a number of predictor variables. The implementation of a classification tree is achieved through a training process (induction) in which a specific algorithm is applied to a sample dataset (a training set) composed of the predictor variables [10].

In our case the normal activity patterns compose a set *Q* with *N* states: q_1, q_2, ..., q_N, which the system passes through its work in the discrete moments of time *t=1, ..., T*. We calculated the elements of the matrix, containing the state transition probabilities, assuming the probability of occupying a state is determined only by the preceding state. Each state transition probability represents the probability of transitioning from a given state to another possible state.

First we construct classification trees of level *L*, describing normal activity. The roots represent each possible state q_k. Inheritors for each vertex are the states for which the corresponding transition probabilities from their predecessor are non-zero

So by traversing the tree from the root to the leaves we can receive all possible state sequences with length *L* along with the corresponding transition probabilities. The obtained lists of system calls consist of all possible sequences with given state in kth position and contain states for which the transition probabilities for each couple of neighbors is non-zero.

Within the created classification trees we apply relative entropy as an indicator to measure the distance between the received sequence and normal sequences for the number of errors calculation. This produces a quantitative value that describes the distortion of the distribution of set of observations from that of the baseline distribution, and this is used as an indication of anomalies.

III. Simulation Experiments

1. Results of the Experiments

A software prototype, based on the described methodology, was developed and a number of simulation experiments were conducted in order to evaluate the implementation of the proposed method. The experimental data were obtained from a project performed by the researches in the Computer Science Department, University of New Mexico [18]. The data are obtained from Unix system examination during some period of time and consist of normal user activity patterns of some privileged processes executed on behalf of the root account as well as some anomalous data. The methods for pattern generation are described in [6]. The input data files consist of sequences of ordered pairs of numbers, where the first number is the process ID (PID) of the executed process and the second one is the system call number. In our experiments we examined data about the processes synthetic sendmail, inetd, login, and ps. The distance distributions for the process synthetic sendmail, named, login and inetd, for *L*=7 are represented in Figure 1:

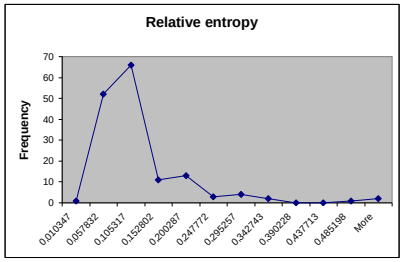

Figure 1. Distance distribution

In figure 1 can be observed that the most obtained values of the relative entropy are between 0 and 0.16, and the number of values, which are greater than 0.25, decrease significantly. These results suggest that such values correspond to call sequences, which can be marked as results of suspicious activity.

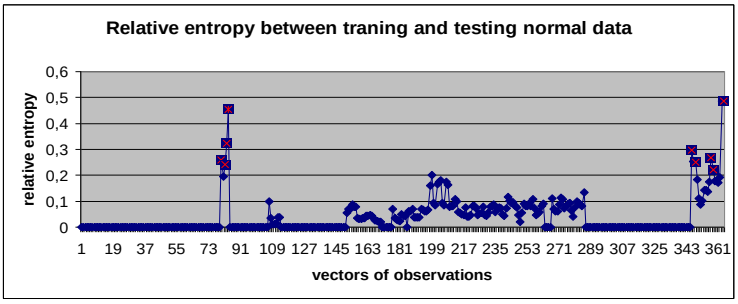

Figure 2. Relative entropy for the process synthetic sendmail

The relative entropy of the observations 80-84, 346-348, 357-358 and 364 increased considerably, as shown in Figure 2. The result relative entropy between training and testing normal data for the processes ps, login, named and inetd are represented in Figures 2. The presence of abnormalities was successfully detected by our relative entropy detection algorithm.

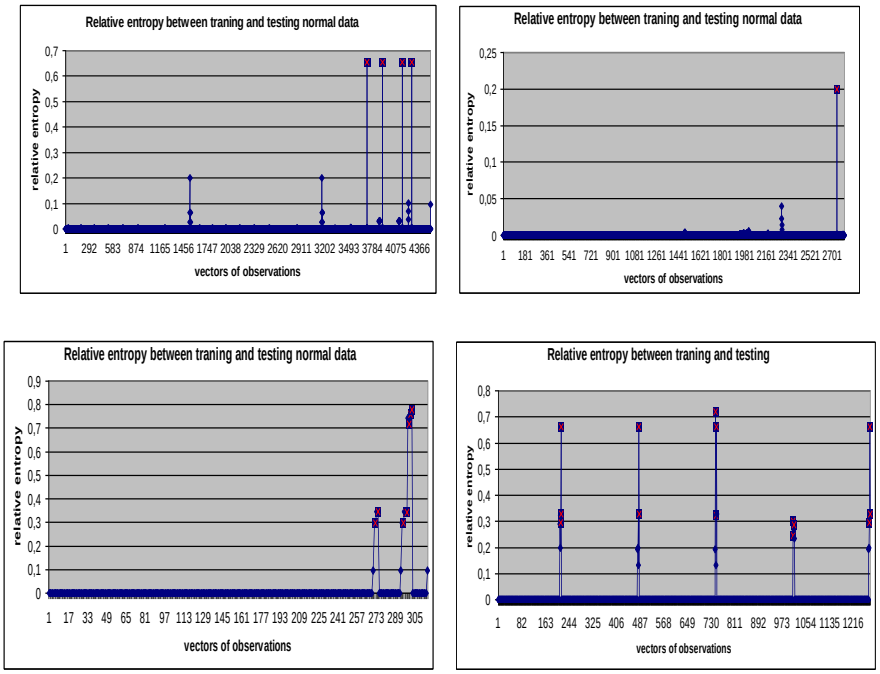

Figure 3. Relative entropy for the processes ps, login, named and inetd

2. Performance Evaluation

The goal of our classification test is to determine whether a given sequence belongs to one of two sets – normal set or intrusion set. For every possible test value there are two kinds of errors - *a false positive* (*FP*) and *a false negative* (*FN*). A false positive occurs when an event is predicted as intrusive but it is in fact normal. A false negative occurs when a truly intrusive event occurs without being signaled. True positive (*TP*) measures the proportion of actual positives which are correctly identified as such. True negative (*TN*) measures the proportion of negatives which are correctly identified as such.

The performance of each classifier was evaluated using the detection rate and overall accuracy. The detection rate shows the percentage of the true intrusions that have been successfully detected. It is a function of the identified intrusions:

$$Detection \quad rate = \frac{TP}{TP + FN} * 100$$

The overall accuracy [17] is calculated as the total number of correctly classified intrusions divided by the total number of observations

$$Overall \quad Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Processes	***Detection rate***	***Overall accuracy***
synthetic sendmail	0,8812	0,8447
ps	0,9836	0,9780
login	0,9647	0,9546
named	0,8608	0,8681
inetd	0,9728	0,9631

Table 1. Algorithm performance

Table 1 summarizes the algorithm performance in the experiments described above in terms of detection rate and overall accuracy. It can be noticed that, the detection rate for the process ps increases to 98,3% and the detection rate for the process inetd increases to 97,2%. The variance of the overall accuracy estimates are used as indicators of the methods benefits. The overall accuracy results are between 0,8447 and 0,9780, which suggest the number of false alarms is not significant, compared to the number of all predictions. The table shows that the method based on the relative entropy detects most of the anomalies, detected by allowing few false negatives and few false positives.

IV. Conclusion

This work represents an adaptive approach for anomaly-based intrusion detection using classification tree and relative entropy and related simulation experiments. The purpose of some future work could be the algorithm optimization and comparison with other detection methods.

References

1. Arjunwadkar M., R.V. Kulkarni, The Rule Based Intrusion Detection and Prevention Model for Biometric System, Journal of Emerging Trends in Computing and Information Sciences, VOL. 1, NO. 2, Oct.2010, pp.117-120.
2. Brieman, L., J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and Regression Trees. Belmont, CA: Wadsworth International Group, 1984.
3. http://www.cert.org
4. Dash S. K.; S. Rawat; A. K Pujari, LLE on System Calls for Host Based Intrusion Detection, Proceedings of the International Conference on Computational Intelligence and Security, 2006, pp. 609 – 612.
5. Evangelista P.F., COMPUTER INTRUSION DETECTION THROUGH STATISTICAL ANALYSIS AND PREDICTION MODELING, PhD Thesis, 2005.
6. Forrest S., S.A. Hofmeyr, A. Somayaji, T.A. Longtaff, *A Sense of Self for Unix Processes*, In Proceedings of the 1996 IEEE Symposium on Security and Privacy, IEEE Computer Society Press, Los Alamitors, CA, pp.120-128.
7. Ghosh A.K., A. Schwartzbard, M. Schatz, Learning Program Behavior Profiles for Intrusion Detection, In Proceedings of the 1st Workshop on Intrusion Detection and Network Monitoring, pp. 51–62, 1999.
8. Han, Te Sun & Kobayashi, Kingo (2002). Mathematics of Information and Coding. American Mathematical Society. pp. 19–20.
9. Haruyama T., H. Nakazato, H. Tominaga**,** Intrusion Detection by Monitoring System Calls with POSIX Capabilities, IEICE Transactions on Communications, Vol. E90-B, Num. 10, pp. 2646-2654, 2007.

10. Kokotos D.X., Y. G. Smirlis, A Classification Tree Application to Predict Total Ship Loss, Journal of Transportation and Statistics, Vol.8, Num. 2, 2005, pp. 31-42.
11. Lee W., S. J. Stolfo, K. W. Mok, Mining audit data to build intrusion detection models, In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD ’98), New York, NY, USA, 1998.
12. Leung K., C. Leckie, Unsupervised anomaly detection in network intrusion detection using clusters, In Proceedings of the Twenty-eighth Australasian conference on Computer Science - Volume 38, Newcastle, Australia, 2005, pp. 333 – 342.
13. Neumann P., P. Porras, Experience with Emerald to Date, In Proceedings of the First Usenix Workshop on Intrusion Detection and Network Monitoring, Santa Clara, CA, 1999.
14. Pannell, G., H.Ashman, Anomaly Detection over User Profiles for Intrusion Detection, Proceedings of the 8th Australian Information Security Management Conference, 2010, pp.81-94.
15. Patcha A.,J. Park, An overview of anomaly detection techniques: Existing solutions and latest technological trends, Computer Networks: The International Journal of Computer and Telecommunications Networking, Vol.51, Issue 12, August 2007, pp.3448-3470.
16. Rajagopalan M., M. A. Hiltunen, T. Jim, R. D. Schlichting, System Call Monitoring Using Authenticated System Calls, IEEE Transactions on Dependable and Secure Computing, Vol. 3, No. 3, 2006, pp. 216-229.
17. Taylor J. R., An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements. University Science Books, 1999, pp.128-129.
18. University of New Mexico’s Computer Immune Systems Project, http://www.cs.unm.edu/~immsec/systemcalls.htm.
19. Valdes A., Detecting Novel Scans Through Pattern Anomaly Detection, In *Proceedings of the Third DARPA Information Survivability Conference and Exposition (DISCEX-III 2003)*, Washington, D.C., April 2003.