The Design and Implementation of Host-based Intrusion Detection System

LIN Ying
School of Software
Yunnan University
Kunming,Yunnan Province, China
linying@ynu.edu.cn

ZHANG Yan
Computer Science Department
Southwest Forestry University
Kunming, Yunnan Province ,China
zydyr@163.com

OU Yang-Jia
School of Information Science and Engineering
Central South University
Changsha, Hunan Province, China
ouyangjia7@163.com

***Abstract*—**Intrusion detection is the process of identifying and responding to suspicious activities targeted at computing and communication resources, and it has become the mainstream of information assurance as the dramatic increase in the number of attacks. Intrusion detection system (IDS) monitors and collects data from a target system that should be protected, processes and correlates the gathered information, and initiates responses when evidence of an intrusion is detected. In this paper, we designed and implemented a host-based intrusion detection system, which combines two detection technologies, one is log file analysis technology and the other is BP neural network technology. Log file analysis is an approach of misuse detection, and BP neural network is an approach of anomaly detection. By combination of these two kinds of detection technologies, the HIDS that we have implemented can effectively improve the efficiency and accuracy of intrusion detection.

Keywords- intrusion detection; intrusion detection system; HIDS; Log analysis; BP neural network;OSSEC

I. INTRODUCTION

In 1980, James Anderson introduced the concept of Intrusion Detection [1], which defined an intrusion attempt or a threat to be the potential possibility of a deliberate unauthorized attempt to access information, to manipulate information, or to render a system unreliable or unusable. Since then, several techniques for detecting intrusions have been studied. In 1987, the first intrusion detection system model was studied out by Georgetown University Dorothy Denning and SR I / CSL's PeterNeumann[2].

An Intrusion Detection System(IDS) monitors and collects data from a target system that should be protected, processes and correlates the gathered information, and initiates responses when evidence of an intrusion is detected[3].Depending on their source of input, IDSs can be classified into Host-based Intrusion Detection System(HIDS), Network-based Intrusion Detection System(NIDS) and Hybrid Intrusion Detection System. Network-based intrusion detection system collects input data by monitoring network traffic. Host-based intrusion detection system collects input data from the host it monitors. Hybrid intrusion detection system collects input data from both of network traffic and hosts it monitors.

"Anomaly" detection and "Misuse" detection are two main techniques that HIDS use. Anomaly detection refers to intrusions that can be detected based on anomalous behavior and use of computer resources. Anomaly detection usually uses methods of statistic analysis methodology, artificial neural network technology, data mining technology, and artificial immune technology. Misuse intrusion detection refers to the detection of intrusions by precisely defining them ahead of time and watching for their occurrences[4]. Misuse intrusion detection usually use methods of expert system, TCP/IP protocol analysis, and pattern matching.

In this paper, we designed and implemented a host-based intrusion detection system, which uses pattern matching and BP neural network as its detection methods. Firstly, the HIDS uses log files as its primary sources of information, and through three steps of pre-decoding log file, decoding log file, and analysis log file, it can effectively identify various intrusions. Secondly, based on BP neural network analysis technology and through establishment of system behavior characteristics profile in advance, the HIDS can identify intrusions by comparison with threshold. Experiment results show that the HIDS can effectively improve the efficiency and accuracy of intrusion detection.

The rest of the paper is organized as follows. Section 2 describes log analysis technology. Section 3 describes BP neural network analysis technology. Section 4 describes the design and implementation of HIDS. Section 5 gives some screenshot of experiment and section 6 concludes.

II. LOG FILE ANALYSIS

Log files record the behavior of computer system and aim at recording the action of operating system, applications, and use behaviors. Log file is widely used for system debugging, monitoring, and security detection. Log system is particularly important in intrusion detection and log file analysis tool have become an indispensable tools for daily inspection and maintenance of the system running.

In general, log analysis-based HIDS includes the following several parts: collection of log file data, pre-decoding of log file, decoding of log file, analysis of log file and report events.

A. Collection of log file

The acquisition of host log file data mainly includes two categories: one is system-level logs, and the other is the application layer logs. You can use your own log tools or third party log tools to access log file. In short, in the collection phase, it is necessary to collect operational information as far comprehensive as possible.

B. Pre-decoding of log file

978-0-7695-4020-7/10 $26.00 © 2010 IEEE
DOI 10.1109/IITSI.2010.127

The purpose of the log file pre-decoding is to extract general information from the log. For example, suppose a new SSHD log produced a SSHD message:

```
Apr 14 17:32:06 linying sshd [1025]: Accepted password for root from
172.16.29.26 port 1618 ssh2
```

After pre-decoding the message, the date "Apr 14 17:32:06", the host name "linying", and the program name "sshd" are extracted. The extracted messages will be recorded as follows:

- Time/date->Apr 14 17:32:06
- Host name->linying
- Program name->SSHD

C. Decoding of log file

Log file decoding is the process to identify key information from logs. In the HIDS, we use regular expressions to identify certain keywords. For example, we still assume that SSHD log produced a SSHD message as above. After decoding this message, the content "accepted password for root from 172.16.29.26" , the source IP address "172.16.29.26" and the user name "root" are all extracted. The extracted messages will be recorded as follows:

- Source IP address->172.16.29.26
- User name->root
- Log->accepted password for root form 172.16.29.26

D. Analysis of log file

After the three stages of log collection, log pre-decoding and log decoding, all the contents are read into the rules tree. In this paper, we constructed the rules tree based on more than 400 rules of the OSSEC[5][6] .The general structure of the rule tree are shown in Figure 1.

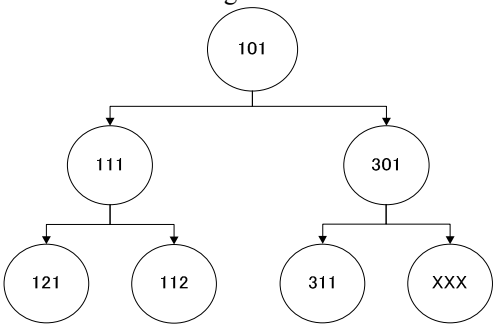

Figure 1. The rules tree.

After we got the decoded sequence of events, we will traverse the rules tree to do the matching process. For example, if we have the rules tree shown in Figure 1 and we got the event IF, then the matching process can be described as follows:

First of all, the event IF will be compared with 101 nodes, if the match is successful, enter 101 in the left node 111, else access to its right node 301. If found matching rules, then the first thing is to determine whether to do the ignore operations, if not, then perform the audit to effectively track the attacks. Then to determine what instructions should be implemented.

E. Report Events

After the process of log file analysis, report to the controller when necessary.

III. BP NEURAL NETWORK

Back propagation(BP) algorithm[7] is an approximate steepest descent algorithm, in which the performance index is mean square error. It can be used to train multilayer neural networks and it is used widely in practice. It is essentially a network of simple processing elements working together to produce a complex output. These elements or nodes are arranged into different layers: input, middle and output. The output from a back propagation neural network is computed using a procedure known as the forward pass[2][8][9].

The forward pass produces an output vector for a given input vector based on the current state of the network weights. Since the network weights are initialized to random values, it is unlikely that reasonable outputs will result before training. The weights are adjusted to reduce the error by propagating.

The output error is backward through the network. This process is where the back propagation neural network gets its name and is known as the backward pass.

The training set is repeatedly presented to the network and the weight values are adjusted until the overall error is below a predetermined tolerance. Since the Delta rule follows the path of greatest decent along the error surface, local minima can impede training. The momentum term compensates for this problem to some degree.

This paper uses the back propagation algorithm to train the multi-layer neural network in order to detect the anomaly intrusions. There are many measures that can be used to be input value of BP network algorithm. The following table illustrates some intrusion detection measures that can be used as the input value of BP network algorithm.

TABLE I. SOME MEASURES THAT INTRUSION DETECTION CAN USE

Login and session activity
● Login frequency ● Login frequency at different positions ● Time consumed by each session ● Website output
Resources utilization
● Password failed times when login ● The implementation of commands and procedures ● Operating frequency ● Utilization of procedure resources
File operating activity
● The frequency of file read, write, create, and delete ● Records read and write ● Read, write, create and delete file

We can use these measures as input value of BP neural network algorithm, through adjust the network parameters to minimize the mean square error, and finally establish characteristic profile in advance. The training phase may take days or weeks of computer time. This has encouraged

considerable research on methods to accelerate the convergence of algorithm.

IV. IMPLEMENTATION

As discussed above, the HIDS combines two approaches of misuse detection and anomaly detection. The structure of the whole system is described as the figure 2:

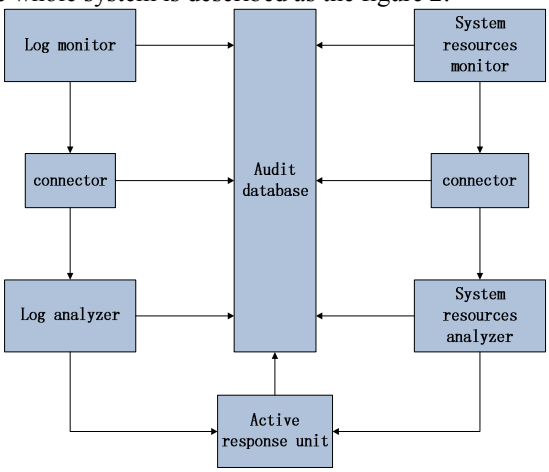

Figure 2. The structure of the HIDS.

1) Log monitor

Monitoring the log file, once the log change, log monitor will send events to the log analyzer immediately.

Generally, we need to monitor three kinds of event logs: application log, security log and system log. We can add three XML nodes in the following configuration file.

```
<localfile>
  <location>Application</location>
  <log_format>eventlog</log_format>
</localfile>
<localfile>
  <location>Security</location>
  <log_format>eventlog</log_format>
</localfile>
<localfile>
  <location>System</location>
  <log_format>eventlog</log_format>
</localfile>
```

The node “localfile” represents the local file when system initialization. The node “location” represents file path in the disk. The node “log_format” represents what type of the log. Log type includes event log, firewall log, SQL log and so on.

For example, if you want to add a firewall log to the configuration file, you just need to write as follows:

```
<localfile>
  <location>C:/WINDOWS/pfirewall.log</location>
  <log_format>syslog</log_format>
</localfile>
```

In this way, when initialize the HIDS, it will automatically load the above log files that need to be monitored. When finished the initialization work, the HIDS will open a demon, and the demon will check every log files to find whether there is changes in the log file. If there really exits a change, then the demon will report to the log analyzer.

2) System resources monitor

Monitoring the use of system resources, and sends the status of the system resources utilization to the system resources analyzer at regular time.

3) Connector

The connector is responsible for receiving messages from log monitor and system resources monitor, and sending these messages to log analyzer and system resources analyzer.

4) Log analyzer

Receiving events from the log monitor, match with the rule base to determine whether there is invasion, if there is invasion occurrence, report to the active response unit.

5) System resources analyzer

Receiving events form the system resources monitor, to calculate whether the abnormal state of current resources use and thus to determine whether the status is invaded, if it find there is invasion, report to the active response unit.

6) Active response unit

Receiving events from the log analyzer and system resources analyzer, decided to perform what kind of operation. Usually, the normal operations include notifying users, auditing, disconnecting from network and so on.

7) Audit database

Recording the entire process of intrusion detection, and the attack situation, prepare for use when necessary.

V. RESULT OF THE EXPERIMENT

Figure 3 is the screenshots that illustrates intrusions that were detected by log analysis technology.

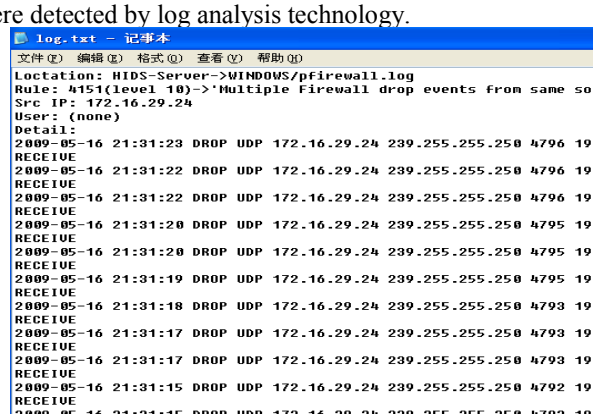

Figure 3. The intrusion information detected by log analysis.

Figure 4 is the screenshots that illustrates the training results when we used CPU utilization as the input values of BP neural network algorithm.

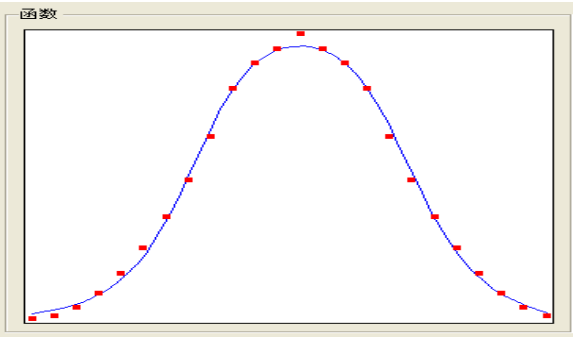

Figure 4. The profile of CPU utilization.

If we use the CPU utilization in one moment as the input value to the trained BP neural network. The BP neural network will calculate the output value, if the output value equals to 0, then it is normal, else if the output value equals to 1, then it is abnormal. Figure 5 is the screenshot of the HIDS detected abnormal CPU utilization.

Figure 5. The CPU utilization is abnormal.

VI. CONCLUSIONS

We used two kinds of techniques in the HIDS. One is anomaly detection technology, and the other is pattern matching detection technology. The advantage of our system lies in:

1) By combining two approaches in the HIDS, these two detection technology can complement each other, which can effectively approve the efficiency and accuracy of intrusion detection.

2) The HIDS can be gradually trained by various input value, and the administrator can set the threshold to prevent it is too low or too high.

3) Based on the technology of OSSEC, the HIDS can monitor various log file, such as firewall log, router log, web server log, and so on, which greatly improve the compatibility of the HIDS.

REFERENCES

[1] J.P Anderson, "Computer Security Threat Monitoring and Surveillance", Technical report, James P Anderson Co., Fort Washington, Pennsylvania, April 1980.

[2] Dorothy Denning, "An Intrusion Detection Model", IEEE Transactions on Software Engineering, February 1987, pp.2- 222.

[3] G. Vigna and C. Kruegel, "Host-based Intrusion Detection Systems," in The Handbook of Information Security, Volume III, John Wiley & Sons, December 2005.

[4] Sandeep Kumar, Eugene H. Spaffor, "An application of Pattern Matching in Intrusion Detection", Technical report 94-013,Purdue University, Department of computer sciences, March 1994.

[5] Daniel B. Cid, OSSEC[OL] , 2008, http://www.ossec.net.

[6] Andrew Hay,Daniel Cid, Rory Bray, Log Analysis using OSSEC[M], Syngress, 2007.

[7] Russell, S. and P. Norvig, 2003, Artificial Intelligence: A Modern Approach[M], 2nd Edn, Prentice Hall, Inc.

[8] Yen, J.C. and J.I. Guo, 2002, "The design and realization of a chaotic neural signal security system", Pattern Recognition and Image Analysis (Advances in Mathematical Theory and Applications), 12,pp. 70-79.

[9] Lian, S., G. Chen, A. Cheung and Z. Wang, 2004. A chaotic-neural-network-based encryption algorithm for JPEG2000 encoded images.Advances in Neural Networks, Intl. Symp. Neural Networks Proc., Part II, Lecture Notes in Computer Science, 3174,pp.627-632.