Detecting Internal Intrusion and Protecting System Using Data Mining and Forensic Techniques

Pujitha. CH[1], Sainath. N[2], Himaja. N[3], Pavani. D[4], Sai teja. D[5]
[2]Associate Professor, [1,3,4,5] Student,
[1]Computer Science and Engineering,
[1]St. Martin's Engineering College, Hyderabad, India.

Abstract **-** Computer domain consists one of the serious threat that is internal intrusion. Internal Intrusion considered as wrongful entry in the software applications. Today's computer system uses username and password for authentication. Log patterns are maintained to store the username and password. The security arouse when user shares log patterns with collaborators. For security purpose internal intrusion detecting and protecting system (IIDPS) is introduced to detect internal intrusion. In addition, System calls (SC) are analyzed to identify the malicious behavior. Malicious behavior may take place by different attacks. It is known that most firewalls protect against outside attackers. IIDPS is combination of data mining and forensic techniques which is used to detect internal attack. This paper is mainly focused on detection of internal intrusion by data mining and forensic techniques.

Index Terms**: Data mining, log patterns, Forensic techniques, System calls**

I. INTRODUCTION

Network based information is accessed by everyone.The inside and outside attackers penetrate into the computer system via network. Outside attackers are provided with the more privileges in the network then the insiders. Outside attackers access the unauthorized profile intentionally or unintentionally. While, Insider concentrates on the information which is secured with the integrity, confidentiality. Different approaches are introduced like encryption and firewall but they are not enough for full security network

Thus, (IIDPS) internal intrusion detection and protecting system is proposed in this paper to detect the malicious behavior .System calls are maintained at the same level. This security system uses data mining techniques to study the system calls pattern, consists of lengthy sequence of repeated commands in the user profile. Forensic features, defined as a SC-patterns frequently appearing in a user submitted SC-sequence yet rarely used by others which are extracted from computer usage history.

The benefaction to this paper are Firstly, to elevate the accuracy of attack detection we need to identify a user forensic features of corresponding SCs by analyzing. Secondly, Withstand of inside attacker .and finally to shorten the response time we need to port the IIDPS to the parallel system

II. LITERATURE SURVEY

The word forensics means of or belonging to law, and forensic techniques are used to solve crimes. This technique addresses the crime scenes in the computer system. The crime in this paper is related to the attackers. It identifies stores, recovers and analysis the information which is collected for a security goal. It analysis the features of the attackers in which they have done like virus spreading, Malicious behavior, denial-of-service (DoS). Distributed denial-of-service (DDoS) and denial-of-service (DoS) are the most dreadful network threats to computer system. An internal intrusion and protecting system is used to identify the malicious behavior. The main aim of internal intrusion is to detect the attacker on network based on the behavior, attackers patterns are stored and compared to the log patterns in the computer history. Firstly, author used self-packet to compare it to the packets in the history so as to detect the attackers in the network. Log file is maintained for network intrusion and attacks pattern. File consists of details of the misuse of the computer. It can be said that by log files the path and information of malicious behavior can be traced. The existing research challenge describes us about the comparison of intrusion systems and to summarize the details. Author overviewed this research by applying different techniques to detect malicious behavior .This techniques works finely with the network security. In previous research was developed by data mining and forensic techniques, command level can be used rather than SC-level. Moreover, when attacker try to attack with multiple attacks then it is not easy to detect the attacker or type of the attack .To ensure this, author came forward with internal intrusion detecting and protection system using data mining and forensic technique. Forensic technique is used to identify the behavior and data mining technique is used to analyze the type of the attacker. However, SC-filters are not mentioned in this work. Author came with another research with the system calls and to execute the sequence of system calls. Anomaly is detected when execution is done. In addition, K.A Garcia, R. Monroy, L.A Trejo and C. Mex-Perera worked on analyzing

log files for postmortem intrusion detection. Attack signatures are maintained for detecting internal intrusion. Fang-Yie Leu and his team members worked on the internal intrusion detection and preventing system using data mining and forensic techniques. In this technique log patterns are preserved to compare it to the computer usage history. Log patterns consist of sequence of system calls (SC). By data mining and forensic techniques systems calls are analyzed and compared .By comparing if any missing of sequence arouse then it tends to malicious behavior .for identifying this behavior IIDPS is used.

2.1 ADVANTAGES

- Confidential information is secured against network hacking by IIDPS system.
- Data analysis in IIDPS system consumes shorter time when compared to other systems.
- Detecting of malicious behavior of user is efficient than existing system
- Monitors both internal and external attacks.
- Accuracy of attack detection is higher.
- Regular network monitoring is done when user is away from computer.

III. OVERVIEW OF THE SYSTEMS

System framework is a compound of a mining server, detection server, local computational grid and system call monitor and filter. It also have three repository systems which are user log file, user profile and an attacker profile.

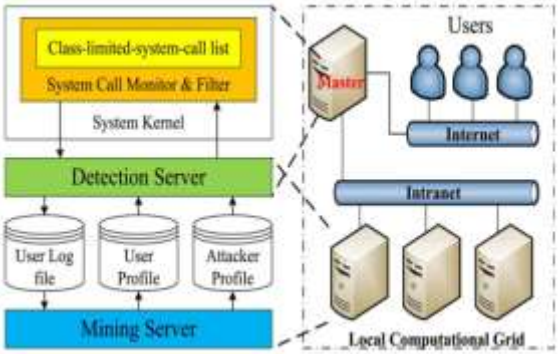

Fig. 1. IIDPS System framework

3.1 Local Computational Grid

Local computational grid represents that the resources from the multiple locations share a task and run it in parallel. The mining server and detection server are placed in local computational grid to increase the IIDS's online detection and mining speeds and to improve its detection and mining.

3.2 Mining Server

Mining server analyzes the users' habits and stores them in user habit file. This was done with the help of data mining techniques. After this it filters the SC patterns that are commonly used by most users and compares the result with other users' habit file to obtain a specific SC-patterns.

3.3 Detection Server

Detection server tries to identify whether the user is the original account holder or not by comparing the user current usage habits with the stored user profile. If it doesn't match server informs the system call monitor and filter to isolate the user from the system.

IV. METHODOLOGY

4.1 System Call Monitor and Filter

A system call provides an interface to the operating system services. The system call monitor and filter is entrenched in the kernel of the system. It collects the system calls and stores them in the form of user id, process id and system call (UID, PID, SC). As there will be SCs generated for every activity, the amount of the SCs will be high. So, the SC monitor and filter, filters the system calls that are generated by shell commands. The term frequency – inverse document frequency model is used for filtering.

$$TF_{i,j} = \frac{n_{i,j}}{\sum_{k=1}^{h} n_{k,j}} \qquad (1)$$

Where $n_{i,j}$ is the number of times that term t_i is present during the execution of j, h is the number of different SCs generated when j is executed

$$\mathrm{IDF}_i = \log \frac{|D|}{|\{j : t_i \in d_j\}|} \quad (2)$$

Where $|D|$, is the total number of shell commands in the concerned corpus and $\{j: t_i \in d_j\}$ is the set of shell commands d_j, in which each member generates t_i during its execution. The TF-IDF weight of t_i generated by j is defined as

$$(\text{TF-IDF})_{i,j} = \mathrm{TF}_{i,j} \times \mathrm{IDF}_i. \quad (3)$$

4.2 Mining Server

There are two steps in this Server

1) User and attacker habits mining.

2) User and attacker profile creation.

Algorithm 1: The algorithm for generating a user habit file
Input: u's log file where u is a user of the underlying system
Output: u's habit file

```
1.  G = |log file| - |Sliding window| ;
    /* |Sliding windows|=|L-window|=|C-window| */
2.  for ( i=0 ; i ≤ G−1 ; i++) {
3.    for ( j=i+1 ; j ≤ G ; j++) {
4.      for (each of ∑_{k=2}^{|Sliding window|}(|Sliding window| − k + 1) k-grams in
          current L-window){
5.        for (each of ∑_{k'=2}^{|Sliding window|}(|Sliding window| − k' + 1) k'-grams
          in C-window){
6.          Compare the k-grams and k'-grams with the longest common
            subsequence algorithm;
7.          if (the identified SC-pattern already exists in the habit file)
8.            Increase the count of the SC-pattern by one;
9.          else
10.           Insert the SC-pattern into the habit file with count=1; }}}}
```

Fig. 2 Algorithm for generating user habit file

4.3 Detection Server

Detection server tries to identify whether the user is the original account holder or not by comparing the user current usage habits with the stored user profile. If it doesn't match server informs the system call monitor and filter to isolate the user from the system.

Algorithm 2: Detecting an internal intruder or an attacker
Input: user u's current input SCs, i.e., NSC_u (each time only one SC is input), and all users' user profiles
Output: u is suspected as an internal intruder or a known attacker

```
1.  NCS_u = ∅;
2.  while ( receiving u's input SC, denoted by h ) {
3.    NCS_u = NCS_u ∪ {h};
4.    if ( |NCS_u| > |Sliding window| ) {
5.      L-window = Right(NCS_u, | Sliding window|); /* Right(x, y) retrieves
                the last L-window of y from x */
6.      for ( j= |NCS_u| - |Sliding window| ; j >0 ; j-- ) {
7.        C-window = Mid (NCS_u, j, |Sliding window|);/* Mid (x, y, z) retrieves
            a sliding window of size z beginning at the position of y from x */
8.        Compare k-grams and k'-grams by using the comparison logic
            employed in Algorithm 1 to generate NHF_u;}
9.      for (each user g, 1≤g≤N)
10.       Calculate the similarity score Sim(u, j) between NCS_u and g's user
            profile by invoking Eq. (8);
11.     if ( ( |NCS_u| mod paragraph size) == 0) {  /* paragraph size = 30,
            meaning we judge whether u is an attacker or the account holder for
            every 30 input SCs */
12.       Sort similarity scores for all users;
13.       if ((the decisive rate of u's user profile < threshold_1) or
            (the decisive rate of attacker profile > threshold_2)){
              /* threshold_1 is the predefined lower bound of average decisive
              rate of user u's user profile, while threshold_2 is the predefined
              upper bound of average decisive rate of attacker profile*/
14.       Alert system manager that u is a suspected attacker, rather than u
            himself/herself; }}}}
```

Fig. 3 Algorithm for detecting whether the user is possibly an internal intruder

V. RESULTS

The main objective is to detect the intruder by correlating the activity, he has done after gaining access to the system, with the precedent activities stored in the log files.

All the activities of the user are monitored and are recorded in the log file. This system identifies the behavior of an intruder which will be significantly different from that of legitimate user. For this process we use data mining and forensic techniques, which analyses the log files and compare them with the attacker's activity.

There are two parts in this procedure they are creation of user's personal profile and a mechanism for detection. The user profile is created in mining server and the detection of the attack is done in detection server. The following are the few test results.

Fig. 4 Upload training signature then generate the SVM model for the given data

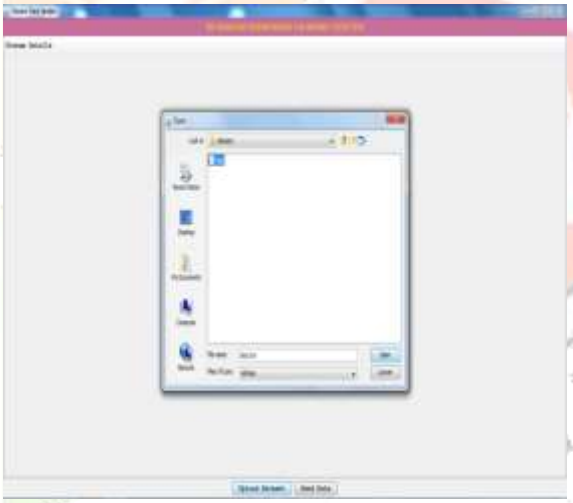

Fig 5. Upload dataset then send data set

Fig 6. View detections

VI. CONCLUSION

This paper summarizes regarding the techniques of data mining and forensic. IIDPS retains both techniques for identification of malicious behavior. This techniques analyzes system calls which are reserved in the form of log patterns and isolates the user when the SC patterns doesn't match.

VII. REFERENCES

[1] H. S. Kang and S. R. Kim, "A new logging-based IP traceback approach using data mining techniques," *J. Internet Serv. Inf. Security*, vol. 3, no. 3/4, pp. 72–80, Nov. 2013.

[2] K. A. Garcia, R. Monroy, L. A. Trejo, and C. Mex-Perera, "Analyzing log files for postmortem intrusion detection," *IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev.*, vol. 42, no. 6, pp. 1690–1704, Nov. 2012.

[3] M. A. Qadeer, M. Zahid, A. Iqbal, and M. R. Siddiqui, "Network traffic analysis and intrusion detection using packet sniffer," in *Proc. Int. Conf. Commun. Softw. Netw.*, Singapore, 2010, pp. 313–317.

[4] Fang-YieLeu, Kun-Lin Tsai, Yi-Ting Hsiao, and ChaoTung Yang,'' An Internal Intrusion Detection and Protection System by Using Data Mining and Forensic Techniques",IEEE Int. Conf. Avail., Rel. Security, Taiwan,pp 1932-8184,2015

[5] S. C. Arseni, E. C. Popovici, L. A. Stancu, O. G. Guta, and S. V. Halunga,"Securing an alerting subsystem for a keystroke-based user identification system," in Proc. Int. Conf. Commun., Bucharest, Romania, 2014,pp. 1–4.

[6] Ya-Ting Fan1 and Shiuh-Jeng Wang, "Intrusion Investigations with Data-hiding for Computer Log-file Forensics", IEEE 2010.

[7] R. Araeteh, M. Debbabi, A. Sakha, and M. Saleh, "Analyzing multiple logs for forensic evidence," Digital Investigation 4S, pp, 82- 91, 2007.

[8] Karen Scarfone& Peter Mell, National Institute of Standards and Technology (NIST) Special Publication 800-94 , " Guide to Intrusion Detection and Prevention Systems", Feb 2007.