IDS: Usage as Honeypot Technology

[1]Brijindra Pal Singh, [2]Ankita Pandey, [3]Sarbjeet Singh
[1]Dept. of Computer Science, NITTTR – Panjab University, Chandigarh, UT, India
[2,3]Dept. of Electronics and Communication NITTTR- Panjab University, Chandigarh, UT, India

Abstract
This paper moves toward the Honeypot technology. Paper presents an IDS module based on honeypot technology, which uses PID track technique. With using the honeypot system, this component has the capability of attacker detection and response; the whole detection component can be extended with ease and be configured easily and flexibly. By using honeypot technology, this module findout the intrusion source farthest.

Keywords
Honeypot Technology, Intrusion Detection, IP Trace Back

I. Introduction

With the quick growth of Internet, various systems for network attacks increasing, more and more complex means of destruction so, the rising problem of network information security. Intrusion Detection System (IDS) as an active protection strategy, intrusion detection and prevention of other safety modules play a unique role. The honeypot technology as a strong balance to IDS can greatly reduce the load of IDS, while the greatest degree of access to information the invader in order to make easy further tracking the attack source. In this paper, we discuss the traditional intrusion detection system is addressed based on honeypot technology, IDS can be traced back through the Honeypot system to achieve interoperability between the various components, use of honeypot technology to get the maximum extent potential attack information in order to make easy further analysis of the source tracking, and signature data to achieve timely and automatic updates.

II. Traditional Intrusion Detection Systems And Their Limitations

Intrusion detection systems can be divided into anomaly detection systems and misuse detection systems into two categories according to different methods of intrusion detection. Misuse detection systems can only detect known attack patterns characteristic, characteristic patterns of unknown attacks can not be detected. The anomaly detection system uses the system's current activities and past behavioral models to compare the methods that can effectively new, unknown attacks detection. As the network continues to expand, increasingly complex [1].

Traditional intrusion detection systems when in use revealed the following deficiencies:

- Misuse detection and intrusion detection technology features related with the diversification of network attacks and new attacks continue to emerge, there are inevitably omissions. In addition, the intrusion detection system using the right pattern-matching algorithm to match the data to detect the presence of invasion. Once a packet matches the content and signature, immediately issued a warning not to judge it is not really attack, so there could be false positives.
- Signature database updates are very difficult. Most traditional intrusion detection systems use pattern matching analysis, which requires the Eigen values of attack signature database should be up to date. However, the existing intrusion detection system does not always provide a good way to update the signatures.
- Method of attack is more complex, a single based pattern matching or statistical methods of analysis have been difficult to find a number of attacks.
- The existing intrusion detection systems can not exchange information, makes it difficult to find the attack the source of the attack, and even to the intruder has created vulnerabilities. Existing intrusion detection systems and other network security products can not interoperate.

Information encryption, attacker increasing number of widely used mobile code (Java, ActiveX, etc.), false alarms, are traditional intrusion detection systems are facing great challenges.

III. Intrusion Detection in Honeypot Technology

Honeypot is a decoy system includes vulnerabilities by simulating one or more vulnerable hosts, the attacker provide an easy target. It is designed to attract and "decoy" who designed the attacker. Honeypot meaning of existence is to be detected, was offensive. If there are no attacks, honeypots will become meaningless. And firewall technology [5], virus protection, data encryption and authentication technologies such as passive protection technologies is best approach towards honeypots. It uses the unique characteristics to attract an attacker to lure attackers to attack its host system, while monitoring the system operation and behavior of all, and the formation of these acts recorded log. Through the log of research, analysis of attackers to the path, use the tools, tactics and purpose, can be more effective intrusion source tracing, but also can conduct real-time network intrusion forensics. Honeypot intruder obtained by information security experts can make a better understanding of the various attacks; security experts to provide a learning platform for all kinds of attacks and better protect the system should be protected. Honeypot intrusion detection system technology is a strong complement. Share a part of its data traffic and simplify the detection process, reducing the burden of intrusion detection systems. When the signature line of events and unknown events after the introduction of the honeypot, the honeypot to monitor visitors to access the action, and then determine whether the incident is unknown attacks. Determine if the aggressive behavior, while recording the attack, on the one hand to provide signatures to the signatures, the intrusion detection system in time to learn new attacks.

Honeypot and intrusion detection system not only reduces the combination of traditional intrusion detection system, false negative rate and false alarm rate, but also to overcome the traditional intrusion detection system can not monitor defect unknown attacks.

IV. System Structure And Main Functions

Fig. 1 shows the honeypot system model, - honeywall [5], proxy server, the honey pot client module, database, and alert generator features five major rules. Among them, the data capturing module, data alert generator module and response module static Agent and Mobile Agent using a combination of ways. They can be run in a protected host, can also be run in a protected network, the specific location of the key nodes in the network can also be a network of border control node. All connections into the honeypot detection module to redirect all the data from. When they find an

intruder, immediately call the response module, block the intrusion and alarm; Failure to detect intrusions, network packets may be normal access to the real environment; if you can not determine whether intrusion is considered suspicious behavior, the need to re-directed to the honeypot system. For suspicious intruders with full interaction honeypot to collect intrusion information, observe the intrusion, in order to further track the intruder.

A. Data Capturing Module

Data Capturing module's main function is to collect system logs, audit data and network data packets, and data pre-processing, and then pass it to the data alert generator module for analysis. The module can be divided into data capturing and data preprocessing in two parts. The main significance of the data preprocessing is to reduce the useless information by using rules into the data alert generator module, to improve real-time [3].

B. Data Alert Generator Module

Data Alert generator module (like SNORT) is composed mainly by the client agent. The destination host, the data collected from the local client received at the data, analysis and processing, to determine whether they are system or network intrusion

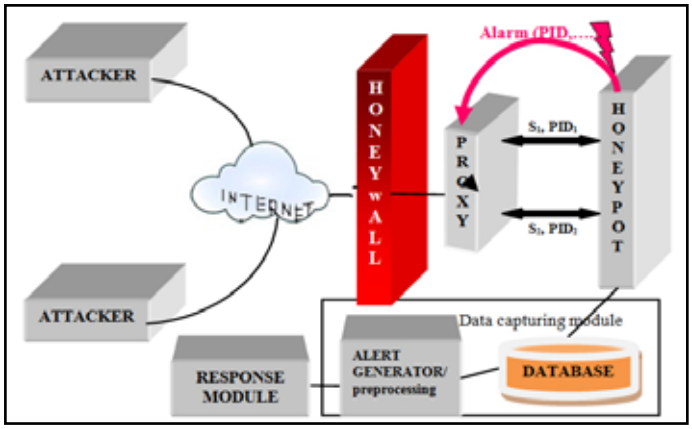

Fig. 1: Honey Pot System

This reduces the transmission of information in the network to ensure data security, while enhancing the response in real time. The module contains a variety of data detection client component rule, along with the different data sources to achieve different alert tasks. They work together to complete complex intrusion detection. Client specific discovery can be divided into host and network testing client, respectively, the client audit data and network data packets to be monitored [6].

V. The Honeypot Client

Fig. 2, describes the internal software architecture of the honeypot host, which consists of the honeypot service, the HIDS-manager (hidsmgr), the honeypot monitor (monitor) and the host intrusion detection system (HIDS). Honeypot service, monitor and HIDS manager are running in user-space, However the HIDS is located in the kernel space.

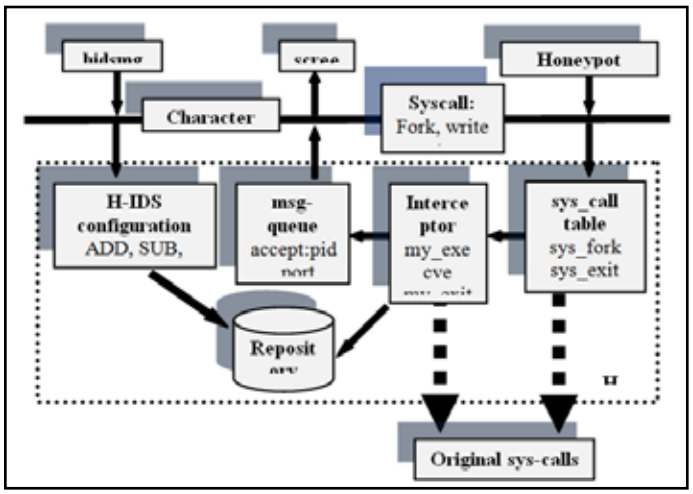

Fig. 2: Honeypot Client System

Usually, most common operating systems make a individual differentiation between application and operating system. Each time a user-space process requires an operating system Service (e.g. the opening of a network socket) the service must system call to the kernel. Kernel tests the request of the process and then it makes a decision whether to fulfill it or not. By inserting a HIDS into kernel-space and by redirecting the system-calls to the HIDS, it is possible to extend the functionality of the kernel.

In this case, it's possible to observe on the basis of system-call level what a process in user-space does. And it is possible to introduce more detailed decision criteria in the Kernel to determine whether the desired action is allowed or not (a same type mechanism has also been addressed for performing access control on active networking nodes [2]). The common method of system call interception is depicted in fig. 3 and proves the interception of the socket system call. The user process uses the socket command to create a socket for network communication. A process must execute a system call to gain access to the operating system services. Generally, this is done by wrapper functions which are element of standard libraries.

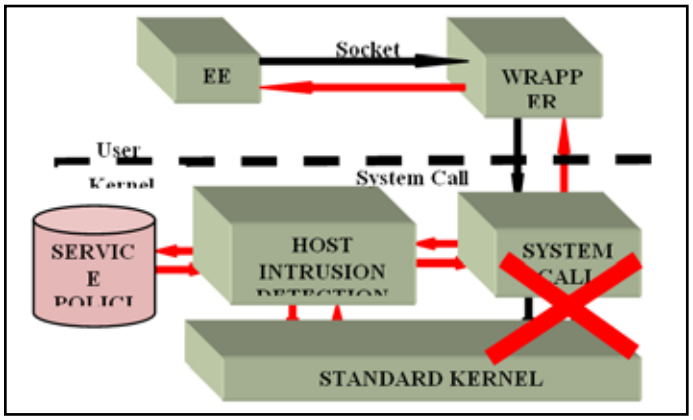

Fig. 3: Interception of a System Call

The wrapper function sets the variables to be submitted into the exact order, and then executes the proper system call. At the entry point into the kernel, the kernel uses a table it called system call table, for the forwarding of the incoming system calls to the matching functions. By conversion the destination of a pointer inside the system call table, we can redirect a defined system call to another function, in this case to the HIDS. That checks if the service is authorized to use a specific operating system service. If this test is passed, the HIDS then calls the standard kernel

function belonging to the system call. The HIDS is recognized as a Linux Kernel Module (LKM) and it detects intrusions on system-call-level. Whenever a user space process tries to execute a series of system calls that matches an attack signature a security alert is raised. In addition, the HIDS refuses to forward the last requested system-call to the operating system in order to prevent the honeypot system from being harmed. Consecutively, the HIDS triggers the monitor process to send an alarm message to the logging proxy. The attack signatures are specified inside the repository as a series of system-calls or simply as a black-list of disallowed system-calls. Besides this, it is also possible to configure the HIDS such that the execution of a specific group of applications (e.g. common gateway interface - CGI) is authorized. The user space front-end monitor handles the synchronization of local process ID and remote session ID between honeypot and proxy server. The HIDS checks for each observed system-call, whether or not it is executed by a process under supervision. The access to the operating-system is either granted or not, depending on the policy. What is more, if necessary a message is sent to the monitor and then forwarded to the logging component. Finally, the HIDS-manager can be used to reconfigure the HIDS at runtime. It provides a set of functions which first can be used to add, delete or change existing attack patterns. Second, the manager also allows modifying the list of services that must be observed by the HIDS.

VI. The Logging Proxy

Logging proxy looks for connection requests to the honeypot service which originate from a potential hacker. Other side proxy server acts itself as a client on behalf of the user or attacker and forwards the request (using its own IP address) to the honeypot services. This forwarding, the proxy server also creates an individual log file for each forwarded session (session ID) in addition and also contains the IP-address of the attacker, ports and a timestamp. For attacker, the proxy server is invisible and all the honeypot service or requests or back responses appear to be directly from the proxy host. The proxy logs the connection data of the honeypot service. Main task is to match attack session and PID of the corresponding process on the honeypot host which is a difficult task. Reason behind this difficulty is that one host keeps the logs while the other one detects the attacks. Incase a new client connection is initiated by an attacker then proxy sends the new session ID via the control channel for monitoring the honeypot . Thus one gets acknowledgement of successful connection of the proxy to the honeypot service. A message to the proxy server is send which contains the PID of the corresponding child process and the ports of the incoming connection. These ports are used to track which connection and session ID belong together.

Proxy server have one advantage that it maintains a list of currently established connections to the honeypot service. On a fork of the honeypot service a new PID message is automatically sent by the honeypot service monitor to the proxy server. On condition when process tries to execute a series of an unauthorized system-calls (attack signature) then honeypot monitor triggers an alert and sends a corresponding message to the proxy server. This alert-message contains the PID of the honeypot service process which violated the security policy. This alert message of proxy server tags the corresponding session and adds the alert information to the proper log file. The proxy server have tendency to stop the ongoing attack session which is another advantages feature, in local security policy.

VII. Process ID Tracking

Usually, processes can be identified by their unique process Identification number called as PID Tracking. Various networking services are as follow;

- It create a new process of environment for each connection which are acceptable. As per mechanism, to keep track of the processes that must be observed by the HIDS. Figure 4 depicts our principle of PID tracking. We open a shell which we subsequently add with the help of the HIDS manager.
- It can be used as the honeypot service
- It provide list of processes that are monitored by the HIDS. Next, the chosen honeypot service is started from inside the shell, which automatically assigns it to the group of services that must be observed by the HIDS.

A new process is created by a networking service through the execution of the fork() system-call (other possibility clone()).a A new process environment and assigns a new process ID is created by the operating system. Let us take a case that an attacker connects to the honeypot service, then the operating system creates the new process by executing the fork() command and returns two values. First value , which is the PID of the newly created process, is returned to the parent process, but the newly created child process receives a zero as return value. Whenever a process that is member of the list of services that must be observed by the HIDS creates a child process, then the newly created child process is immediately assigned to be a member of the list.

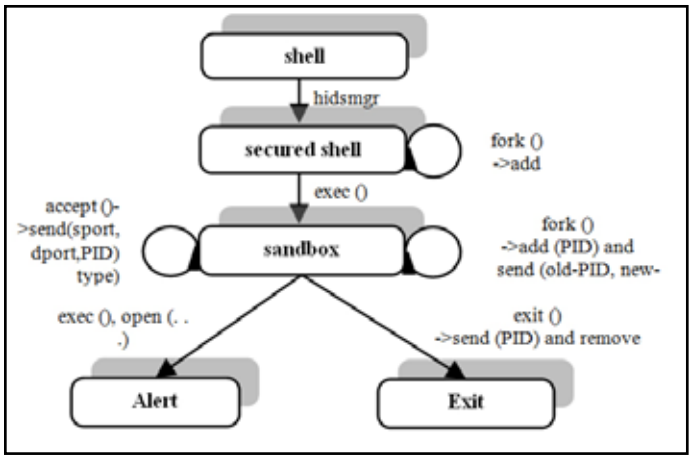

Fig. 4: Process ID (PID) Tracking

VIII. Response Module

Response module is designed as part of the invasion and track record of the source part of the invasion. This Tracking technology helps to track and allows the use of reverse invasion intruder to determine the source IP. If any host detects suspicious invasion, it will be redirected to the honeypot system, by the honeypot system monitor and record all information and suspicious behavior, the Mobile Agent log in to the router from the victim host recently to monitor the router packet to determine the direction of migration continue. Reverse tracking refers to the detected attack, attack path from the end of the actual attack path along the opposite direction back to the source of the attack.. Until you reach the best track was the source, reconstruct attack path, to determine the source of the attack.

Rule base is to describe the role of the user using the system of normal and abnormal behavior or describe the used system defects and other known methods of intrusion attack mode. Intrusion detection system improves the Performance, while preserving the rule base (abnormal and normal rules), so one can match with the

exception rules to determine a more accurate intrusion. At same instant, by comparison with the normal rules, we can determine the unknown intrusions. Honeypot system when used for determining the behavior of an unknown event for the invasion of the rules will be updated after a database intrusion detection system in order to achieve self-learning.

IX. Conclusion

With the rapid expansion of computer networks, Network intrusion prevention more difficult. By honeypot technology combined with intrusion detection systems, build a complementary active defense system. To meet the more complex network environment, reduce the honeypot system configuration and maintenance workload, could be considered dynamic honeypot technology to enhance the function of the system.

References

[1] Verword T, Hunt R.,"Intrusion detection techniques and approaches", Transaction on Computer Communication, pp. 1356-1365, 2002.

[2] A. Hess, G. Schafer,"Realizing a flexible access control mechanism for active nodes based on active networking technology", In IEEE International Conference on Communications (ICC 2004), Paris, France, June 2004.

[3] Tao Wenlin,"VMware-based virtual honeynet system", transaction on Computer Application and Software, pp. 131-136, 2006.

[4] Mukkamala S, Sung Ah, Abraham,"Intrusion detection using all ensemble of intelligent paradigms", Journal of Network and Computer Application, 2005, pp. 167-182.

[5] Zhang Chao,"Honeynet and intrusion detection and firewall linkage techniques", Technology market economy, pp. 42-44, 2007.

[6] Zheng Junjie, Xiao Jun mold, Liu Zhihua,"Based on Honeypot technology and network intrusion detection system", University of Electronic Science and Technology, pp. 257-25, 2007.

[7] Jigiang Zhai, Yining Xie,"research on network intrusion prevention system based on Snort", IEEE, in international forum on strategic technology(IFOST), Vol. 2, pp.1133-1136, Aug 2011.