Data Acquisition Module Research and Implementation of Distributed Intrusion Detection System

Li Liu

Department of Computer Science, LinYi Normal University, Linyi 276005, China

E-mail: xiaoke_1981@126.com

Abstract

This article first introduced the invasion and the concept of intrusion detection, and then described the data acquisition module in the Distributed Intrusion Detection. Finally, with a programming package intercepted, and through experiments to prove that good data packets intercepted.

Keywords: Intrusion detection system, Data packets, Data collection introduction

1. Preface

With the rapid development of computer network technology, more and more units will be make their own business to migrate the network, with the network attack tools and techniques opening, it made a highly developed network security has become a very important issue. The firewall is most common method of preventing the network attacks, which can be provided between internal and external network security, and reducing the risk of internal network attacks, but only rely on a firewall to protect the network is far from enough. Firewall achieve only coarse-grained filtering, can not provide real-time intrusion detection capability.

2. Intrusion Detection System Overview

2.1 Intrusion and intrusion detection systems concept

Intrusion was defined as any attempts to undermine the confidentiality of information and resources, integrity, and availability of behavior. Intrusion Detection System is for intrusion detection software and hardware combinations. It can detect real-time monitoring system activities, real-time discovery of aggressive behavior and take appropriate measures to avoid or minimize the occurrence of attacks have resulted in harm.

2.2 Distributed Intrusion Detection System

Distributed Intrusion Detection System has become an important research topic in network technology, Its data sources is packets of the network and host data, distributed intrusion detection technology uses a distributed intelligent agent architecture approach, there is a central agency and multi - constitute a distribution of local agents, local agents are responsible for the handling of local events, the central agency responsible for coordinating and handling the overall analysis .Every intrusion detection system to monitor every part of a single host, a number of intrusion detection system running and mutual cooperation. Any one intrusion detection system mistakes will not affect network-based detection, but have an impact on network traffic, and most are not very good for data collection.

In view of these shortcomings, this paper presents a method of data acquisition module, which achieved a good data collection.

3. Data Acquisition Module

Intrusion Detection System data sources include host-based data sources and network-based data sources.

Host-based data sources mainly refer to the operating system audit logs and a variety of applications such as logs. It can provide more complete log information.

Network -based data sources mainly refer to raw data real-time network packet capture. It can be found in many host-based data source, means of attack that can not be found in.

The system mainly achieved the network-based intrusion detection. However, in theory aspects of host-based intrusion detection, if the Agent communication in accordance with the data format as long as the input-related event data, you can combine the network with host-based intrusion detection, it not only can detect from outside attacks, but also can detect illegal operations within the staff.

3.1 Data Filtering

The data transmission of Ethernet was achieved through broadcasting, usually within a shared local area network access to all network interfaces have the physical media, all data transmission capacity. When connecting to the LAN during communication, TCP / IP protocol ensure.

The figure 1 is the data filtering process:

The machine can only receive the destination address is the packets of this machine, or the broadcast or multicast packets, other packet filtering out.

The data link layer determines whether the Ethernet address received is your MAC address; If not, and nor is it a broadcast or multicast address, then discarded, not to submit to the upper.

The network layer determines whether the target IP address is the IP address bound to your computer, if not, there will be no submission to the upper management.

The transport layer determines whether the target port has been opened in the computer, if not, there will be no commitment to the application layer.

After such a process, TCP / IP protocol only leaving the packets of the machine, all the other data packets discarded to ensure that the machine received only one destination address is the machine, broadcast or multicast packets.

To monitor flows through the network does not belong to its own native data, it must bypass the normal processing of the system, and direct access to the network infrastructure. First of all, the card mode set to promiscuous mode, when the network interface for this kind of "promiscuous" mode, the network interface with "broadcast address" for all received frames are generated to alert the operating system handles hardware interrupts flow through the physical media for each packet. The operating system direct access to the data link layer, intercepting data, you can listen to all the data flowing through the network card.

3.2 Packet interception

If you want to get the packets of the machine or the machine network, we can use TCP / IP protocol, bypassing the normal workflow system software, the card's operating mode set to promiscuous mode.

This system is in the Windows environment to intercept network data packets need to NDIS (Network Driver Interface Standard) Ethereal drive – Winpcap's support, WINPCAP from the UNIX platform LIPCAP transplant, it may bring device drivers added to the Win98, WinNT, Win2000 and WinXP on.

Windows is different UNIX from accessing the data link layer information on the mechanisms: Win32 platforms do not provide a direct low-level network access interface, it is necessary virtual device driver (VXD) for network monitoring capabilities, VXD driver provides the interface between an external program and the network card. There are three components for WINPCAP: one based on the BPF mechanism (kernel-level) data packets monitoring device driver, a low-level dynamic link library Packet.dll and a high-level static link library (Wpcap.dll), Wpcap.dll produces a UNIX platform under Ethereal Library. Libpcap compatible capture the output of the original collection of high-level.

Network data acquisition module is mainly responsible for monitoring the host system to capture the network connection information, and to get the data into standard form. The data is mainly crawl TCP / IP protocol header information. The system uses the Java programming for network packet capture.

The following is a program for the model used in Jpcap provide interfaces to intercept network packets (including TCP connections, UDP connections and ICMP packets of information):

```
Public void handle Packet (Packet packet ) {
    Try {
            System.out.println (packet);
File Output Stream out = new File Output Stream ("pcaplog.txt", true);
Print Stream output=new Print Stream (out);
If (packet instance of TCP Packet) {// Interception of TCP packets
Output.print(newDate(packet.sec*1000+packet.usec/1000));
output.print ("|");
output.print (((IPPacket) packet).src_ip.getHost Address ( ));
```

```
output.print ("->");
output.print (((IP Packet) packet).dst_ip.getHost Address ( ));
output.print ("|");
output.print (((TCP Packet) packet).src_port);
    output.print (":");
output.print (((TCP Packet) packet).dst_port);
output.println ("|TCP|");
}
else if (packet instance of UDP Packet){// Interception of UDP packets
output.print(newDate(packet.sec*1000+packet.usec/
1000));
output.print ("|");
output.print (((IP Packet)packet).src_ip.getHost Address());
output.print("->");
output.print(((IP Packet)packet).dst_ip.getHost Address());
output.print("|");
output.print(((UDP Packet)packet).src_port);
output.print(":");
output.print(((UDP Packet)packet).dst_port);
output.println("|UDP|");
}
else if (packet instance of ICMP Packet){/ / Interception of ICMP packets
output.print(new Date(packet.sec*1000+packet.usec/1000));
output.print("|");
output.print(((IP Packet)packet).src_ip.getHost Address());
output.print("->");
output.print(((IP Packet)packet).dst_ip.getHost Address());
output.print("|");
output.print(((ICMP Packet)packet).src_port);
output.print(":");
output.print (((ICMP Packet) packet).dst_port);
output.println ("|ICMP|");
}
        ….
      }
                    Catch (IOExption e)
              { }
}
```

3.3 Test Results

Data acquisition module has been tested is to test whether the intercept data packets smooth and accurate, the test environment for this module composed by the two hosts, one as an attack on the host, one as the master host, operating system is Windows XP, CPU is a P4 1.79GHz , NIC is a 100M Ethernet, memory is 1G.

After testing some results are as follows to Figure 2 :

Experimental results show that the conduct of this module can accurately intercept the data packets.

4. Summary

This system is characterized by the realization of data acquisition under Windows, and the work mode is set to promiscuous mode network card that can be better capture data packets.

References

Dai, Yingxia, even a peak, Lu Zhenyu. (2002). Adaptive Agent-Based Intrusion Detection System. *Computer Engineering*.

Lin, Qi, Zhang, Jian-wei. (2002). Aglet-based security model, *Computer Engineering*. 2002. 4.

Tang, Zhengjun. (2004). Introduction to Intrusion Detection Technology. *Machinery Industry Press*. 2004: 124-126.

Xiao, Jianhua, Zhang, Jianzhong. (2003). Mobile Agent-Based Distributed Intrusion Detection System Design and Implementation of MADIS. *Computer Engineering and Applications*. 2003.17: 164-165.

Zhao, Ming, Luo, Jun-Zhou. (2002). Agent-Based Framework for Intrusion Detection System. *Computer Engineering and Applications*.2002. 18: 176-178.

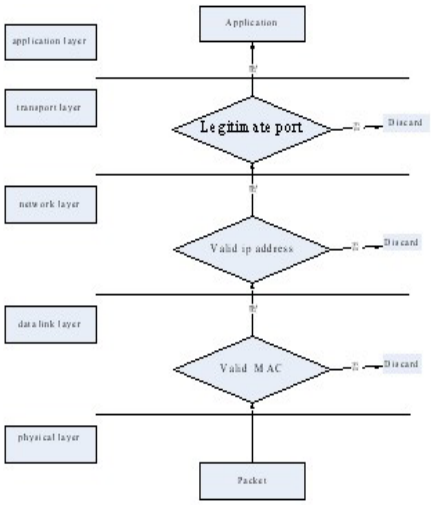

Figure 1. packet filtering process

Protocol Type	Source Address	Source port	Destination	Destination port	Data	Data contents
TCP	219.218.30.138	80	192.168.0.5	2867	0	
TCP	219.218.30.139	80	192.168.0.5	2867	0	
TCP	219.218.30.137	80	192.168.0.5	2866	568	
TCP	219.218.30.66	80	192.168.0.5	2867	568	HTTP/1.0 200 OK DATE:Ja,22
TCP	219.218.30.67	80	192.168.0.5	2867	200	
TCP	219.218.30.68	80	192.168.0.5	2867	8	

Figure 2. The testing results