Proceedings of the 2004 IEEE
International Conference on Networking, Sensing & Control
Taipei, Taiwan, March 21-23, 2004

NetFlow Based Intrusion Detection System

Tsang-Long Pao
Dept. Computer Science and Engineering
Tatung University
Taipei, ROC
tlpao@ttu.edu.tw

Po-Wei Wang
Dept. Computer Science and Engineering
Tatung University
Taipei, ROC.
g9006033@mail.ttu.edu.tw

Abstract - *In this paper, a NetFlow based anomaly intrusion detection system is presented. In addition, guidelines to properly configure and setup network device to minimize the possibilities that network attacks come from inside are also proposed. As the Internet becomes the platform of daily activities, the threat of network attack is also become more serious. Firewall along is not able to protect the system from being attacked through normal service channel. Furthermore, most of the current intrusion detection system focuses on the border of organization network. If the attack comes from inside, this setup does not provide any protection to hosts in the local network and the network itself. Therefore, we need to use other mechanism to protect the critical system as well as the network itself. We propose an inexpensive and easy to implement way to perform the anomaly type intrusion detection based on the NetFlow data exported from the routers or other network probes. Our system can detect several types of network attack from inside or outside and perform counter maneuver accordingly.*

Keywords: Intrusion detection system, network security, net flow, network probe.

1 Introduction

The explosive growth of the Internet makes it the popular platform of lots of daily activities. Example applications are online ticket reservation, banking, news publishing, and computer games. When more and more applications are porting to the Internet, the damage resulted from network attack will be much more serious.

For a medium or large organization, there will be a WAN (Wide Area Network) router sitting in the entrance of its local network. The first line of defense for the network or a group of mission critical servers is typically a firewall. However, since firewall will let the normal services, such as SMTP or HTTP, pass through, attack that exploit the bug of those servers may not be able to detected and denied by firewall along. Therefore we need an intrusion detection system (IDS) as the second line of defense. The packet passing through or sniffed by the IDS will be examined and compared with the known attack patterns to determine if the packet represents suspicious activity.

Among the existing IDSs, Snort is perhaps the best known open-source intrusion detection system [1]. Snort is attractive because the entire system including source code is freely available. Furthermore, quite a lot of dedicated programmers and users continually contribute to its evolution. However, there are several problems in the deployment of Snort. The biggest problem is that, with the default configuration, it gives too much alerts and the tremendous amount of logged activities sometimes let the manager completely ignores the real network attacks.

Another problem that most IDSs suffer is the performance. It is quite expensive to deeply examine the payload in a packet. This is especially true when the link goes to Gigabit or even 10 Gigabit. Consequently, hardware assisted IDSs are proposed [2,3]. However, the cost of this kind of devices is quite expensive and thus may not be an option for most users.

Most current IDS systems are deployed at the entrance of the organization network to defend the attack initiated from outside. But this setup is not able to protect the organization network when the attack is from inside. The problem is getting worse because more and more users bring mobile computing device such as notebook computer or tablet PC back and forth between home and office. Most user connect to the Internet at home does not provide enough protection to their computer. Once the mobile computer is being infected by a virus, it may initiate attack as soon as being connected to the network at office.

Our goal is to propose an infrastructure that uses existing facilities and some extra low cost hardware to establish an intrusion detection system without affecting the performance of the high speed backbone. In addition, this system can deny service to both inside and outside attackers. To achieve these goals, we use the network flow extracted from the packet header instead of examining the payload as the input to the system.

The network flow data is exported from routers and switches or other network probes in Cisco NetFlow format. To fully utilize the flow data, it is desirable to store them into a database. The publisher of ntop supplied a perl script that can read the NetFlow version 5 data from

0-7803-8193-9/04/$17.00 ©2004 IEEE

the network and store them into the MySQL database. MySQL is a free public domain relational database management system which is very fast and easy to manage. As the flow data being stored into the database, we can perform various analyses on them to determine if network attack exist.

In the proposed system, a statistical model similar to the one presented in [4] is used to determine if there is any suspicious network activity. Aggregation of data store in the database is a fast and easy task. Several aggregation operations are performed to extract the possible network attacks according to the traffic patterns of network attack behavior.

The organization of this paper is as follows. In Section 2, we will review the previous works on intrusion detection and prevention. The architecture of our proposed system will be discussed in Section 3. Test results of our system are presented in Section 4. Concluding remark is given in Section 5.

2 Intrusion detection and prevention

The DOS (Denial-of-Service) attacks are network attacks by loading computing or memory resources with a large number of requests. This makes the resources too busy or too full to handle legitimate requests. One of the most distinctive characteristics of DOS attack is the increase on network activity since the attacker needs to establish a large number of connections in order to achieve its objective. The PROBE attacks make use some network scanning utilities which automatically scan a network of computers to find vulnerabilities. The DOS and PROBE attacks generated network activity is what we are focused in this paper.

2.1 Intrusion Detection

Intrusion detection can be divided into two categories, host-based and network-based [4,5]. Host-based intrusion detection is generally used in protecting critical server systems or network devices by analyzing the operating system and application behaviors. Network-based intrusion detection, on the other hand, is used as a network monitor that can identify suspicious network activities through analyzing the network traffic pattern.

Misuse detection and anomaly detection are the two main analysis techniques in intrusion detection. Misuse detection uses the signatures of the known intrusion to identify the attacks. Anomaly detection, on the other hand, tries to find any abnormal behavior by comparing network activities to those established normal profiles. Any network activities that have unacceptable deviation will be categorized as possible result of intrusion. Misuse detection uses known signatures so it may not be able to catch new intrusions. However, the accuracy is high and the false positive rate is relatively low. Anomaly detection can detect unknown intrusion but may sometimes result in high false positive rate. That is, it may flag a normal activity as an intrusion and prohibits that activity to continue.

A good intrusion detection system should have high detection rate and low false positive rate [6]. These are still the most serious challenges to the current intrusion detection technologies. Since most IDS available today rely on signature matching and is thus unable to catch even a slightly deviated version of known intrusion. Therefore, using misuse detection technique alone is not enough. Our goal is to develop an anomaly intrusion detection system that can act as the first line of defense using only the network flow information.

2.2 Intrusion Signatures

There are several types of anomaly attacks: reconnaissance, exploits and DOS. Reconnaissance attacks include ping sweeps, TCP or UDP port scans, etc. In the exploits attack, the intruders use known or unknown hidden features or bugs to gain access to the system. In the DOS attacks, the intruder attempts to crash a system, jam network links, overload the CPU, exhaust the system resource, or fill up the storage. In the DOS attacks, the intention of the intruder is not trying to retrieve sensitive information, but to simply prevent the system from being usable, jam the network links or crash routers to make the network inaccessible.

In the reconnaissance attack type, ping sweeps simply ping a range of IP addresses to find which machines are alive. TCP scans probes for open well known TCP ports looking for services the intruder can exploit. Scans can use normal TCP connections or stealth scans that use half-open connections or FIN scans. Because UDP is a connectionless protocol, UDP scans are more difficult and slow to perform. To scan an UDP port, we can send a garbage UDP packet to the destination port. If the port has no service listen on, most machines will respond with an ICMP message indicating destination port unreachable. However, many operating systems limit the rate of ICMP messages, which will slow down the scanning process.

In the DOS attacks, the ping-of-death attack sends an oversized IP packet (larger than 64 kBytes after packet reassembled from fragmentation) that may caused the targeted machine to freeze, crash, or reboot. In SYN flood attacks, the attacker sends huge amount of TCP SYN packets to the victim machine. Since the connection is never complete and the amount of requests is tremendous, the system will run out of resources and start dropping normal connection request. In Land and LaTierra attack, the attacker sends SYN packets with same source and destination address as well as source and destination port. This will make the vulnerable system goes into infinite loop trying to complete the TCP connection.

The IP spoofing technique is commonly used in the DOS attacks because the attacker is not expecting any response from the victim. Since the source IP address in a packet is not used in routing, the attacker can forge (or spoof) the source IP address while the designated victim still can receive the packet. As a consequence, it is difficult to trace down the real source of attack and also quite hard to prevent the DOS attack without affecting the normal services provided by the victim.

3 NetFlow-based IDS

3.1 Components of intrusion detection

Network intrusion detection systems are consisted by probes, assessment engines, response agents, rule database, and user interface for rule editing and control console [7]. In this paper, we will focus on the relationships between probes, assessment engines, and response agents.

Probes are responsible to capture the network packets and extract required data from either the header or the payload portion of the packets captured. The collected data is sent to the detection engine. In this paper, we only use the information gathered from the header of packets. Flow data exported in NetFlow format from the routers or other probe utility such as nProbe is the input to our system. We can extract intrusion signatures of several types of network intrusion from these flow data.

We are not trying to detect all kinds of anomaly intrusions but just certain types of intrusions that can affect the performance of the network access. Our focus is on the port scan, DoS attack and other attacks that will send out numerous amounts of packets. The assessment engine collects and analyzes the information gathered from the probes and issue one or more responses if any intrusion behavior is found. The data is stored into a relational database for long term analysis. Since a relational database engine is capable of aggregate data in different ways, it is easier to adapt new type of intrusion.

3.2 System architecture

Our proposed system architecture is shown in Fig. 1. This architecture can be used in general cooperate or campus network which has limiting resource to acquire high end network devices. The WAN router may be an ordinary router or a layer 3 switch if Gigabit Ethernet is used in the WAN connection. The edge switch is a layer 3 switch which support access-control list capability. These router or switches should have SNMP interface or command-line interface similar to the user interface provided by Cisco IOS. The interface is for the response agent of the IDS to send command to the network device in order to setup the access-control list. The probes may be a process running inside the router or L3 switch or a computer running ntop, nProbe or similar utilities that can export flow data in NetFlow format.

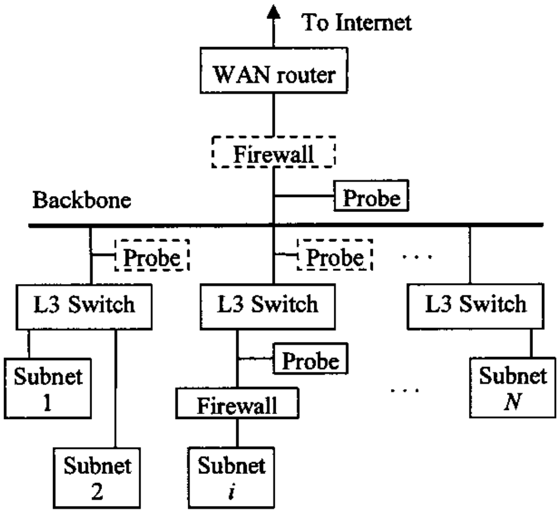

Figure 1. System architecture of a enterprise or campus network with multiple subnets.

The firewall is used to protect mission critical servers sit inside Subnet *i*. We may use multiple probes on the link to the Internet or important servers to gather traffic information. The probes enclose by dashed box is optional depending on the requirements. We can also install multiple assessment engines to distribute the load when the volume of traffic information need to be analyzed is large. Thus, this system is quite scalable.

3.3 NetFlow

NetFlow is a flow data exporting mechanism defined by Cisco and is supported by all Cisco router families as well as routers from some other companies. Some network probes, such as ntop (nprobe), a free public domain network monitoring and analysis tool, are also able to export flow information in NetFlow version 5 format. A network flow is defined as a unidirectional sequence of packets between identical source and destination endpoints. In each flow data, the number of packets and octets transferred during that time frame with identical source and destination endpoints will be accumulated. Therefore, we can extract the traffic information from the flow data received. The flow data can be used as the base for accounting/billing, network planning and analysis, network monitoring, application and user monitoring, etc. From the collected data, we can analysis the flow pattern and determine if there is any possible network attack.

As stated in the NetFlow documents by Cisco, it is improper to activate the NetFlow service in a "hot" core/backbone routers or routers running at very high CPU utilization rates. It recommends careful planning to activate on edge/aggregation routers for ISP and WAN access routers for Enterprise or campus network which capture the data required for planning, monitoring and accounting applications. Therefore, we use the nProbe program to collect traffic information mirrored from the link we are interested. This will not affect the performance of the router.

3.4 Intrusion Detection

We will discuss the methods we used to detect the intrusion in this section. Different intrusion types may need different aggregation operations. But, sometimes, different intrusion types may also share the same traffic flow pattern.

3.4.1 Ping sweep detection

Ping sweep detection can be done by collecting UDP flow data with identical source IP address but different destination port or IP address. Another characteristic of ping sweep is that the packet length is short. New ping sweep technique may try to hide itself by distributing the scan to long time interval. Therefore, short time and long time aggregation are required to detect ping sweep.

Another signature of ping sweep is one-way traffic. Since the attacker does not know exactly what machine is up and what services are on line, there will be quite a lot of packets sent to IP address that no host is using. A short time aggregation with this characteristic in mind will uncover ping sweep attack easily.

If the attack is from inside, we will see quite a lot of flows with identical source IP but different destination addresses or ports. Even when distributed scanning is used, we can still observe this phenomenon since most of the packets will pass through and captured by the probe. Therefore, NetFlow based intrusion detection system can detect ping sweep without difficult.

3.4.2 TCP/UDP port scan

Attackers use TCP/UDP port scan to find out which service is open and whether it is venerable. Since the intention of the activity is to collect information of systems that the attacker can exploit, the source of this type of attack can easily be identified, although it may come from a computer being cracked. Thus, we can uncover this attack by counting the number of packets to same IP address with different ports. Also, in TCP port scan, it use normal TCP connections or stealth scans that use half-open connections or FIN scans. So, the size of packet is usually short. Thus, even the attacker try to distribute the scan to long period of time, it can still easily be identified. Longer term aggregation can also be used to uncover distributed TCP/UDP port scan to different hosts in our network.

3.4.3 DOS detection

The characteristic of DOS and its variant DDOS is that large amount of connection requests sent to one or more IP address in a shore period of time. Usually, the source IP of this type of attack is forged, thus we may not see many successful communication from the victim IP address. What we see in the traffic pattern is that a packet is sent from a possibly forged source IP address to our victim IP address and a response packet is returned. Then, no further communication exists between these two IP addresses. The response to this type of attack is difficult because it is impossible to deny this attack without affecting the normal service provided by the victim host. However, we can prevent this kind of attack coming from inside by properly setup the edge layer 3 switch that allow only source IP in that subnet to come out. Thus, if all ISP, cooperate or campus network manager follow the guidelines state below, we can eliminate quite a lot of DOS attacks without affecting the performance of the network access.

Edge switch setup guideline: Use the access-control list to allow only packets with source IP address belong to the subnet assigned to that interface or VLAN to come out and deny all packets with source IP addresses other than those allowed.

If an intrusion is detected by assessment engine, one or more responses are performed by a response agent. Examples responses are sending an e-mail to the network administrator; denying the specific IP address at the WAN router or edge switch using access-control list; or providing information to another assessment engine. Since our goal is to deny the intrusion no matter the intrusion is coming from outside or inside, we need to develop ways to communicate with the network devices, specifically the WAN router and edge layer 3 switches. If the router or switch has a command line interface, we can use telnet or ssh to set up the access-control list with the help of 'expect' program, a freely available public domain utility. If the router or switch does not have a command line interface but with SNMP management capability, we can use SNMP to setup the access-control list in a similar way.

4 Implementation and testing results

Our system is focus on detection and prevention of the following types of attacks:

1. Ping sweep: This type is fairly easy to detect by aggregating the network flow on single source IP address to different IP addresses or ports. To prevent distributed scan, we will aggregate over a period of 10 seconds, 60 seconds and 300 seconds, each with different threshold. Any event that is over the threshold is susceptible and will be put into warning list. While in a certain period of time, the same IP address is put into the list, it will be identified as intrusion.

2. DOS: This type of attack can be prevented by using the access-control list in the routers or layer 3 switches. On the edge, we will deny any source IP address which is not supposed to appear on that subnet. This will prevent the attack coming from inside node. Figure 2 shows an example of such an access-control list. From the number of deny access, it surely indicates that some sort of abnormal activities were going on in that subnet. On the entrance of our own network, the rule is to deny any packet that with source IP address belongs to private IP address or in the IP address range of our own. When DOS do occur and the source is from outside, the response agent may issue command to the WAN router to stop the service of the victim host temporarily. When the attack stopped, the response engine will send command to reopen the service.

```
Standard IP access list 26
  permit 140.129.26.0, wildcard bits 0.0.0.255
  deny   any (98612 matches)
Standard IP access list 30
  permit 140.129.30.0, wildcard bits 0.0.0.255
  deny   any (14946 matches)
```

Figure 2. Example of access-control list to deny source IP addresses not from the assigned subnet.

3. TCP and UDP scans: This type of attack can be detected by aggregate the flow data in a way similar to ping sweeps with more restrictive threshold. This is because it is rare that a normal user will connect to multiple hosts in a very short time interval. Thus, if we see a source IP address which is not owned by a server tries to connect to a large number of remote addresses, it is very probably a TCP or UDP scans.

4.1 Implementation of intrusion detection system

If the traffic is light and the router is capable of exporting NetFlow data, we can activate the NetFlow service in the router. Otherwise we will mirror the network traffic to be monitored from the switch and gathered by a computer running ntop or nProbe, or other similar utilities. The flow information is stored into a relational database. Depending on the traffic, we may need to delete the flow data older than certain time in order to keep the database operation fast enough. For our purpose, retaining flow data span around 30 minutes should be enough. In high volume traffic environment, we may need to reduce the amount of data.

The assessment engine can be implemented by any programming language that is capable of retrieving data from database. Some implementation guidelines and modules are addressed in [8]. However, when considering the user interface, we choose PHP script to implement the system. The PHP interpreter can be used standalone or as a plug-in of apache or other web server. It also has the capability to execute other programs in the same system. Therefore, it is a fairly good choice as the development tool.

The assessment engine aggregates the data stored in the database using the signature discovery methodologies discussed in the previous section to detect intrusions. To detect possible attack as thoroughly as possible, we aggregate the data with interval over 10 seconds, 1 minute, 10 minutes and even 30 minutes if it is possible. Some network attacks such as slammer (infecting MS SQL server 2000) which emits huge amount of packets in a very short period time can be easily detected by aggregation of data from recent 10 seconds. The slammer attack will pull down the network performance severely and should be stopped immediately. If the attack is from inside, the response agent should be able to communicate with the edge switch and modify the access-control list accordingly. If the attack is coming from outside, we need to use the IP address or the port the attacking packets used to deny the packet from penetrating into our network.

In our implementation, the response agents communicate with the router and switch using telnet. The router and switches used have a command-line interface that ordinary network manager is quite familiar with. Since the command-line interface is working in interactive mode, the "expect" utility is used to transfer the interactive operation to batch operation. There are several different expect scripts for different purposes. The PHP script call the expect script through "exec" system call and wait the execution to terminate successfully before continue. Of course, we can use SNMP as the protocol to communicate with the routers and switches, however, it is much more difficult to learn and implement by general network managers.

The system is implemented in a campus network which span for 50 class C subnets. The router and edge switches used are the Cisco 3550 because the WAN connection is a Gigabit Ethernet. Each switch is responsible for routing several class C subnets. The VLAN technique is used to partition the 24 ports of Cisco 3550 into groups to accommodate the actual network structure. Each VLAN is assigned to handle a class C subnet. The standard access-control list is used to permit only the source IP matches the class C subnet handled by that VLAN. Packets with other IP addresses are discarded. From the access-control list log, when the inside node is infected by virus or being intruded, it will send out packets with forged source IP address. Those packets are denied by the switch, thus put no harm to the network.

As the intrusion coming from outside, the access-control list in the WAN router is adjusted accordingly. The denying rule may be based on the IP address or the TCP/UDP port depending on the type of intrusion. However, this defense may pull down the router performance if the attack changes the IP address and port simultaneously. Therefore, we need to implement a recover procedure to discard rules no longer effective. Thus, we need to know the information of packets count that is denied by the WAN router for each access-control list rule. If no any packets matched with the rule, it will be eliminated. Also, we need to limit the number of access-control list so that the CPU utilization will not get too high. If the attack uses different source IP address as well as different port, we may consider to shutdown all the connection to the victim host. As long as the deny count of specific access-control list rule growing faster than normal, we can not eliminate that rule. However, this would effectively shutdown the services provided by the victim host. The choice is on the network manager.

5 Conclusions

In this paper, we propose a NetFlow based IDS and a network architecture to deny certain types of intrusion at the network entrance or at the edge. As the Internet is increasingly used in commercial purpose, network intrusion, even the DOS type, becomes a real threat. Furthermore, the evolution of intrusion techniques makes the successful detection more and more difficult. So we can not rely only on single IDS to protect our critical systems as well as the network itself. The system we propose can be used as the first line of defense. The cost of implementation is not high and will generally not affect the backbone performance. From the logs produce by the routers/switches and other components in this system, we can verify that this system can effectively detect and deny the types of network intrusions we are focus on at the edge of the network so the critical server systems and the core part of the network can be protected.

References

[1] "Snort, The Open Source Network Intrusion Detection System", http://www.snort.org/.

[2] Necker, M., Contis, D., and Schimmel, D., "TCP-Stream Reassembly and State Tracking in Hardware", Field-Programmable Custom Computing Machines, 2002. Proc. 10th Annual IEEE Symposium on, pp. 286-287, Apr. 2002.

[3] Hutchings, B.L., Franklin, R., and Carver, D., "Assisting Network Intrusion Detection with Reconfigurable Hardware", Field-Programmable Custom Computing Machines, 2002. Proc. 10th Annual IEEE Symposium on, pp. 111-120, Apr. 2002.

[4] Caberera, J.B.D., Ravichandran, B., and Mehra, R.K. "Statistical Traffic Modeling for Network Intrusion Detection", Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000. Proceedings. 8th International Symposium on, pp. 466-473, Sep. 2000.

[5] Yau, S.S. and Xinyu Zhang, "Computer Network Intrusion Detection, Assessment and Prevention Based on Security Dependency Relation", Proc. COMPSAC 1999, pp. 86-91, Oct. 1999.

[6] Yan Qiao and Xie Weixin, "A Network IDS with Low False Positive Rate", Proceedings of the 2002 Congress on, Vol. 2, pp. 1121-1126, May 2002.

[7] E. L. Witzke, T. D. Tarman, S. Ghosh, and G. Woodard, "A Novel Scaleable Architecture for Intrusion Detection and Mitigation in Switched Networks", MILCOM 2002, Proceedings, Vol 1, pp395-399, Oct. 2002.

[8] Hashim, S.J., Jumari, K., and Ismail, M., "Computer Network Intrusion Detection Software Development", Proc. TENCON 2000. Vol. 3, pp. 117-123Sep. 2000.