www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 6 June 2013 Page No. 2120-2124

A NEW f-IDPS APPROACH FOR THE INTRUSION DETECTION IN HIGH-SPEED NETWORKS

Anshuman Saurabh[1] Deepti Sharad Nirwal[2]

[1]Department of Computer Science and Engineering, [2]Department of Computer Science and Engineering

[1]Assistant Professor, Swam Vivekanand Subharti University,Meerut ,India, [2] M.Tech Scholar, Swami Vivekanand Subharti University, Meerut ,India

[1]anshumansaurabh@gmail.com

[2]deepti.nirwal13@gmail.com

ABSTRACT

As the networks become faster and faster, the emerging requirement is to improve the performance of the Intrusion Detection and Prevention Systems (IDPS) to keep up with the increased network throughput. In high speed networks, it is very difficult for the IDPS to process all the packets. Since the throughput of IDPS is not improved as fast as the throughput of the switches and routers, it is necessary to develop new detection techniques other than traditional techniques. In this paper we propose a rule-based IDPS technique to detect Layer 2-4 attacks by just examining the flow data without inspecting packet payload. Our approach is designed to work as an additional component to existing IDPS as we acknowledge that the attacks at Layer 5 and above require payload inspection. The rule set can be constructed and tested on a real network to evaluate the performance of the system.

Keywords: Intrusion Detection and Prevention Systems, High Speed Network, Layer 2-4 Attacks, Signature based detection, Flow-based intrusion Detection, Host based attacks

I. INTRODUCTION

An intrusion detection system (IDS) inspects all network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system. An ID can detect many types of malicious network traffic and computer usage that cannot be detected by a conventional firewall. These include network attacks against vulnerable services, data driven attacks on applications[1], host based attacks such as unauthorized logins and access to sensitive files, and malware (viruses, Trojan horses, and worms). An IPS is the next security layer that combines the protection of firewalls with the monitoring ability of an IDS to protect networks[3]. IPS are designed to sit in line with traffic flows and prevent attacks in real-time. Therefore, IPS can be named as IDPS[1].Rapid increase of Internet users throughout the world has resulted in exponential growth of Internet traffic in wide area networks (WANs). On the other hand, throughput of intrusion detection and prevention systems does not improve as fast as that of

network infrastructure. The global Internet connections reach 10Gbps capacities. Data payload inspection in high speed networks is almost impossible[6] . Currently, the most of the intrusion detection methods look for specific signatures in the IP packet payload. This signature based detection methods have scalability problems to work at backbone rates slowing down the response time of the network. Also, inspection of payloads violates the user data privacy. And these methods are useless for the encrypted data[1][2]. When the speed of IDPS and network does not match it is often the case that the packets are dropped. These dropped packets may have data with attack signatures, which causes a high false negative rate in the IDS[6]. Therefore it is necessary to develop efficient and high performance based intrusion detection techniques. In this paper, we propose a new Engine is not enough to detect all the attacks by itself and the attacks that cannot be detected by it are detected by the IDPS. The proposed Detection Engine works with a set of *rules* that describe the behavior of the attacks in time at Layer 2- Layer 4 *flow* level. We can construct the rules by performing the attacks in a small scale lab network by using the attacks that we generate artificially. We can then collect flow level data from different network devices and then can correlate this data to intrusions that are detected by a commercial IDPS to deduce the rules[7]. Note that as the Detection Engine does not perform DPI it can eliminate the attacks even if they are encrypted. Since this solution is flow and rule based, it can eliminate a new attack. For signature based systems, an attack and its variants must be defined in the database. Since the behaviors of an attack and variants are the same, it is easy to detect and prevent variants of attacks.

A. Intrusion detection and prevention systems for high speed networks

The goal of proposed model f-IDPS in this paper is to reduce the load on the traditional IDPS. Using IDPS more efficiently increases the network throughput. We first present an overview of high speed networks. Finally, the attack generation tools and traffic analysis tools that are used to develop intrusion detection technique in the solution in this paper are introduced.

II. RELATED WORK

A. Existing IDPS Implementations for High Speed Networks

approach called as f-IDPS for the efficient intrusion detection and prevention for high speed networks. We observe that the attacks in the network are composed of attacks that act at Layer 5 and above and attacks that act at Layer 2 to Layer 4[4]. IDPS perform DPI by checking a long list of signatures for all of the packets regardless of the network layer the attack operates. Acknowledging that the attacks at Layer 5 and above require DPI,[1] we propose an additional component that we call f-IDPS *Detection Engine(here f stands for flow)* to existing IDPS which analyzes and detects the Layer 2-Layer 4 attacks and informs the IDPS to take the required actions. It is no longer necessary for the IDPS to check the related signatures to the attacks that can be detected by the Detection Engine and hence the network can respond faster [1]. However, the Detection

[2].

The netflow based anomaly IDS aims for the detection of ping sweeps, DoS attacks and port scans using netflow data taken from routers and switches. Wide variety of DoS and scanning attacks are examined and it is shown that several categories (bandwidth based, claim-and-hold, port-scanning) can be scalable detected. The existing system is shown in below figure

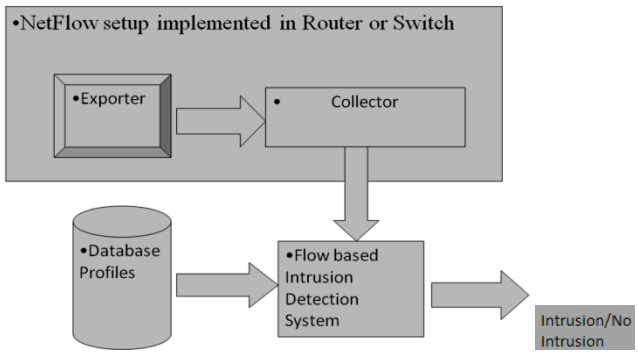

Fig 1: Existing IDPS for high speed network [2].

III. THE PROPOSED WORK

A. The proposed detection engine (f-IDPS)

Our approach is based on the hypothesis that majority of the all of the common L2-L4 attacks can be detected by inspecting the flow level characteristics without performing deep packet inspection [1]. We note that, deep packet inspection is required for L5-L7 type of attacks which do not have L2-L4 reconnaissance phase. We expect that, detecting the attacks by

using simple flow information before IDPS have to examine them decreases the resource consumption of the IDPS hence provides a scalable solution for IDPS for high-speed networks. Detected attacks can be dropped automatically if the intrusion detection server in our approach is on the path of the intrusion traffic (in in-line mode). Otherwise, server makes denial requests to the switch, router or firewall.

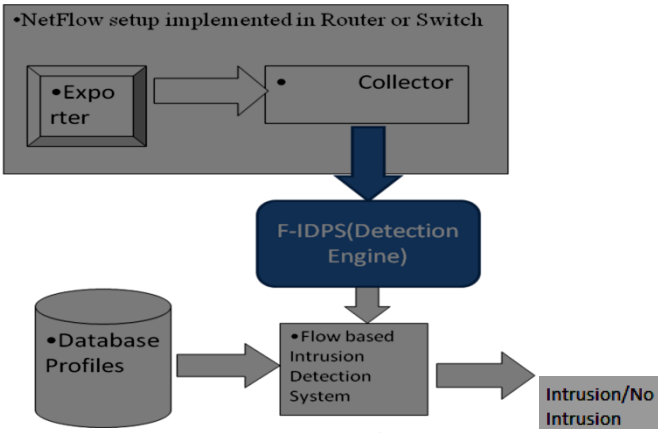

Fig 2: The proposed f-IDPS[1]

The approach in this paper can be implemented and tested using laboratory networks. A laboratory network is used for preliminary characterization of the attacks and building a first set of rules. Layer 2-4 attacks are first characterized in the laboratory network to detect through the complete network traffic using both flow-based and payload inspection based methods such as flows, logs, sniffer and SNMP traps. The discovered characteristics include the usage of TCP and UDP ports at the same time or one after another, connection duration, IP packet size, number of connection, and flow using netflow analyzer, Syslog server, SNMP collector and sniffer in the laboratory stage small network. The flow pattern is defined using source/destination IP address, number of bytes and packets associated with an IP flow, protocol, TCP/UDP source/destination port, packet length, MAC address. We first extract the flow patterns of Layer 2-4 attacks. Information gathered from the flows and captured packets on the network can be used to expose the attacks' behavior. A given flow goes through a set of *states* that we define as reconnaissance, scanning, gaining access, attack, maintaining access and covering tracks in time[1]. Depending upon this, the rules are constructed. For instance, we observe that if a packet destined to a port 9996 is following a packet destined to 445, it is most probably a worm. If a packet destined to time[2]. The simplified final rule set is obtained by merging the rules for different types of attacks that expose similar characteristics. This final rule set can be tested in the same real network.Note that, payload inspection is not the component of the intrusion detection technique in our approach. It is only used in the laboratory networks for analysis and verification of the attacks' characteristics. In our approach, detection is based on not only using netflow data but also logs and traps. Statistical information gathered from netflow data is correlated with the syslogs and traps. It is also possible to add new attack types to our database. Rather than attempting to detect of all kinds of L2-L4 attacks, certain types of attacks (such as DoS attacks, uncomplicated DDoS attacks, scans, worms, poisoning, spoofing, protocol anomaly attacks, password attacks) are the focus in the solution in this paper.

B. Flow analysis and small laboratory network

Our aim is to analyze the attacker behavior first in a small laboratory network, then in a larger network. We have to examine just about 40 MB file that includes one hour traffic of the small network.

C. Flow Pattern

In the small laboratory network, a set of network attacks can be staged and examined to extract the characterization for these attacks. Our aim is identifying any flow pattern that is specific before the attacks or undesired traffic generated or while attack is happening. Attacks can be investigated

port 445 is seen in the network, flow pattern is supposed to be in reconnaissance state [1].

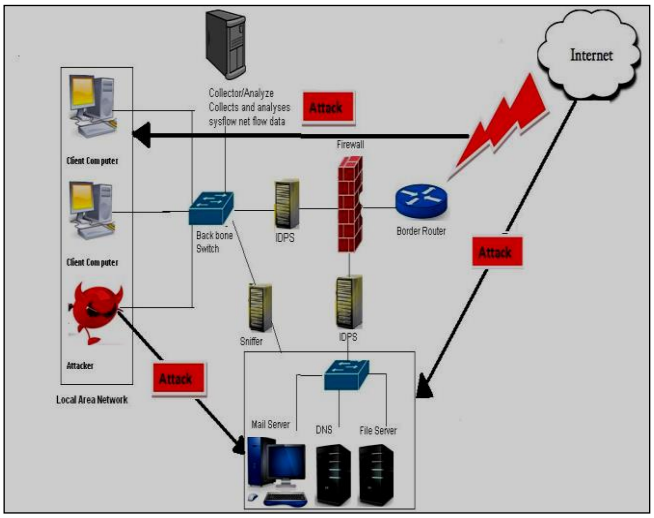

Fig 3: Small Laboratory Network setup for f-IDPS [1]

The small laboratory network as shown in Figure 3 consists of a border router, backbone switch, DMZ switch, firewall, intrusion detection and prevention system (IDPS), servers and client computers. According to the IDPS's logs, attacks' behaviors are examined from the stored data in the Collector-Analyzer and Sniffer Server. Sniffer is just used to examine the packet payload and the signature of the IDPS as a part of the verification

IV COMPONENTS OF PROPOSED F-IDPS

A Firewall

We use the firewall in this topology to collect syslog data. The syslog collected from the firewall depends on the filter configuration since the logs are generated as packet matches to a filter. Firewall is configured to send logs in syslog format to the Collector- Analyzer.

B. Border router:

configuration of border router, the IP address of the Collector is defined as the destination of the exported netflow data and also as the destination of SNMP traps.

C. Backbone Switch:

Backbone switch is configured to route the traffic between VLANs (Virtual LAN) and outside network. A VLAN is a group of hosts with a common set of requirements that communicate as if they are attached to the same wire, regardless of their physical location.

D. DMZ Switch:

DMZ Switch is configured to send netflow data and SNMP traps to Collector- Analyzer.

E. Collector-Analyzer, Sniffer:

To collect the data, log and analyze them, two computers are used, which we call Collector-Analyzer and Sniffer. Collector-Analyzer collects syslog, netflow data and SNMP traps directed to it by router, switches, firewall and IDPS and perform correlation of this data.Sniffer Server captures the packets passing through backbone switch and DMZ switch to verify the attacks detected by our solution are not false positive.

F. IDS:

We can use two different leading commercial signature based intrusion detection systems. In their database about 2300+ signatures are defined and enabled. In addition, we get the detection logs of these two commercial products to the Collector-Analyzer to analyze our syslog data, SNMP traps, netflow data and sniffed data. Commercial IDPS's are

All of the interfaces of the border router are configured to export the flow traces which are supplied by netflow.[2] In addition, SNMP traps are enabled on the border router. SNMP traps enable an agent on the router or switch to notify the management station of significant events by way of an unsolicited SNMP message. In the

not the component of the intrusion detection technique in this approach. They are used for analysis and verification of the attacks' characteristics. When a new attack is detected by commercial product, it triggers the process on Collector-Analyzer to analyze the flows related with this attack[9].

V COLLECTING DATA AND CONSTRUCTING THE RULES

A. Collecting Data

While a sample attack is created by our attack generation tools on the attacker PC to the victim PC, the traffic is mirrored to the Sniffer and also the flows are exported to the netflow analyzer from the switch. We collect the syslogs from the firewall and the servers. Netflow data and SNMP traps are taken from border router, backbone switch and DMZ switch. On the other hand commercial intrusion prevention system's logs are taken. All of the logs and alerts are collected and analyzed on the Collector- Analyzer. Flow patterns such as the usage of TCP and UDP ports, connection duration, attacker's behavior before it attacks, and victim's behavior after it is attacked and flow time are examined. Ethereal is working as a packet capturer and also as a protocol analyzer[1][3]. Data is store on the Sniffer Server. Ethereal captures and evaluates the data by examining the headers and payloads of the packets without affecting the client on the original port[1][8]. Signatures of the intrusion detection systems and the payloads are compared. Marking the IP address of an attacker and tracking the attacker behavior is done using sniffer software and PERL scripts written on the Sniffer Server. The

processes on the Collector-Analyzer trigger that scripts and get the output of them to analyze and verify whether the packet that is supposed to be an intrusion because of the rules is really an attack or not[1].

B. Constructing the rules

Reconnaissance attacks collect information about target network before the attack and the suffered victim's flows are passing through the network after. As we deepen the examination of the flows, it is seen that not only an attack is happening at a specific time but also there are some unusual flows that happen before and after an attack happens[1]. Attacker's behavior is defined by the flows that are for reconnaissance and victim's behavior is defined by the involuntary flows after it is attacked. When we catch an attack packet using commercial IDPS or sniffer, we get the timestamp of the packet and examine the traffic of victim and attacker in the interval of 10 minutes, 5 minutes before the attack happens and 5 minutes after the attack happens. This 10 minute interval is the starting value to analyze and find an optimum value for the detection accuracy. If we encounter an unusual flow of the attacker or the victim in that interval, we extract the flow pattern and the state changes for that attack[1]. All of the flows, logs and traps are stored in a database in Collector-Analyzer. When the sample attack is generated, the flows, logs, trap as well as the CPU and memory usage levels in the devices. 5 minutes before the attack happens and 5 minutes after the attack happens, adding up to a 10 minute total interval, are taken and inserted in another table in the database on Collector-Analyzer[1][5].

VI CONCLUSION

In this paper, we present our new approach to scalable intrusion detection for high- speed networks. Currently, the most accurate intrusion detection methods look for specific signatures in the IP packet payload. Hence, the signature based detection methods have scalability problems to work at backbone rates. In addition packet inspection is useless for encrypted data and it violates privacy. We use flow data as well as network monitoring, management and control data such as logs and SNMP traps collected from the devices in the network -router, switch, firewall, server- to construct a set of rules that describe the behavior of the attack flows in time. These rules are then used by our Detection Engine (f-IDPS) to detect layer 2-4 attacks decreasing the load on the IDPS and improving the performance of the network. Throughput of intrusion detection and prevention systems does not improve as fast as that of high speed network infrastructure. Our engine is reducing the load on the intrusion detection and prevention systems without giving up target network security. A very important contribution of our approach is demonstrating that the solutions reducing the load on the intrusion detection and prevention systems increase the whole throughput of network.

REFERENCES

[1] Umit Burak Sahin *"A new approach for the scalable intrusion Detection in high- speed networks "* Middle east technical university, 2007

[2] Hashem Alaidaros, Massudi Mahmuddin, Ali Al Mazari school of computing, University Utara Malaysia, Malaysia department of information technologies, Al Faisal university, Prince Sultan college for tourism and business Jeddah, Saudi Arabia an overview of *":Flow-based and packet-based intrusion detection performance in high speed networks".*

[3] Labib, K. *Computer security and intrusion detection*. ACM Crossroads special issue on Computer Security, Issue 11.1, Fall 2004.

[4] *Application layer attacks* By: Sameer J Ratolikar Chief Information Security Officer (CISO)2010.

[5] Flow-tools [Online]. Available: http://www.splintered.net/sw/flow-tools/, November 2011.

[6] Plonka, D., Flowscan: *A network traffic flow reporting and visualization tool,* USENIX LISA, pp. 305–317, December 2000.

[7] Harley Kozushko "Intrusion detection: host-based and network-based intrusiond systems" Thursday, September 11, 2003 Independent Study

[8]Day, K., Inside the Security Mind: Making the Tough Decisions, Prentice Hall, 2010.

[9] G. Vliek. Detecting spam at the network level. Master's thesis, Master Thesis in Computer Science, University of Twente, Feb. 2009.