International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org, editorijettcs@gmail.com
Volume 4, Issue 3, May-June 2015 ISSN 2278-6856

Transformation of honeypot raw data into structured data

[1]Majed SANAN, Mahmoud RAMMAL[2] ,Wassim RAMMAL[3]

[1]Lebanese University, Faculty of Sciences.

[2]Lebanese University, Director of center of Research & Studies

[3]Lebanese University, Faculty of Sciences.

ABSTRACT

Network analyzer is capture, recording, and analysis of network events in order to find out the source of security attacks or other problem incidents. This system addresses the major challenges in collection, examination and analysis processes. We suggest a model for collecting network data, identifying suspicious packets, examining protocol features changed and validating the attack. This model has been built with exact reference to security attacks on TCP/IP protocol[1]. The packet capture file is analyzed for important TCP/IP protocol features to mark suspicious packets. The header information encapsulated in the packet capture file is ported to a database. Rule sets designed for various TCP/IP attacks are queried on the database to calculate various statistical thresholds. This information validates the presence of attacks and will be very useful for the investigation phase. The reduced packet capture size is easy to manage as only marked packets are considered. The protocol features usually manipulated by the attackers is available in database format for next stage analysis and investigation. The model has been tested with a sample attack dataset and the results are satisfactory. The model can be extended to include attacks on other protocols.

Keywords: honeypots, network, analyzer,attacks.

1. INTRODUCTION

Network security attacks are handled by firewalls and intrusion detection systems. These tools are planned to address prevention, detection, and response view to an attack. They lack any investigative features. It is very difficult to trace back the source of attack and accuse the skillful attackers who are covering their tracks. The analysis, investigation and reconstruction of an attack cannot be based on the firewall logs and interference detection alerts.

Network analyzer is a dedicated analysis technology that enables capture, recording and analysis of network packets and events for investigative purposes. It involves observing network traffic and determining if there is an abnormality in the traffic and determining whether it indicates an attack. If it is so then the nature of the attack is also determined. When attacks are successful, detection techniques enable investigators to catch the attackers [2]. The vital goal is to provide sufficient evidence to allow the perpetrator to be prosecuted. The network analysis process includes preparation, collection, preservation, examination Analysis, investigation and presentation phases; the examination and analysis phases are most challenging and difficult [3].

We propose a network analyzer system for TCP/IP based network attacks which can be extended to any of the network attacks. This model enables experts to analyze the marked suspicious network traffic, thus enabling cost effective storage and faster analysis of high bandwidth traffic.

We identify the important features which enable security attacks on TCP/IP protocol and mark suspicious packets.

The information of protocols in the TCP/IP suite encapsulated in the packet capture file is ported to a database [4]. The protocol attributes of each packet are stored as a record. Rule sets for various TCP/IP attacks have been designed and are queried on the database to calculate many statistical parameters and thresholds. This information is used for authorizing the presence of attacks. The packet capture information in database records and related attack data is available for analysis process. This model gives the investigation phase a qualitative data.

2. BACKGROUND

2.1. TCP/IP Protocols

Protocols are sets of rules for message formats and techniques that allow machines and application programs to exchange information. These rules must be followed by each machine involved in the communication in order for the receiving host to be able to understand the message. The TCP/IP suite of protocols can be understood in terms of layers. This sketch represents the layers of the TCP/IP protocol. From the top they are, Application Layer, Transport Layer, Network Layer, Network Interface Layer, and Hardware.

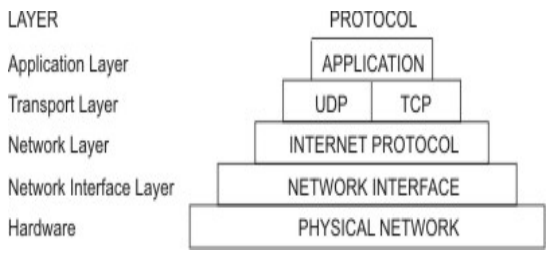

Figure 1 TCP/IP Suite of Protocols

TCP/IP sensibly defines how information moves from sender to receiver. First, application programs send messages or streams of data to one of the Internet Transport Layer Protocols, either the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP)[5]. These protocols receive the data from the application, divide it into smaller parts called packets, add a destination address, and then pass the packets along to the next protocol layer, the Internet Network layer.

The Internet Network layer surrounds the packet in an Internet Protocol (IP) datagram, puts in the datagram header and trailer, chooses where to send the datagram (either directly to a destination or else to a gateway), and passes the datagram on to the Network Interface layer.

The Network Interface layer accepts IP datagrams and spreads them as frames over specific network hardware, such as Ethernet.

In TCP/IP, there are four layers of protocols that provide the means for a system to carry data to other systems on the network. Each layer works independently of the other layers and passes the data to either the layer above it or the layer below it. The TCP/IP model defines 4 layers - The Link, Internet, Transport and Application:
In TCP/IP, there are four layers of protocols that provide the means for a system to deliver data to other systems on the network. Each layer works independently of the other layers and passes the data to either the layer above it or the layer below it. The TCP/IP model defines 4 layers - The Link, Internet, Transport and Application:

- **NETWORK ACCESS LAYER (Link):**

Is the first layer, and it deals with the physical network. It has a protocol for every physical network standard. Functions achieved at this level include encapsulating IP datagrams into the frames transferred by the network[6]. What it does is prepare the data so that it can be operated on by the level above it, the Internet Layer.

- **INTERNET LAYER:**

The most important of the TCP/IP protocol suite, its functions include defining the Internet addressing scheme and learning whether or not to pass it up or pass it on. This is where the fragmentation or reassembly of datagrams occurs. It is connectionless, meaning it does not have to connect to the system that is receiving or sending the data.

- **TRANSPORT LAYER:**

Is where the TCP exist in, TCP provides reliable data delivery by allowing that it received the data to the sending host and setting the protocol for checking for errors. TCP is responsible for providing the data to the correction application at the next layer, the Application Layer.

- **APPLICATION LAYER:**

Contains many application protocols such as **TELNET, FTP** and
SMTP (Simple Mail Transfer Protocol) which delivers electronic mail

2.2. TRAFFIC ANALYSIS WITH WIRESHARK

2.2.1. WHY WIRESHARK?

Wire shark is open-source protocol analyzers designed by Gerald Combs that runs on Windows and UNIX platforms originally known as Ethereal, its main objective is to analyze traffic as well as being an excellent, easy-to-use application for analyzing communications and resolving network
Problems
Wire shark implements a range of filters that facilitate the definition of search criteria and currently supports over 1100 protocols; all with a simple and intuitive front-end that enables you to break down the captured packets by layer.
Wire shark understands the structure of different networking protocols, so you are able to view the fields of each one of the headers and layers of the packets being monitored, providing a wide range of options to network administrators when performing certain traffic analysis tasks.

2.2.2. WHERE TO CAPTURE DATA

The first step in auditing networks is to define where to analyze the traffic.
Picture yourself in a common scenario. You find yourself in a switched environment made up of a number of switches, several terminals and a file server. Network performance has dropped in recent days and the cause is unknown.
You do not have IDS (Intrusion Detection System) that can raise the alarm or inform of attacks or network malfunction, and you know that there are no problems with the transfer rate of the file server to LAN (Local Area Network) terminals [7]. Furthermore, your network equipment does not have Net flow protocols to analyze traffic remotely, which is why you decide to use Wire shark. The first doubt that comes to mind is where to install it.
It would seem logical to install Wire shark on the file server itself to analyze the traffic that flows through this network segment, but you could come across situations in which you cannot access the server physically or quite simply for security reasons, such as
SCADA (Supervisory and Control Data Acquisition) environments, you cannot install it there.
Some alternatives will be provided with usage techniques that enable you to capture traffic without having to install Wire shark on the server[8]. The exception to the rule would be in the latter case, where several methods are given to perform remote capture in which case it is necessary to execute, or at least install, applications on the terminal you wish to analyze.

2.3. MYSQL

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary settlements. MySQL was held and supported by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation [9].

MySQL is a standard choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack.
LAMP is an acronym for "Linux, Apache, MySQL, Perl / PHP / Python[10]." Free-software-open source projects that require a full-featured database management system often use MySQL.

2.4 PHP

Stands for Hypertext Preprocessor
PHP is an HTML-embedded Web scripting language. [11] This means PHP code can be injected into the HTML of a Web page.
When a PHP page is accessed, the PHP code is read or parsed by the server the page exist in. The output from the PHP functions on the page is normally resumed as HTML code, which can be read by the browser. Because the PHP code is converted into HTML before the page is loaded, users cannot view the PHP code on a page. This make PHP pages protected enough to access databases and other secure information

2.5THE PROCESS SCENARIOS

2.5.1. First Scenario

As we mentioned above, we use the WIRESHARK in order to capture and store the requests on the main server. See figure 2

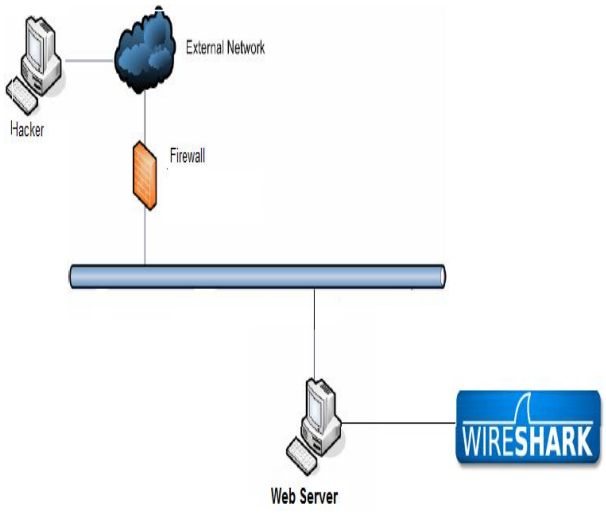

Figure 2 Flow Chart

In the figure above, we can see the hacker that sends a finite of request to the server through a network.

The role of the wire shark is to capture and save these requestsin a (.PCAP) file that shows all the IP's that makes a request on the server and from which port.

When we collect these data, we will notice that it's unclear and unstructured data, and we can't analyze it well to know reach the hacker's IP and prevent it.See the figure 3

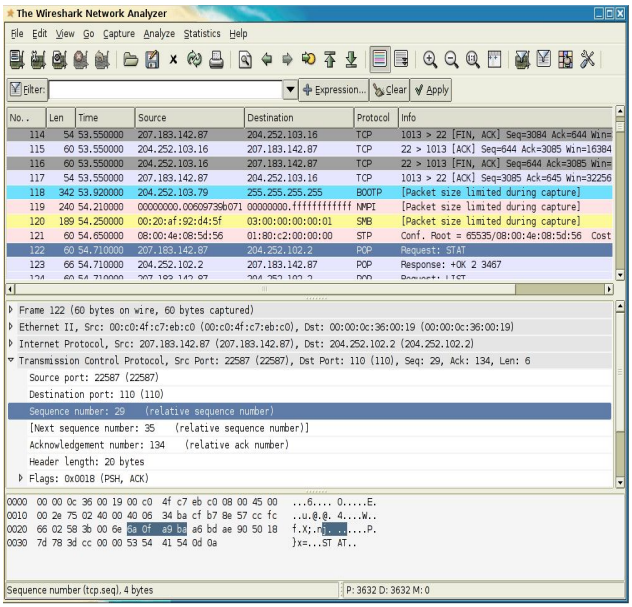

Figure3 Wire Shark Network Analyzer

As we notice in the figure the data saved in the (.PCAP) file by the wire shark, thus we work to clear these data and to make it easy to understand and analyze.

2.5.2. Second Scenario

we discuss in the first scenario how the hacker send the requests to the server and the role of the wire shark in capturing saving these request in a (.PCAP) file, also we discover that the data is unclear, that's why we manipulate these data and worked to make it easy to understand and analysis.

We use the wire shark to export the data from the (.PCP) file as an XML file, and then we developed a PHP code in order to take the data from the XML file and insert them to MYSQL table as a structured data. See the figure 4

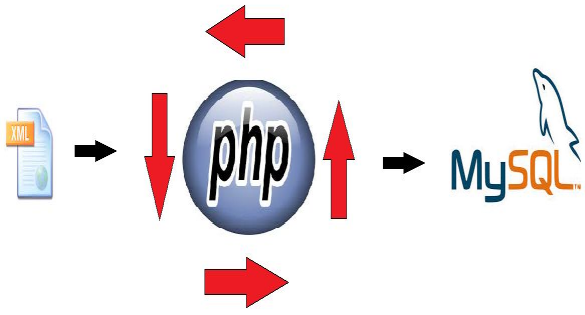

Figure 4 Data Transformation Mechanism

We notice in the figure how the PHP code manipulates the data from the XML file and insert them to the MYSQL table.

In the MYSQL table, the data become structure, easy to read and understand, and also we create a MYSQL trigger on the table to manipulate the data inserted to the table and starts structured the IP's and count the hits of each IP.

HITS:number of request done by the IP.See the Figure 5

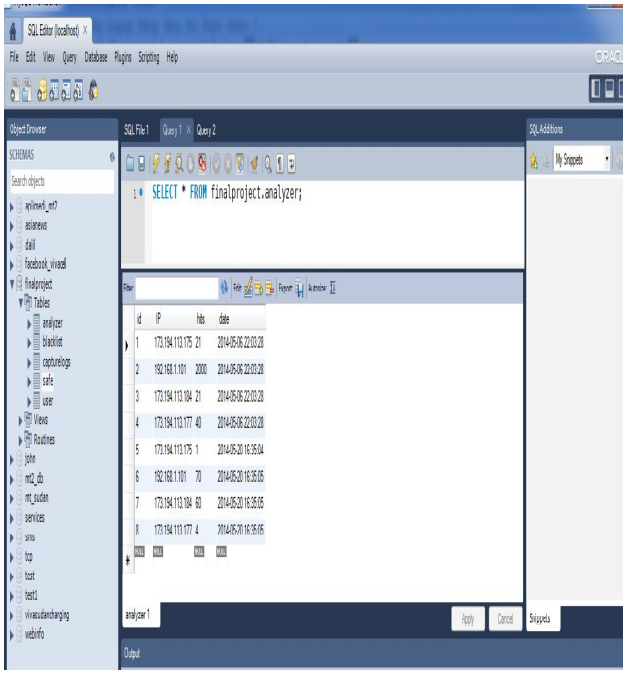

Figure 5 MYSQL Table

After this step, we notice that the data become easier to understand and structure, thus we create a simple website in order to structure the analysis step, let's take a look about the role of the web page.

Figure 6 Login page

The first page contains username and password to access it.See the Figure 6

In the second page we will see all the IP's that reaches the average number of hits (2000 hits) among one day (24 hours), then the admin will take the decisions.

Figure 7 Admin Main page

The admin can:See the Figure 7

.Block: admin can block the IP's that reaches the average numbers of hits, and insert them into theblacklist table in the database, also in structure way.

.Safe: admin can save the IP's from blocking, by inserting them into the safe table in the database.

.Details: admin can see all the information of the IP's (location, company name)

2.6 Result and Discussion

As we discussed before the main aim of this work that is all about transform the unclear data saved by the honeypot, into clear and structured data that can be analyzed and managed by the MYSQL queries. See the figure 5

After we finish the capturing phase, we noticed that the data saved by the honey pot is unclear and can't be managed; we use a PHP script to transfer the data from the (.PCAP) file into a specific MYSQL table[12].

In MYSQL; data become structured, clear and can be managed by MYSQ queries, thus we can start the analysis phase by making (2000 hits) among one day (24 hours) as a threshold for our system protocol that helps to determine the request IP of the hacker.

We create the protocol and present the result on a web interface by a PHP code that can access the MYSQL table, Moreover, we create some function to the user that give him all the rights to take his own decisions, and help him to know for whom this IP belongs to.

As a result, the main task in network analyzer is control and regulates the huge size of network packets capture[13]. Thus we focused on some specific attacks on TCP/IP protocol and have tested our approach on a two packet capture files from a victim system.

According the lack of security now a day, network analyzer can solve many security problems happened inside any company on its server; capturing, structuring, managing and analyzing data is much enough to know the hackers request attempts and prevent it.

2.7 Conclusion and Future Work

The main task in network analyzer is control the huge size of network packets capture. It is difficult to store, manage and analyze. For marking the attack packets, we linked various attacks and its corresponding recognized significant features[14]. We focused on some specific attacks on TCP/IP protocol and have tested our approach on a two packet capture files from a victim system.

We would like to extend our work to ensure that the system can handle all the TCP/IP attacks. We also would like to add some more attacks on various protocols at the network and application layers. The topology and situation effects on the thresholds also need attention. We want to automate the process of rule-sets querying the database. The analysis phase including various tools and the attack report generated by our model is under development.

References

[1] W. R. Stevens. TCP/IP Illustrated, volume 1Addison-Wesley, 1994.
[2] Network Security Baseline Americas Headquarters Cisco Systems, Inc.170 West
[3] Tasman Drive San Jose, CA 95134-1706 Internetworking Technology Overview,June1999;http://fab.cba.mit.edu/classes/MIT/961.04/people/neil/ip.pdf
[4] TCP/IP Protocol Suite; Marshal Miller, Chris Chase, Robert W. Taylor(Director of Information Processing Techniques Office at ARPA 1965-1969);http://www.inst.eecs.berkeley.edu/~ee233/sp06/student_presentations/EE233_TCPIP.pdf
[5] TCP/IP Tutorial http://web.calstatela.edu/faculty/nganesa/College%20Courses/Student%20Projects/TCP-IP/TCP-IP%20Tutorial.pdf
[6] MySQL and PHP http://downloads.mysql.com/docs/apis-php-en.pdf
[7] Wireshark Tutorial http://cs.gmu.edu/~astavrou/courses/ISA_674_F12/Wireshark-Tutorial.pdf
[8] STRUCTURED AND UNSTRUCTURED DATA;http://www.brightplanet.com/2012/06/structured-vs-unstructured-data/
[9] Jacobson, V., Leres, C. and McCanne, S. Pcap andLibpcap. Lawrence Berkeley National Laboratory,www.tcpdump.org
[10] Almulhem, A. and Traore, I. 2005. Experience withEngineering a Network Forensics System. In Proceedings of International Conference on Information Networking.
[11] Postel, J. Internet Control Message Protocol, RFC 792.http:// tools.ietf.org/html/rfc0792
[12] [Comer, D.E. and Stevens, D.L. 1991. Internetworkingwith TCP/IP.
[13] SANS Institute Reading Room. ICMP attack illustrated.http://www.sans.org/reading_room/whitepapers/threats/icmp_attacks_illustrated_477?show=477.php&cat=threats
[14] Kenney, M. Ping ofdeath.http://www.insecure.org/sploits/ping-o-death.html.